High-order finite difference methods for partial differential equations

High-order finite difference methods for partial differential equations

By Matthew Bowen

A Doctoral Thesis

Submitted in partial fulfilment of the requirements

for the award of

Doctor of Philosophy of Loughborough University

Abstract

General n-point formulae for difference operators and their errors are derived in terms of elementary symmetric functions. These are used to derive high-order, compact and parallelisable finite difference schemes for the decay-advection-diffusion and linear damped Korteweg-de Vries equations. Stability calculations are presented and the speed and accuracy of the schemes is compared to that of other finite difference methods in common use. Appendices contain useful tables of difference operators and errors and present a stability proof for quadratic inequalities. For completeness, the appendices conclude with the standard Thomas method for solving tri-diagonal systems.

Acknowledgements

I would like to thank my supervisor Professor Ron Smith of Loughborough University for his inestimable help with my thesis and the papers we have co-authored. He has taught me a huge amount on the subjects of numerical methods and fluid dynamics and for that I cannot thank him enough

Thank you to my Director of Research, Professor Phil McIver of Loughborough University, for his guidance during this work and for his help as my supervisor during my undergraduate degree.

Thank you to Dr Paul Matthews of the University of Nottingham and Professor Roger Smith of Loughborough University for their kind comments that helped tidy my work

There are too many friends and colleagues to list so if you know me, thank you.

My sponsors Natural Environment Research Council (NERC) must be thanked for the funding they provided during the first three years of my thesis.

My final gratitude is reserved for my mum, dad, and brother for their encouragement and support through the past few years and not least for providing me with somewhere to live during the final months of this work.

For my mother

Contents

Preface

Many partial differential equations (PDEs) in common use do not have solutions in 'closed form', that is to say they do not have solutions that can be expressed in terms of well known, and simple to calculate, functions. Often no analytical solutions are known or perhaps are known only in certain cases, e.g. with trivial initial and boundary conditions. For these reasons numerical methods are used in all industries throughout the world for modelling all manners of problems.

Any properly constructed and well-posed PDE that models physical processes will have solutions (after all, nature finds them) so the problem lies with how to extract these solutions, and moreover how to extract them in a reasonable time and to a high accuracy. It is a testament to the attractive properties of finite difference schemes (their ease of derivation/solution and their generally acceptable stability properties) that the Crank & Nicolson (1947) method is still in use today, more than fifty years after its publication. Refinements have of course been made. Crandall (1955) presented a high-order method and McKee & Mitchell (1970) used alternating direction implicit (ADI) methods to simplify solution in higher dimensions. More recently Smith (1999) introduced a method for deriving high order schemes that is well suited to parallel solution in higher dimensions and forms the basis of this work. Smith & Bowen (2003) extend a one-dimensional (1D) case to non-constant coefficients and demonstrate non-trivial boundary conditions.

Designing numerical schemes is often considered an art in itself, due to the apparent abundance in choice of how to go about such a task. However, with certain constraints, such as locally ensuring a scheme is accurate and forcing a parallelisable ADI structure in higher dimensions, the schemes presented here almost design themselves such that the derivation for any scheme follows essentially the same straightforward steps The basic approach is that schemes are derived by calculating weights of difference operators, arranged in such a way that in higher dimensions a parallel solution is possible. By matching expansions (in derivatives of the spatial dimensions) over an exact time-stepped framework, the scheme is tuned to a high order It is the cancellation of error terms that provides higher accuracy than that given by the term by term replacement used in traditional finite difference methods. A compact module with three points in each spatial dimension results in a solution that involves solving tri-diagonal systems Whether these systems are solved in serial or parallel, the improvement in speed by solution of tri-diagonal systems over laborious matrix inversion or relaxation methods is clear

Chapter 1 provides an introduction to the methods used, covering all areas from designing a 1D scheme for the decay-advection-diffusion equation to assessing its stability criteria and interpreting the accuracy of schemes in terms of its wave properties. Comparisons are made with several finite difference schemes including the classic Crank & Nicolson (1947) method.

In chapter 2 an explicit formula for n-point difference operators is derived in terms of elementary symmetric functions. The derivation culminates in a recurrence formula that gives the errors between the difference operators and their corresponding derivatives. This chapter has been accepted for publication (Bowen & Smith 2005a). For $n = 1, \ldots, 5$ a table of difference operators is presented in appendix A and their corresponding errors in appendix B

Chapter 3 contains the derivation of a high-order scheme for the linear damped Korteweg-de Vries (KdV) equation. The stark demonstration of this chapter is that a two time-level module with three points in the spatial dimension can be used to model the effects of a third derivative term, which would require a minimum of four points to model directly. This chapter has been submitted for publication (Smith & Bowen 2005).

The method is expanded to higher dimensions in chapter 4 where a numerical scheme for the three-dimensional (3D) advection-diffusion equation is derived. This chapter demonstrates the use of ADI methods that allow the solution to be split into stages (one for each spatial dimension), each containing tri-diagonal systems that may be solved in parallel over the remaining dimensions. The results are compared to various methods including that of McKee, Wall & Wilson (1996). This chapter has been submitted for publication (Bowen & Smith 2005b).

Chapter 5 concludes this work and details more areas to be explored. The remaining appendices C and D provide, respectively, stability proofs and the Thomas algorithm for solving tri-diagonal systems.

Chapter 1

1D decay-advection-diffusion equation

1.1 Introduction

This chapter provides a basic introduction to the methods used throughout this work. A high-order numerical scheme for the 1D decay-advection-diffusion equation is derived, along with stability conditions. Wave properties of the resulting scheme are interpreted and the scheme is compared to $\theta-$methods, including the popular Crank & Nicolson (1947) method

In operator notation, the 1D decay-advection-diffusion equation with time-dependent coefficients, is

$$\partial_t c(x,t) + L(t)c(x,t) = q(x,t)\,, \tag{1.1a}$$

where the operator

$$L(t) = \lambda(t) + u(t)\partial_x - \kappa(t)\partial_x^2\,. \tag{1.1b}$$

This is a linear parabolic PDE with one dependent variable, c, and two independent variables, x and t An application of this PDE is in modelling the dispersion of a pollutant in an estuary. Then c denotes the concentration of the pollutant, λ its decay rate, u represents its velocity (as carried by the flow), κ its diffusion and q represents forcing The dimensional scales t denotes time and x denotes space, i e. the position along the estuary

There are several extensions that can be made, many of which are explored in later chapters. For example, a third order derivative is accounted for, whilst retaining a compact scheme, in the derivation for the linear damped Korteweg-de Vries (KdV) equation in chapter 3 and chapter 4 extends the decay-advection-diffusion scheme to three dimensions on a moving grid, whilst retaining a structure suitable for fast solution.

1.2 Difference operators and errors

To introduce the methodology involved in deriving the schemes, the structure of the numerical scheme is first considered after which the exact problem is moulded into a form suitable to tune the numerical scheme to as high an order as possible. Finite difference methods involve discretising the PDE into a local module over the spatial and temporal dimensions, resulting in a system of, typically implicit, equations to solve. The precise size of this module dictates the maximum order of accuracy that can be obtained, as well as affecting the scheme's stability and the method/speed of solution. That does not, however, imply that a larger module gives a more accurate scheme; in fact the schemes derived here improve upon traditional methods using a compact module of three points in each spatial dimension, over two time-levels, so for this 1D example the module is said to be of size 3×2. Figure 1.1 shows such a module on a regular grid (with constant grid spacing Δx), although the schemes derived allow for arbitrary spacing along each dimension.

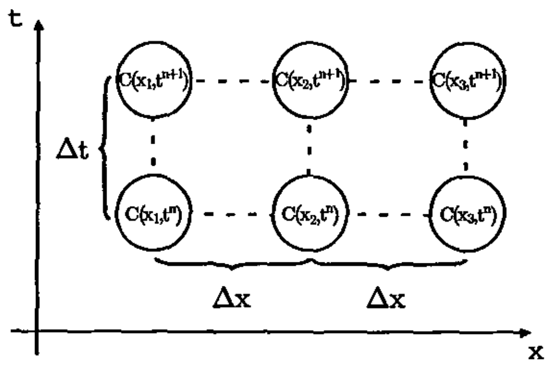

Figure 1.1 A 3×2 (3 spatial points, 2 time-levels) local module with constant spacing

Such a module size offers room for dramatic improvements (see the results in §1.6) over standard methods by better use of the available degrees of freedom, as well as retaining good stability criteria and a fast solution time (by solving tri-diagonal systems). There is, however, no need to restrict modules to this size as the tools derived in chapter 2 are applicable to any number of points.

A general discretisation, with zero forcing, can immediately be written as

$$\sum_{i=1}^{3} a_i C\left(x_i, t^{n+1}\right) = \sum_{i=1}^{3} b_i C\left(x_i, t^{n}\right) \tag{1.2}$$

with non-zero undetermined coefficients a_i and b_i providing weights to the discrete concentration $C(x_i, t^n)$ evaluated at three points x_1, x_2 and x_3 at each of the two time-levels n and $n+1$. Traditional finite difference methods such as Crank & Nicolson (1947) involve term by term discretisation of (1 1a,b), yielding the coefficients a_i and b_i. The method used here calculates the coefficients by insistence that the numerical discretisation models the operator L to as a high a degree as possible given the available degrees of freedom. With six coefficients a_i and b_i there are five degrees of freedom, since dividing (1.2) by e g. a_1 and relabelling gives the same choice of schemes Five degrees of freedom will lead to the scheme matching from the identity to fourth order with errors arising at minimum fifth order.

Smith (2000) derives schemes in terms of difference operators acting on the module instead of by direct consideration of the discrete points in (1.2) The two approaches result in identical schemes but the difference operator notation is more succinct and is thus the approach used here A full derivation of the formulae used to derive the difference operators for an arbitrary number of points is presented in chapter 2 With three spatial points x_1, x_2 and x_3, the 1D difference operators acting on the numerical concentration C, from appendix

A, are

$$
\begin{aligned}
D_x^0[C] &= \frac{\alpha_2\alpha_3 C(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)} + \frac{\alpha_1\alpha_3 C(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)} + \frac{\alpha_1\alpha_2 C(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)}, && \text{(1 3a)}\\
D_x^1[C] &= -\frac{(\alpha_2+\alpha_3) C(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)} - \frac{(\alpha_1+\alpha_3) C(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)} - \frac{(\alpha_1+\alpha_2) C(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)}, && \text{(1.3b)}\\
D_x^2[C] &= \frac{2C(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)} + \frac{2C(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)} + \frac{2C(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)}. && \text{(1.3c)}
\end{aligned}
$$

The subscript x denotes the dimension along which the operators act - chapter 4 builds a high-order scheme for the 3D decay-advection-diffusion-equation using these 1D operators acting in each of the three spatial dimensions. The superscripts denote the derivative, so that D_x^0, D_x^1 and D_x^2 are the discrete analogues of identity, first derivative and second derivative operators respectively.

The notation α_i represents a displacement from a reference point χ, so that $\alpha_i = x_i - \chi$ and $\alpha_i - \alpha_j = x_i - x_j$ In the schemes presented χ is chosen to represent the centroid of the three points along any dimension on any time-level, i e. $\chi = (x_1 + x_2 + x_3)/3$. This simplifies the error formulae below, although χ will not arise in the final scheme whatever its value. On a regular grid with spacing Δx, then $x_1 = x_2 - \Delta x$, $x_2 = \chi$ and $x_3 = x_2 + \Delta x$ so that the three-point difference operators reduce to the familiar central difference form:

$$
\begin{aligned}
D_x^0[C] &= C(x_2), && \text{(1 4a)}\\
D_x^1[C] &= \frac{C(x_3) - C(x_1)}{2\Delta x}, && \text{(1.4b)}\\
D_x^2[C] &= \frac{C(x_1) - 2C(x_2) + C(x_3)}{\Delta x^2}. && \text{(1.4c)}
\end{aligned}
$$

The discrete template (1.2) can now be written in terms of difference operators acting on the numerical concentration C^n at two time-levels n and $n+1$,

$$
\left[D_x^0 + \tfrac{1}{2}\Delta t\left(U_1^+ D_x^1 + U_2^+ D_x^2\right)\right] C^{n+1} = U_0^- \left[D_x^0 - \tfrac{1}{2}\Delta t\left(U_1^- D_x^1 + U_2^- D_x^2\right)\right] C^n, \qquad (1.5)
$$

with five degrees of freedom given by U_0^-, U_1^-, U_2^-, U_1^+ and U_2^+. In fact, the $U_i^\pm$ will be read directly from a manipulated form of the exact scheme, which will itself be written in terms of undetermined parameters M_p (introduced in §1 4). The $\frac{1}{2}\Delta t$ factors are extracted for tidiness in the matching. The difference operators (1 3a-c) may be solved for $C(x_i)$ so that the notations (1 2) and (1.5) are interchangeable.

No discrete representation of an arbitrary smooth function can be exact and errors will arise at some order Thus the difference operators (1 3a-c), exactly representing derivatives at lower orders, will have errors arising at minimum third order with three spatial points The essential step in deriving the high-order schemes requires matching future and previous time-level operators to calculate M_p and hence $U_i^\pm$ and, for this, knowledge of the errors of the difference operators is required. From appendix B, the three-point error formulae are:

$$
\begin{aligned}
D_x^0 &= I + \frac{e_3}{6}\partial_x^3 - \frac{e_2 e_3}{120}\partial_x^5 + \dots, && (1\ 6a)\\
D_x^1 &= \partial_x - \frac{e_2}{6}\partial_x^3 + \frac{e_3}{24}\partial_x^4 + \frac{e_2^2}{120}\partial_x^5 + \dots, && (1.6b)\\
D_x^2 &= \partial_x^2 - \frac{e_2}{12}\partial_x^4 + \frac{e_3}{60}\partial_x^5 + \dots. && (1\ 6c)
\end{aligned}
$$

The parameters $e_2 = \alpha_1\alpha_2 + \alpha_1\alpha_3 + \alpha_2\alpha_3$ and $e_3 = \alpha_1\alpha_2\alpha_3$ have the geometrical interpretation of measuring grid spacing and asymmetry, respectively. Formally these quantities are known as elementary symmetric functions and their advent and generalisation to an arbitrary number of points is detailed in chapter 2. Appendix B lists the error formulae with arbitrary χ and by comparison to the formulae (1.6a-c) the notational benefit of positioning χ at the centroid, and therefore making $e_1 = \alpha_1 + \alpha_2 + \alpha_3 = 0$, is evident. On a regular grid the elementary symmetric functions reduce to $e_2 = -\Delta x^2$ and $e_3 = 0$.

With I denoting the identity operator, the right-hand side of (1 6a-c) shows that the difference operators exactly mimic derivative operators up to the second derivative, beyond which order errors arise. With three points this is the best that can be achieved, although the choice of χ gives an extra degree of matching to the second order derivative (1 6c).

1.3 Exact time-stepping and time dependency

This section introduces forcing and begins the manipulation of the PDE (1 1a,b) into a form similar to (1.5), in preparation to match the numerical discretisation to as high an order as possible The PDE is transformed into an exact time-stepping form as explored by Mitchell & Griffiths (1980, chapter 2). Multiplication by an integrating factor $\exp\left(\int_0^t L(\tau)\,\mathrm{d}\tau\right)$ and integrating over a time-step of size Δt from $t = t^n$ to $t = t^{n+1} = t^n + \Delta t$ yields

$$\int_{t^n}^{t^{n+1}} \exp\left(\int_0^t L(\tau)\,\mathrm{d}\tau\right)\{\partial_t c(x,t) + Lc(x,t) - q(x,t)\}\,\mathrm{d}t = 0\,. \tag{1.7}$$

Integrating the first term by parts results in

$$\left[\exp\left(\int_0^t L(\tau)\,\mathrm{d}\tau\right)c(x,t)\right]_{t^n}^{t^{n+1}} = \int_{t^n}^{t^{n+1}} \exp\left(\int_0^t L(\tau)\,\mathrm{d}\tau\right)q(x,t)\,\mathrm{d}t\,. \tag{1 8}$$

Dividing by $\exp\left(\int_0^{t^{n+1}} L(\tau)\,\mathrm{d}\tau\right)$ and rearranging yields the exact time-stepped form of the PDE (1 1a,b)

$$c\left(x,t^{n+1}\right) = \exp\left(-\int_{t^n}^{t^{n+1}} L(\tau)\,\mathrm{d}\tau\right)c(x,t^n) + \int_{t^n}^{t^{n+1}} \exp\left(-\int_t^{t^{n+1}} L(\tau)\,\mathrm{d}\tau\right)q(x,t)\,\mathrm{d}t\,. \tag{1.9}$$

The forcing term is interpolated over the two time-levels

$$\begin{aligned}\exp\left(-\int_t^{t^{n+1}} L(\tau)\,\mathrm{d}\tau\right)q(x,t) \;&\approx\; \left(1-\frac{t-t^n}{\Delta t}\right)\exp\left(-\int_{t^n}^{t^{n+1}} L(\tau)\,\mathrm{d}\tau\right)q(x,t^n)\\ &\quad + \frac{t-t^n}{\Delta t}\,q(x,t^{n+1}) + O\left(\Delta t^3\right)\end{aligned} \tag{1.10}$$

Integration of the forcing term by the Trapezium rule leads to the particularly simple time-stepped structure with interpolated forcing (exact when forcing is absent)

$$c(x,t^{n+1}) - \tfrac{1}{2}\Delta t\, q(x,t^{n+1}) = \exp(-\Delta t\,\hat{L})\left\{c(x,t^n) + \tfrac{1}{2}\Delta t\, q(x,t^n)\right\} + O\left(\Delta t^3\right)\,, \tag{1.11}$$

where the time-averaged operator

$$\hat{L} = \tfrac{1}{\Delta t}\int_{t^n}^{t^{n+1}} L(\tau)\,\mathrm{d}\tau = \tfrac{1}{\Delta t}\int_{0}^{\Delta t} L(t^n+\tau)\,\mathrm{d}\tau\,. \tag{1.12}$$

Thus time dependent coefficients in the operator L should be integrated over the time-step and divided by the step-length With this in mind the coefficients are henceforth treated as if constant.

1.4 Viewpoint operator

The discrete form (1 5) has multipliers at both future and previous time-levels To manipulate the time-stepped equation (1.11) to such a form, a 'viewpoint' operator M (so called since it can shift between explicit and implicit views) is introduced as a truncated series of derivatives in the spatial dimension,

$$M = I + \Delta t\sum_{p=1}^{4} M_p \partial_x^p\,. \tag{1.13}$$

I denotes the identity operator and M_p are adjustable parameters For a 3×2 module (with five degrees of freedom) it is sufficient that p ranges from 1 to 4 but for differing module sizes the upper limit would need to be altered accordingly (see §3.3 for a 1D generalisation). Multiplying (1.11) by the operator $M\exp\left(\frac{1}{2}\Delta t\;(L-\lambda)\right)$ gives the desired form

$$\xi_x^+\left\{c(x,t^{n+1}) - \tfrac{1}{2}\Delta t\, q(x,t^{n+1})\right\} = \exp\left(-\lambda\Delta t\right)\xi_x^-\left\{c(x,t^n) + \tfrac{1}{2}\Delta t\, q(x,t^n)\right\}\,, \tag{1.14}$$

where the future and previous time-level operators $\xi_x^{\pm}$ are, respectively:

$$\xi_x^+ = M\exp\left(+\tfrac{1}{2}\Delta t\;(L-\lambda)\right)\,, \tag{1 15a}$$

$$\xi_x^- = M\exp\left(-\tfrac{1}{2}\Delta t\;(L-\lambda)\right)\,. \tag{1.15b}$$

With the introduction of forcing, the final form (still with unmatched parameters) of the discrete two time-level scheme is constructed in terms of the difference operators (1.3a-c), based on (1.5) and (1.11),

$$E_x^+\left\{C^{n+1}-\tfrac{1}{2}\Delta t\,Q^{n+1}\right\}=U_0^-E_x^-\left\{C^n+\tfrac{1}{2}\Delta t\,Q^n\right\}, \tag{1.16}$$

where the superscripts on the discrete concentration and forcing, C and Q, denote the time-level. The future and previous time-level discrete operators are, respectively,

$$E_x^+ = D_x^0+\tfrac{1}{2}\Delta t\left(U_1^+D_x^1+U_2^+D_x^2\right), \tag{1.17a}$$

$$E_x^- = D_x^0-\tfrac{1}{2}\Delta t\left(U_1^-D_x^1+U_2^-D_x^2\right). \tag{1.17b}$$

The exponent of an operator can be written in series form as

$$\exp(\tau L)=I+\sum_{n=1}^{\infty}\frac{\tau^n}{n!}L^n. \tag{1.18}$$

Now the multiplier U_0^- can be immediately calculated (using one degree of freedom) by matching the identity terms (so that $D_x^1=D_x^2=0$ and $\xi_x^\pm=I$) of (1.14) and (1.16), giving $U_0^-=\exp(-\lambda\Delta t)$.

With the series form (1.18), the exponential structure of the operators (1.15a,b) can be expanded as a series of derivatives and multiplied by the truncated series (1.13) so that:

$$\xi_x^+ = I+\tfrac{1}{2}\Delta t\sum_{p=1}^{4}U_p^+\partial_x^p+\dots, \tag{1.19a}$$

$$\xi_x^- = I-\tfrac{1}{2}\Delta t\sum_{p=1}^{4}U_p^-\partial_x^p+\dots. \tag{1.19b}$$

The coefficients $U_p^\pm$ are simple to extract with a computer algebra package (e g Maple or

Mathematica), giving

$$
\begin{aligned}
U_1^{\pm} &= u \pm 2M_1\,, && (1.20a)\\
U_2^{\pm} &= -\kappa \pm \tfrac{1}{4}u^2\Delta t + M_1 u\,\Delta t \pm 2M_2\,, && (1.20b)\\
U_3^{\pm} &= \mp\tfrac{1}{2}u\,\kappa\,\Delta t + \tfrac{1}{24}u^3\Delta t^2 + M_1\Delta t\left(-\kappa \pm \tfrac{1}{4}u^2\Delta t\right) + M_2 u\,\Delta t \pm 2M_3\,, && (1.20c)\\
U_4^{\pm} &= \pm\tfrac{1}{4}\kappa^2\Delta t - \tfrac{1}{8}u^2\kappa\,\Delta t^2 \pm \tfrac{1}{192}u^4\Delta t^3 + M_1 u\,\Delta t^2\left(\tfrac{1}{24}u^2\Delta t \mp \tfrac{1}{2}\kappa\right)\\
&\quad + M_2\Delta t\left(-\kappa \pm \tfrac{1}{4}u^2\Delta t\right) + M_3\,u\,\Delta t \pm 2M_4\,. && (1.20d)
\end{aligned}
$$

Each coefficient $U_i^{\pm}$ is a linear combination of the, as yet undetermined, viewpoint parameters M_i, ensuring four degrees of freedom remain in (1.17a,b).

1.5 Optimal matching

It remains to tune the numerical scheme to the exact scheme. This is accomplished by matching the operators of the numerical scheme (1.17a,b) to those of the exact scheme (1.19a,b) to as high a degree as is possible given the chosen module size, as has already taken place at the identity order. To this end, the difference operators $D_0^{\mp}$, $D_1^{\mp}$ and $D_2^{\mp}$ are substituted by their error expansions (1.6a-c). From these expansions it is clear that the difference operators are exact at orders I, ∂_x^1 and ∂_x^2 so the time-level operators (1.17a,b) immediately match their exact counterparts (1.19a,b) at these orders, whatever the choice of the adjustable parameters M_i.

By inspection (or by use of a computer algebra package), at order ∂_x^3, the relevant equations to match are

$$\pm\tfrac{1}{2}\Delta t\,U_3^{\pm} = \tfrac{1}{6}e_3 \mp \tfrac{1}{12}\Delta t\,e_2 U_1^{\pm}\,. \tag{1.21}$$

The solution of this pair of equations gives the parameters M_2 and M_3

$$\begin{aligned} M_2 &= \frac{\kappa M_1}{u} - \frac{e_2}{6\,\Delta t} - \tfrac{1}{24}u^2\Delta t\,, && (1.22a)\\ M_3 &= \frac{e_3}{6\,\Delta t} + \tfrac{1}{4}\kappa\,u\,\Delta t - M_1\left(\tfrac{1}{6}e_2 + \tfrac{1}{8}u^2\Delta t^2\right). && (1.22b)\end{aligned}$$

To avoid a singularity in M_2, the adjustable parameter M_1 is written as

$$M_1 = -Su\,. \tag{1.23}$$

Substitution of (1.22a,b) into (1.20a,b) gives the scheme parameters:

$$\begin{aligned} U_1^{\pm} &= u(1 \mp 2\,S)\,, && (1.24a)\\ U_2^{\pm} &= -\kappa(1 \pm 2\,S) + \left(\pm\tfrac{1}{6} - S\right)u^2\Delta t \mp \frac{e_2}{3\,\Delta t}\,. && (1.24b)\end{aligned}$$

At order ∂_x^4, the equations to match, after dividing through by $\pm\frac{1}{2}\Delta t$, are

$$U_4^{\pm} = \tfrac{1}{24}e_3U_1^{\pm} - \tfrac{1}{12}e_2U_2^{\pm}\,. \tag{1.25}$$

The solution of this pair of equations provides the optimal choice for the high-order parameter S.

$$S_{opt} = -\frac{2\kappa\left(e_2 + 2u^2\,\Delta t^2\right) + 3u\,e_3}{2\,\Delta t\,(12\kappa^2 + u^2e_2 + u^4\Delta t^2)}\,. \tag{1.26}$$

M_4, written in terms of S for brevity, is given by:

$$\begin{aligned} 1152M_4\,\Delta t &= 16e_2^2 + \left(3 + 32S^2\right)u^4\Delta t^4 - 48\left(3 - 8S^2\right)\kappa^2\Delta t^2\\ &\quad +16e_2\Delta t\left(\left(1 + 2S^2\right)u^2\Delta t + 8S\kappa\right) - 80Su^2\Delta t^3\kappa\,. && (1.27)\end{aligned}$$

With all available degrees of freedom used in the matching then the derivation is complete and the scheme is formally said to be high-order, with errors arising at minimum order ∂_x^5.

Save for a different method of derivation and notational differences, this scheme is identical to that derived by Smith (2000).

To summarise, the numerical scheme is

$$\begin{aligned}&\left[D_x^0+\tfrac{1}{2}\Delta t\left(U_1^+D_x^1+U_2^+D_x^2\right)\right]\left\{C^{n+1}-\tfrac{1}{2}\Delta t\,Q^{n+1}\right\}\\ =\;&\exp\left(-\lambda\Delta t\right)\left[D_x^0-\tfrac{1}{2}\Delta t\left(U_1^-D_x^1+U_2^-D_x^2\right)\right]\left\{C^n+\tfrac{1}{2}\Delta t\,Q^n\right\}\,, \qquad (1\,28)\end{aligned}$$

with parameters

$$U_1^\pm = u(1\mp 2S)\,, \qquad (1\,29a)$$

$$U_2^\pm = -\kappa(1\pm 2S)+\left(\pm\tfrac{1}{6}-S\right)u^2\Delta t\mp\frac{e_2}{3\,\Delta t}\,, \qquad (1.29b)$$

and high-order parameter S given by

$$S_{opt}=-\frac{2\kappa\left(e_2+2u^2\,\Delta t^2\right)+3u\,e_3}{2\,\Delta t\,(12\kappa^2+u^2e_2+u^4\Delta t^2)} \qquad (1.30)$$

This particular scheme is referred to as the $S=S_{opt}$ scheme. A trivial choice for S is given by $S_0=0$ A scheme with this non-optimal choice is referred to as the $S=S_0$ scheme.

With zero decay and zero velocity (i e. the diffusion equation) on a regular grid ($\lambda=0$, $u=0$, $e_2=-\Delta x^2$, $e_3=0$), the high-order parameter becomes $S=\Delta x^2/(12\kappa\,\Delta t)$ and the scheme is that of Crandall (1955).

For comparison in the subsequent sections, a family of numerical schemes known collectively as the θ−method is introduced In difference operator notation the θ−method, with zero decay and no forcing, can be written

$$\frac{C^{n+1}-C^n}{\Delta t}+u\left\{(1-\theta)\,D_x^1[C^n]+\theta D_x^1[C^{n+1}]\right\}-\kappa\left\{(1-\theta)\,D_x^2[C^n]+\theta D_x^2[C^{n+1}]\right\}=0\,, \qquad (1.31)$$

where $\theta=0$ is an explicit finite difference scheme, $\theta=\frac{1}{2}$ corresponds to Crank & Nicolson

(1947) and $\theta = 1$ is fully implicit

The solution of the 1D schemes is simple - the 3×2 module, along with boundary conditions applicable to the problem being solved, produces a tri-diagonal system that can be solved for the future time-level in terms of the known previous time-level Tri-diagonal systems can be solved in O(n) by the Thomas algorithm (appendix D). A higher dimensional scheme is derived in chapter 4 with a structure designed to take advantage of parallel computers by distributing solution of tri-diagonal systems across processors. If solving the 1D case on a parallel computer then algorithms such as recursive doubling (Stone 1973) and recursive striding (Evans 1997) may be used for an increase in speed

1.6 Numerical results

Four tests are performed consisting of a single point source of unit strength (a unit delta function) left to advect/diffuse from the centre of a grid of p points. The $S = S_{opt}$ and $S = S_0$ schemes are compared to the θ-method (1.31) with $\theta = 0$ (explicit), $\theta = \frac{1}{2}$ (Crank & Nicolson 1947) and $\theta = 1$ (fully implicit). The $S = S_{opt}$ and $S = 0$ schemes match in the long-wave (see the wave interpretation in §1.7) so such tests with short scale initial conditions are particularly severe For the PDE (1.1a,b), an exact Gaussian solution exists for an initial point source of strength s at position x':

$$c(x,t) = \frac{s}{2\sqrt{\pi\kappa t}} \exp\left[-\lambda t - \frac{1}{4\kappa t}\left(x - x' - ut\right)^2\right]. \tag{1.32}$$

This solution assumes zero concentration at infinity. For these tests the concentration is held at zero at the boundary, so that the Gaussian solution provides a valid comparison before the profile builds up at the boundary.

Standard error norms are introduced to measure the accuracy of the various schemes, compared to the exact solution (1.32):

$$l_1 = \frac{\sum\limits^{p} |C^n - c(t^n)|}{\sum\limits^{p} |c(t^n)|}, \; l_2 = \frac{\left(\sum\limits^{p} (C^n - c(t^n))^2\right)^{\frac{1}{2}}}{\left(\sum\limits^{p} c(t^n)^2\right)^{\frac{1}{2}}}, \; l_\infty = \frac{max\,|C^n - c(t^n)|}{max\,|c(t^n)|}. \tag{1.33}$$

In all tests, error norms are shown at the geometrically progressive time-steps Δt, $4\Delta t$ and $16\Delta t$. If the schemes were perfect then, in the absence of errors at the boundary, the error norms would be zero Error norms further from zero signify poorer results.

The first two tests have time-step $\Delta t = 0.2$ and are on a grid size of 21 points (from $x = 1$ to $x = 21$, with the source of strength 1 at $x = 11$). The first test is of pure diffusion with parameters

$$\Delta x = 1, \; \lambda = 0, \; u = 0, \; \kappa = 0.8 . \tag{1.34}$$

The results are shown in table 1 1. The second test introduces advection

$$\Delta x = 1, \; \lambda = 0, \; u = 1, \; \kappa = 0.8 \tag{1.35}$$

The results are shown in table 1.2. Figure 1 2 contains a plot (from $x = 8$ to $x = 20$, following the advecting solution) of these results after sixteen time-steps, with the exact solution in bold solid line.

The third and fourth tests increase the time-step Δt to 0.6 and the number of grid points to 61 (from $x = 1$ to $x = 61$ with the source of strength 1 at $x = 31$) but otherwise have the same parameters as the first two tests (1.34) and (1.35), respectively. The pure diffusion results with the increased time-step are shown in table 1 3 and the advection-diffusion results are shown in table 1.4. Figure 1 3 contains a plot (from $x = 32$ to $x = 48$, following the advecting solution) of these results after sixteen time-steps, with the exact solution in bold solid line

For both pure diffusion tests the $S = S_0$ scheme performs as well as the Crank &

Time	Scheme	l_1	l_2	l_∞
Δt	$S = S_{opt}$	0.0607	0 0477	0.0404
	$S = S_0$	0 3905	0.2710	0 2029
	$\theta = 0$	0 0521	0 0416	0 0358
	$\theta = \frac{1}{2}$	0.1048	0 0725	0 0504
	$\theta = 1$	0 2026	0.1445	0 1072
$4\Delta t$	$S = S_{opt}$	0 0142	0 0149	0.0159
	$S = S_0$	0.0740	0 0709	0 0759
	$\theta = 0$	0 0170	0 0183	0.0204
	$\theta = \frac{1}{2}$	0.1144	0 1148	0.1369
	$\theta = 1$	0.2095	0.2152	0 2565
$16\Delta t$	$S = S_{opt}$	0.0006	0 0006	0 0007
	$S = S_0$	0 0225	0 0199	0.0226
	$\theta = 0$	0.0015	0.0014	0 0017
	$\theta = \frac{1}{2}$	0 0240	0 0221	0.0272
	$\theta = 1$	0 0469	0 0440	0 0571

Table 1.1· Error norms for diffusion test (1.34) with $\Delta t = 0.2$

Time	Scheme	l_1	l_2	l_∞
Δt	$S = S_{opt}$	0 0383	0 0327	0 0248
	$S = S_0$	0.3332	0.2347	0.1790
	$\theta = 0$	0 0372	0 0322	0 0264
	$\theta = \frac{1}{2}$	0 1864	0 1431	0 1082
	$\theta = 1$	0.2886	0.2227	0.1645
$4\Delta t$	$S = S_{opt}$	0 0216	0 0207	0 0229
	$S = S_0$	0 0787	0 0671	0.0677
	$\theta = 0$	0 0803	0 0731	0.0784
	$\theta = \frac{1}{2}$	0 1726	0 1778	0 2135
	$\theta = 1$	0.2910	0.2913	0.3448
$16\Delta t$	$S = S_{opt}$	0 0039	0 0058	0.0110
	$S = S_0$	0 0216	0 0192	0 0208
	$\theta = 0$	0.0658	0.0598	0 0704
	$\theta = \frac{1}{2}$	0 0739	0 0686	0.0728
	$\theta = 1$	0 1191	0.1157	0 1139

Table 1 2 Error norms for advection-diffusion test (1 35) with $\Delta t = 0.2$

Nicolson (1947) scheme When advection is introduced the $S = S_0$ scheme improves upon the Crank & Nicolson (1947) scheme by a factor of three in the second test and a factor of ten in the fourth test, after sixteen time-steps. The $S = S_{opt}$ scheme gives a dramatic

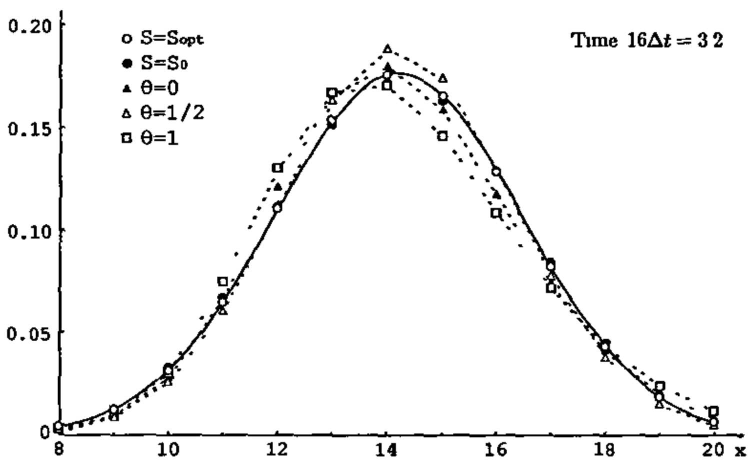

Figure 1.2 Advection-diffusion plot on a 21 point grid ($x = 1$ to 21) with $\Delta t = 0$ 2

Time	Scheme	l_1	l_2	l_∞
Δt	$S = S_{opt}$	0.1996	0.1863	0 1689
	$S = S_0$	0 5110	0 5033	0.4841
	$\theta = 0$	0.9525	0.9381	0.9018
	$\theta = \frac{1}{2}$	0 0592	0 0510	0 0526
	$\theta = 1$	0 3975	0 4073	0.4373
$4\Delta t$	$S = S_{opt}$	0.0072	0.0064	0 0065
	$S = S_0$	0 0603	0 0606	0 0593
	$\theta = 0$	0.7241	0.6927	0 6166
	$\theta = \frac{1}{2}$	0.0264	0.0231	0.0270
	$\theta = 1$	0 1135	0 1217	0.1822
$16\Delta t$	$S = S_{opt}$	0.0004	0.0004	0 0005
	$S = S_0$	0.0079	0 0073	0.0084
	$\theta = 0$	0.2636	0.2573	0 2546
	$\theta = \frac{1}{2}$	0.0072	0.0066	0 0078
	$\theta = 1$	0 0293	0 0276	0.0339

Table 1.3: Error norms for diffusion test (1.34) with $\Delta t = 0\ 6$

Time	Scheme	l_1	l_2	l_∞
Δt	$S = S_{opt}$	0 5247	0.5039	0.5000
	$S = S_0$	0.6420	0 6117	0 6156
	$\theta = 0$	0.9561	0.9866	1.0822
	$\theta = \frac{1}{2}$	0 1494	0.1473	0.1583
	$\theta = 1$	0.5201	0.5064	0 5729
$4\Delta t$	$S = S_{opt}$	0.0812	0.0784	0.0878
	$S = 0$	0.1756	0 1637	0.1747
	$\theta = 0$	0.7988	0.8650	1.1248
	$\theta = \frac{1}{2}$	0 0932	0.0853	0.0888
	$\theta = 1$	0.2516	0.2382	0 2382
$16\Delta t$	$S = S_{opt}$	0 0034	0.0031	0.0033
	$S = S_0$	0 0042	0 0040	0.0042
	$\theta = 0$	0.3365	0.3720	0 5658
	$\theta = \frac{1}{2}$	0 0467	0.0430	0.0457
	$\theta = 1$	0.1769	0 1577	0.1772

Table 1.4. Error norms for advection-diffusion test (1.35) with $\Delta t = 0\ 6$

improvement over all schemes, in particular by a factor of 36-40 for the first test, 6-18 for the second, 15-18 for the third and 14 for the fourth, over the Crank & Nicolson (1947) scheme after sixteen time-steps. The explicit scheme does particularly well in the first test

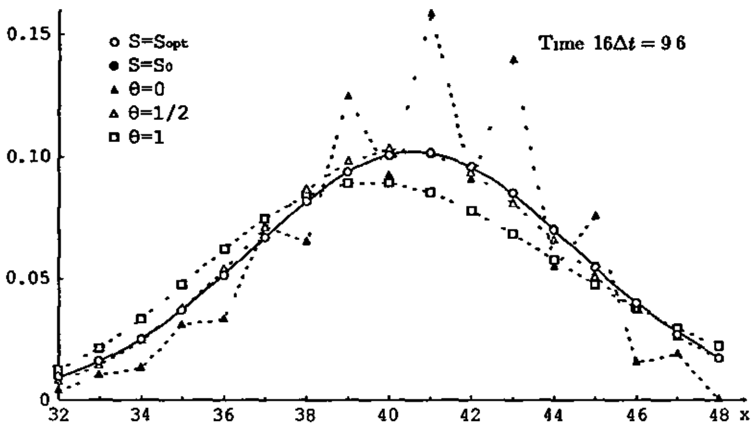

Figure 1 3. Advection-diffusion plot on a 61 point grid ($x = 1$ to 61) with $\Delta t = 0\ 6$

with pure diffusion and small time-step, with the $S = S_{opt}$ scheme being a factor of three better. But with the introduction of advection the $S = S_{opt}$ scheme pulls away

The third and fourth tests demonstrate instabilities with the explicit ($\theta = 0$) scheme, with the error norms suffering as a result. The $S = S_{opt}$ and $S = 0$ schemes are virtually indistinguishable from each other and the exact solution as shown in figures 1.2 and 1.3, the latter of which vividly demonstrates the instability of the explicit scheme

The improvement of the $S = S_{opt}$ and $S = 0$ schemes with the increased Δt is for a combination of reasons For the second test, figure 1 2 shows how the solution is starting to build up at the boundary This has a small effect on the error norms after 16 time-steps (the schemes themselves are performing correctly but the exact solution from which the error norms are calculated is becoming inappropriate). The second reason is briefly explained here and covered in more depth in §3 8 There is a particular time-step $\Delta t = \Delta x^2/(\sqrt{20}\kappa) =$ 0 2795 .. that provides an extra level of matching For these tests $\Delta t = 0\ 60 > 0\ 28$ performs slightly better than $\Delta t = 0\ 20 < 0\ 28$, demonstrating that a small time-step is not always the best choice for accuracy

1.7 Wave interpretation

Consider the Fourier component

$$c(x,t) = Ae^{i(wt-kx)}, \tag{1.36}$$

where A is a constant, k is the wavenumber and w is the angular frequency. The wavenumber denotes the number of waves that exist over a distance of 2π and the angular frequency is the number of waves that pass a fixed point over a time of 2π

Inserting (1.36) into the PDE (1 1a,b) shows that for the Fourier component to satisfy

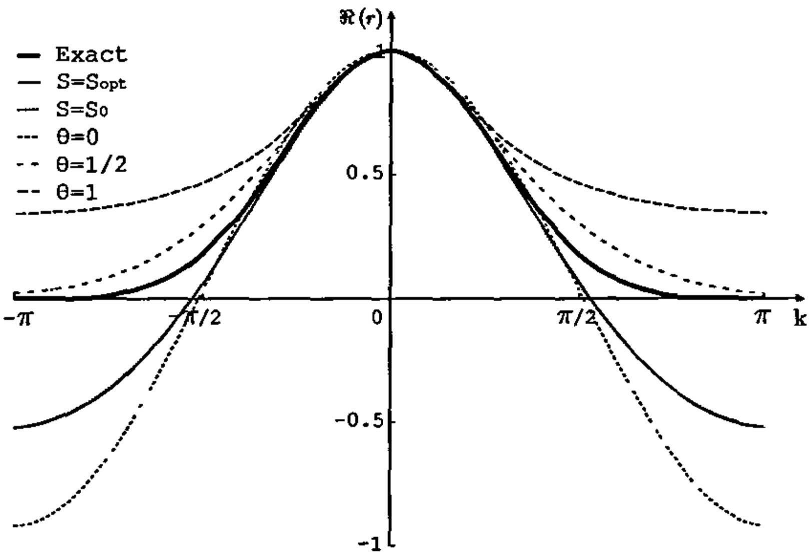

Figure 1.4. Real part of growth factors

the PDE then the dispersion relation is

$$w(k) = \mathrm{i}\lambda + ku + \imath k^2\kappa. \tag{1.37}$$

This gives the angular frequency as a function of the wavenumber. Over a time-step Δt the Fourier component (1.36) changes by a quantity formally known as the complex multiplier:

$$r(k, \Delta t) = \frac{c(x, t + \Delta t)}{c(x, t)} = \mathrm{e}^{\mathrm{i}w\Delta t} = \mathrm{e}^{-\left(\lambda - \imath ku + k^2\kappa\right)\Delta t}. \tag{1.38}$$

By inserting the spatial part $\exp(-\mathrm{i}\,k\,x)$ of the Fourier component (1.36) into the numerical schemes, the numerical complex multiplier can be calculated over a time-step Δt on a regular grid with spacing Δx. Thus knowledge of $D_x^j[\exp(-\imath\,k\,x)]$ is required, which can be calculated from (1 3a-c) as,

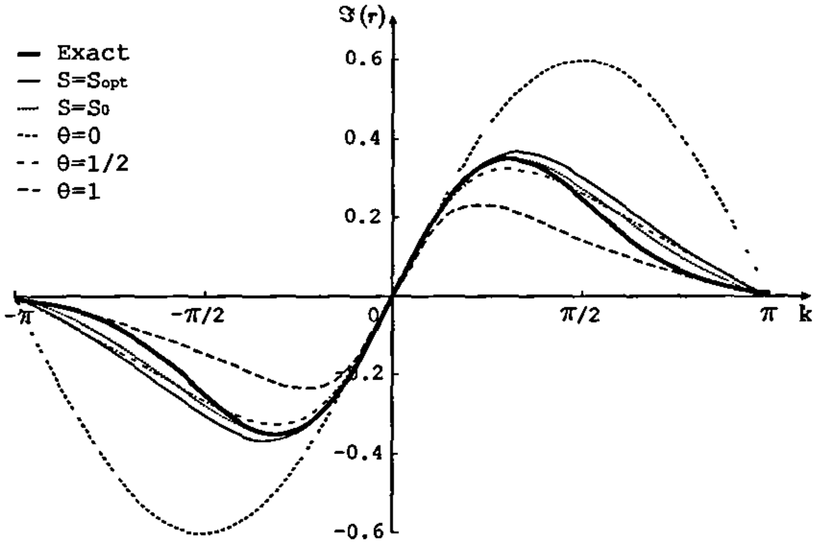

Figure 1 5 Imaginary part of growth factors

$$D_x^0[\exp(-\mathrm{i}\,k\,x)] \;=\; \exp(-\mathrm{i}\,k\,x)\;, \tag{1.39a}$$

$$D_x^1[\exp(-\mathrm{i}\,k\,x)] \;=\; -\mathrm{i}\,k\frac{cs}{\frac{1}{2}k\,\Delta x}\exp(-\mathrm{i}\,k\,x)\;, \tag{1.39b}$$

$$D_x^2[\exp(-\mathrm{i}\,k\,x)] \;=\; -k^2\frac{s^2}{(\frac{1}{2}k\Delta x)^2}\exp(-\mathrm{i}\,k\,x)\,, \tag{1.39c}$$

where for brevity $c = \cos\left(\frac{1}{2}k\,\Delta x\right)$ and $s = \sin\left(\frac{1}{2}k\,\Delta x\right)$. Then the complex multiplier for the θ-method (1.31) is

$$R\,(k,\Delta t) = 1 - \frac{2\Delta t\, s\,(u\Delta x\, c + 2\mathrm{i}\kappa\, s)}{2\theta\Delta t\, s\,(u\Delta x\, c + 2\mathrm{i}\kappa\, s) + \mathrm{i}\Delta x^2}\,. \tag{1.40}$$

For the $S = S_{opt}$ and $S = S_0$ schemes the complex multiplier, with zero decay, is given by:

$$R = \frac{D_x^0[\exp(-\mathrm{i}\,k\,x)] - \frac{1}{2}\Delta t\left(U_1^- D_x^1[\exp(-\mathrm{i}\,k\,x)] + U_2^- D_x^2[\exp(-\mathrm{i}\,k\,x)]\right)}{D_x^0[\exp(-\mathrm{i}\,k\,x)] + \frac{1}{2}\Delta t\left(U_1^+ D_x^1[\exp(-\mathrm{i}\,k\,x)] + U_2^+ D_x^2[\exp(-\mathrm{i}\,k\,x)]\right)}\,. \tag{1.41}$$

Figures 1.4 and 1.5 show the real and imaginary parts of the exact (bold line) and numerical ($S = S_{opt}$, $S = S_0$, $\theta = 0$, $\theta = \frac{1}{2}$ and $\theta = 1$) multipliers with parameters used in the fourth

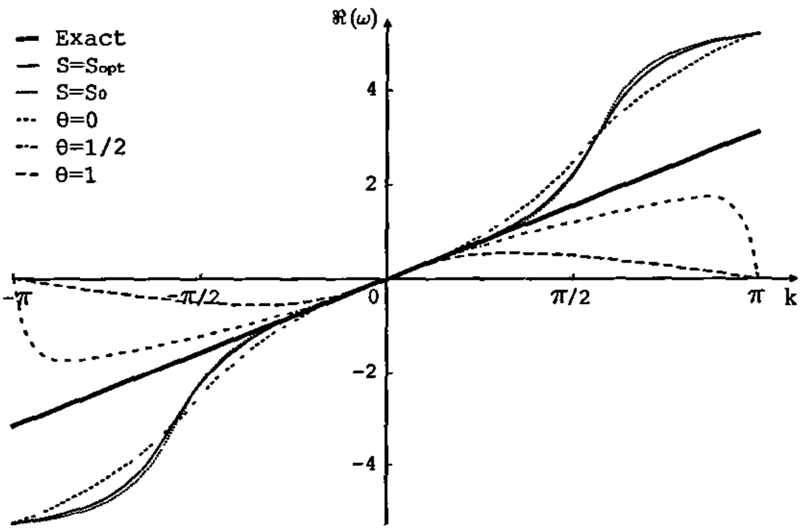

Figure 1 6: Real part of dispersion relations

test in the results §1 6, that is (1 35) with $\Delta t = 0\ 6$ The $S = S_{opt}$ and $S = S_0$ results are almost indistinguishable.

Since the numerical multipliers (1.40) and (1.41) are calculated through the use of (1.39) then it is immediately apparent that the multipliers must be periodic such that $2\pi = \frac{1}{2}k\Delta x$ In fact, by inspection of (1.39), it can be seen that the double angle trigonometric formulae, $\sin(k\Delta x) = 2cs$ and $\cos(k\Delta x) = 1 - 2s^2$, are directly applicable so that periodicity is given by $2\pi = k\Delta x$. Thus, with $\Delta x = 1$, the numerical multipliers are necessarily 2π periodic in the wavenumber k

With the component

$$C = R(k, \Delta t)^n \, e^{-ikx} \tag{1.42}$$

then the numerical growth factor R and numerical dispersion relation $W(k)$ are related by

$$e^{iWt} = R(k, \Delta t)^n \tag{1 43}$$

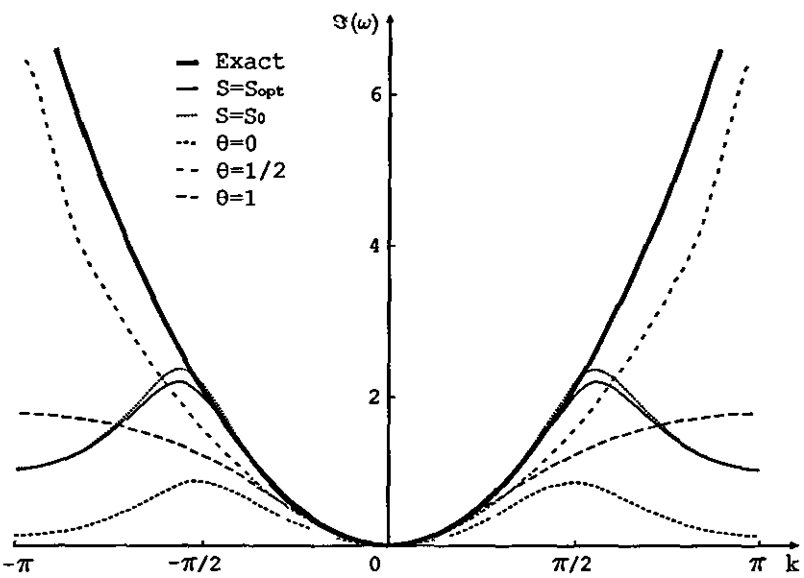

Figure 1.7: Imaginary part of dispersion relations

Thus the numerical dispersion relation, with branch cut chosen by arbitrary m, is given by

$$W(k) = -i\frac{\log R(k, \Delta t) + 2m\pi}{\Delta t}. \tag{1.44}$$

Figures 1 6 and 1.7 show the real and imaginary parts of the exact (bold line) and numerical dispersion relations, with the same parameters as before. Again, the numerical dispersion relations are necessarily 2π periodic in the wavenumber k, with the addition of branch cuts to the real part

The phase velocity is the speed and direction in which an individual component of a wave moves and is given by

$$c_p = \frac{w}{k}. \tag{1.45}$$

The group velocity denotes the speed and direction in which information is transmitted and is calculated as

$$c_g = \frac{\partial w}{\partial k}, \tag{1.46}$$

so both phase and group velocity are directly related to the dispersion plots already shown.

It is evident from all the graphs that in the long-wave region $k \approx 0$, the $S = S_{opt}$ and $S = S_0$ schemes are more accurate than the θ–methods. This is a consequence of the derivation which is equivalent to matching of the exact and numerical growth factors as expansions in the wavenumber. Indeed, that is the approach taken by Smith (2000) to derive equivalent schemes.

1.8 Stability conditions

A scheme is said to be stable if its growth over some time period is bounded. Stability for a two time-step linear PDE on a regular grid is equivalent to the condition $|R| \leqslant 1 + O(\Delta t)$ where R is the growth factor (Richtmyer & Morton 1967, §4.7). This is known as the Von Neumann stability condition and is shown in figure 1.8 for the complex case $R = \alpha + \mathrm{i}\beta$.

When deriving stability conditions it is required only to find sufficient conditions, in terms of the scheme parameters, that guarantee the Von Neumann stability condition. As long as these conditions are not too strict then they provide a framework in which the numerical scheme can be used with prior knowledge that instabilities will not arise.

The Courant-Friedrichs-Lewy (CFL) condition states that a necessary condition for stability is that the analytical domain of dependence is a subset of the numerical domain of dependence. The domain of dependence for some point (x, t) is the set of initial values which influence that point and figure 1.9(a) shows this case for a typical 1D hyperbolic equation. If the numerical domain of dependence, as shown in 1.9(b) for an explicit case, does not include the initial values of the analytical domain of dependence then there is no way that the scheme can react to changes in the initial conditions, hence the CFL condition is necessary for stability. Finally, 1.9(c) shows the typical case for an implicit numerical method in which the domain of dependence includes all initial conditions and hence the CFL condition is always satisfied.

The $O(\Delta t)$ term of the Von Neumann stability condition allows for limited growth but

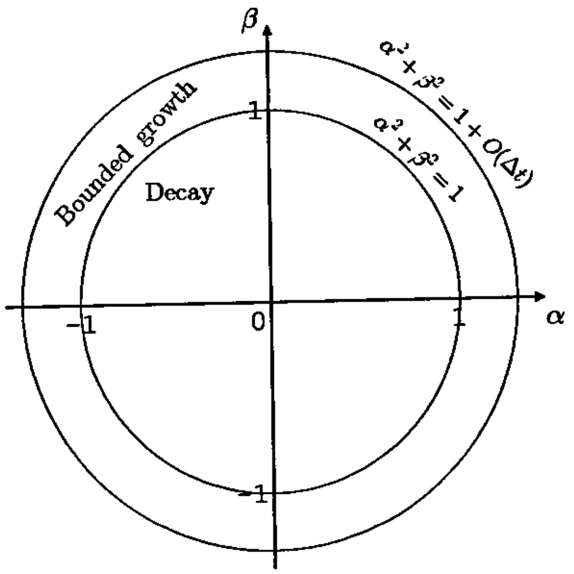

Figure 1 8· Conditions for stability

the stability is considered with $\lambda = 0$ (exponential decay ensures that with $\lambda > 0$ the scheme will also be stable). The complex growth factor (1.41), with (1 39a-c), is of the form

$$R = \frac{\alpha_1 + \beta_1 i}{\alpha_2 + \beta_2 i} \tag{1.47}$$

where

$$\alpha_1 = 3\Delta x^2 - \left(2\Delta x^2 + (1+6S)\,u^2\Delta t^2 + 6\,(1-2S)\,\kappa\Delta t\right) s^2\,, \tag{1.48a}$$

$$\beta_1 = 3\,(1+2S)\,u\Delta t\Delta x\, cs\,, \tag{1.48b}$$

$$\alpha_2 = 3\Delta x^2 - \left(2\Delta x^2 + (1-6S)\,u^2\Delta t^2 - 6\,(1+2S)\,\kappa\Delta t\right) s^2\,, \tag{1.48c}$$

$$\beta_2 = -3\,(1-2S)\,u\Delta t\Delta x\, cs\,. \tag{1.48d}$$

Then the stability constraint $|R|^2 \leqslant 1$ simplifies to

$$(\alpha_2 - \alpha_1)(\alpha_2 + \alpha_1) + (\beta_2 - \beta_1)(\beta_2 + \beta_1) \geqslant 0\,. \tag{1.49}$$

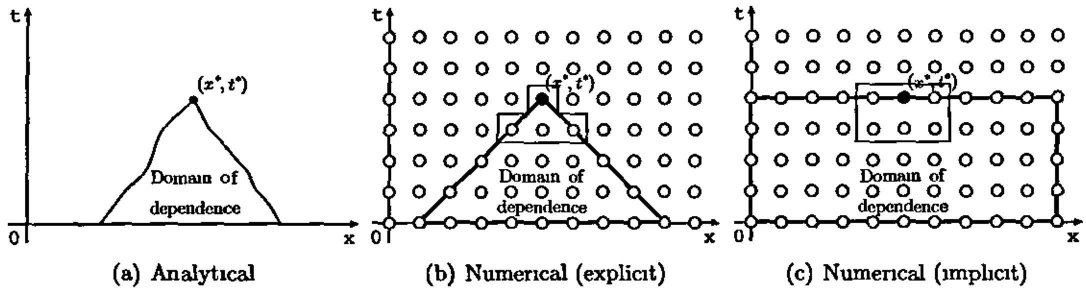

Figure 1 9 Domain of dependence for analytical and numerical cases

Inserting equations (1.48a-d), applying the trigonometric identity $c^2 = 1 - s^2$, and dividing by the positive quantity $24\Delta t\, s^2$ yields

$$3\Delta x^2 \kappa - \left[3\Delta x^2 \kappa - \left(Su^2\Delta t + \kappa\right)\left(12S\kappa\Delta t + \Delta x^2 - u^2\Delta t^2\right)\right] s^2 \geqslant 0 \qquad (1\ 50)$$

where $0 \leqslant s^2 = \sin^2\left(\frac{1}{2}k\,\Delta x\right) \leqslant 1$. Evaluating the linear (in s^2) inequality (1 50) at the end points, $s^2 = 0$ and $s^2 = 1$, gives two conditions that, when both satisfied, are sufficient for stability

$$3\Delta x^2 \kappa \geqslant 0 \text{ and } \left(Su^2\Delta t + \kappa\right)\left(12S\kappa\Delta t + \Delta x^2 - u^2\Delta t^2\right) \geqslant 0\,. \qquad (1.51)$$

The first condition holds true by definition. The second condition holds true if the high-order parameter S is constrained such that

$$S \geqslant -\frac{\kappa}{u^2\Delta t} \text{ and } S \geqslant \frac{u^2\Delta t^2 - \Delta x^2}{12\kappa\Delta t}\,. \qquad (1.52)$$

With $u = 0$ (no flow), the only requirement is

$$S \geqslant \frac{-\Delta x^2}{12\kappa\Delta t} \qquad (1\ 53)$$

With the non-optimal choice $S = S_0$, a sufficient condition for stability is the classical CFL condition $|u|\Delta t \leqslant \Delta x$.

1.9 Concluding remarks

The main aspects in deriving a high-order numerical scheme have been presented in the form of a derivation for the 1D decay-advection-diffusion equation. The $S = S_{opt}$ scheme uses all available degrees of freedom available with a 3×2 module to match future and previous time-level operators to their exact counterparts. This has the consequence of matching the numerical dispersion relation, complex multiplier, phase velocity and group velocity to the exact values in the long-wave limit, the results of which have been demonstrated graphically

Conditions sufficient for stability have been derived and the scheme has been compared to standard θ-methods, including the popular Crank & Nicolson (1947) method. The dramatic improvements possible whilst preserving a simple method of solution through solving a tri-diagonal system on each time-step have been demonstrated in tabular and graphical form for point source test cases with flow and diffusion

Chapter 2

Derivative difference operators [1]

2.1 Introduction

Computational engineering often requires the numerical solution of differential equations A natural and direct way to construct finite difference computational models is to replace the differential operators $\partial^d/\partial x^d$ at some reference point $x = \chi$ by discrete counterparts D_d corresponding to derivatives of polynomial Lagrange interpolation from the function values at $n > d$ distinct grid points $x_1, \ldots, x_n$ If the grid points are regularly spaced then the χ-dependence of the finite difference operators D_d is known explicitly and tabulated (Abramowitz & Stegun 1965, equations 25 2.7, 25.3 4-6, tables 25.1, 25.2). In applications the grid spacing might not be uniform (e g grid points to include sites where data is available or is sought). For non-uniform grids Fornberg (1988, 1998) and Corless & Rokicki (1996) give neat computer algorithms that construct D_d for $0 \leqslant d < n$. In §2.3 of the present chapter an explicit formula for D_d is derived in terms of elementary symmetric functions. Appendix A evaluates D_d in terms of the displacements $\alpha_i = x_i - \chi$ for the cases $0 \leqslant d < n$, $n = 1, \ldots, 5$.

In a term-by-term finite difference model of a differential equation, the size of the errors is related to the worst of the errors that arise in replacing $\partial^d/\partial x^d$ by D_d For a computational scheme constructed in terms of D_d, it may be possible to make slight adjustments to the coefficients multiplying each of the D_d, so that there is extra cancellation of the errors.

[1] Accepted for publication (Bowen & Smith 2005a)

Crandall (1955) performed such error cancellation with $n = 3$ and a uniform grid for the diffusion equation at two levels in time. Mitchell & Griffiths (1980, chapter 2, table 1) demonstrate the leap in accuracy over the Crank-Nicolson (1947) scheme Smith (2000) extended the Crandall (1955) scheme to include grid non-uniformity via neat Taylor series for the $n = 3$ errors in D_0, D_1, D_2. The motivation for the present chapter is to derive error Taylor series for all n. Appendix B states the first four error terms for $0 \leqslant d < n$, $n = 1, \ldots, 5$. Computer algebra packages (e g Maple or Mathematica) make it straightforward to confirm the validity for $n = 1, \ldots, 5$ of the neat error expressions.

The next section introduces elementary symmetric functions and states the main results, from which operators and errors can be constructed for any number of grid points. The subsequent four sections detail a direct derivation of the main results, involving generalised Vandermonde determinants and Schur functions (De Marchi 2001). Functions introduced by Schur in his 1901 thesis on groups of matrices are today called S or Schur functions (MacDonald 1995)

2.2 Elementary symmetric functions and main results

In this chapter, α denotes the ordered set of displacements $\alpha_i = x_i - \chi$. For the set α, the elementary symmetric functions e_i^α are defined as the sum of all distinct permutations of order i over the set. An equivalent algebraic definition (Baker 1994, MacDonald 1995) is that for arbitrary z:

$$\sum_{i=0}^{n} e_i^\alpha z^i = \prod_{i=1}^{n} (1 + \alpha_i z) \ . \qquad (2.1)$$

For indices $i < 0$ or $i > n$, it is implicit that $e_i^\alpha = 0$. The zero order elementary symmetric function is $e_0^\alpha = 1$. To minimise confusion with powers, the superscript indicating the set will usually be omitted. For example, with $n = 3$:

$$e_1 = \alpha_1 + \alpha_2 + \alpha_3 \, , \quad e_2 = \alpha_1\alpha_2 + \alpha_1\alpha_3 + \alpha_2\alpha_3 \, , \quad e_3 = \alpha_1\alpha_2\alpha_3 \ . \qquad (2\ 2)$$

The derivatives, with respect to a varied reference point, are:

$$\frac{\partial \alpha_i}{\partial \chi} = -1 \ , \ \ \frac{\partial e_j}{\partial \chi} = -(n+1-j)e_{j-1} \ , \ \ \frac{\partial}{\partial \chi}\left\{ e_j|_{\alpha_i=0}\right\} = -(n-j)e_{j-1}|_{\alpha_i=0} \qquad (2.3)$$

If the chosen reference point χ is the centroid, then there is the simplification

$$e_1 = \sum_{i=1}^{n} \alpha_i = \sum_{i=1}^{n} x_i - n\chi = 0 \,. \qquad (2.4)$$

With $e_1 = 0$, equations (B 6a-c) correspond to equations (3.3a-c) of Smith (2000) If the reference point χ coincides with any of the grid points, the simplification is

$$e_n = \prod_{i=1}^{n} (x_i - \chi) = 0 \,. \qquad (2.5)$$

On uniformly spaced grids with χ chosen to be the centroid, e_i=0 for all odd i.

In §2.3, the n-point finite difference operator D_d operating on a function $f(x)$, is shown to be the weighted sum of the function values at the grid points

$$D_d[f] = d!\,(-1)^{n-d-1} \sum_{i=1}^{n} \frac{e_{n-d-1}|_{\alpha_i=0}}{\prod\limits_{1 \leqslant j \neq i \leqslant n} (\alpha_i - \alpha_j)} f(x_i) \,. \qquad (2.6)$$

Extensive numerical tests confirm the agreement of this explicit formula (2 6) with results from the computational algorithms of Fornberg (1988, 1998) and of Corless & Rokicki (1996). $D_0[f]$ is n-point Lagrange interpolation (Abramowitz & Stegun 1965, 25 2 2) and $D_d[f]$ is the d'th derivative with respect to χ of the Lagrange interpolation (Fornberg 1988) A mathematical way of expressing the equivalence of the subscript d to the number of χ-derivatives is the consistency relationship

$$D_{d+1}[f] = \frac{\partial}{\partial \chi} D_d[f] \qquad (2.7)$$

In appendix A, the sign changes and the increasing factorial numerators between successive

$D_0, \dots, D_{n-1}$ can be explained from equations (2 3, 2.7)

If the function $f(x)$ is not a polynomial in x of degree $\leqslant n-1$, then an error will arise at degree n or beyond For uniform spacing, series for differences in terms of derivatives are well known (Abramowitz & Stegun 1965, equations 25 3.16-20) In §2 4 it is shown that the error terms from the weighted sum of Taylor series about the reference point χ, can be written as a series involving Schur functions in the displacements

$$D_d[f] - \left.\frac{\partial^d f}{\partial x^d}\right|_{x=\chi} = d!\,(-1)^{n-d-1} \sum_{j=n}^{\infty} \left(\frac{S_{\Lambda(j,d,n)}}{j!} \left.\frac{\partial^j f}{\partial x^j}\right|_{x=\chi} \right). \tag{2.8}$$

After some technical preliminaries in §2.5, it is shown in §2 6 that the higher order Schur functions can be calculated through the recurrence relation for $j \geqslant n$

$$S_{\Lambda(j,d,n)} = \sum_{k=1}^{n} (-1)^{k+1}\, e_k S_{\Lambda(j-k,d,n)} \tag{2.9}$$

Exact arithmetic avoids instability for large j. An interpretation of the left-hand side term in equation (2 8) gives the low-order error coefficients for $0 \leqslant j < n$

$$S_{\Lambda(j,d,n)} = \begin{cases} (-1)^{n-d-1}, & j = d, \\ 0, & j \neq d. \end{cases} \tag{2.10}$$

From these degree zero starting values (2.10), at the ℓ'th application the recurrence relation (2.9) generates the $j = n + \ell - 1$ term, that has homogeneous degree $n + \ell - 1 - d$ in the displacements and is polynomial of order ℓ in $e_1, \dots, e_n$. In particular, the leading four error terms presented in appendix B are respectively linear, quadratic, cubic and quartic in $e_1, \dots, e_n$.

For the errors, a consequence of the consistency relationship (2 7) is

$$D_{d+1}[f] - \left.\frac{\partial^{d+1} f}{\partial x^{d+1}}\right|_{x=\chi} = \frac{\partial}{\partial \chi} \left\{ D_d[f] - \left.\frac{\partial^d f}{\partial x^d}\right|_{x=\chi} \right\}. \tag{2 11}$$

In appendix B, the sign changes and decreasing factorial denominators for the lowest-order error terms $f^{(n)}(\chi)$ in D_d can be linked to equations (2 3, 2 11)

2.3 Derivation of difference operators

Let the operator $D_d[f]$ be the weighted sum of discrete values of a function $f(x)$ over n distinct points so that

$$D_d[f] = \sum_{i=1}^{n} w_i f(x_i) . \tag{2 12}$$

Taking the Taylor series of $f(x_i)$ about the position χ and writing $\alpha_i = x_i - \chi$:

$$D_d[f] = \sum_{i=1}^{n} \left(w_i \sum_{j=0}^{\infty} \left(\frac{\alpha_i^j}{j!} \left. \frac{\partial^j f}{\partial x^j} \right|_{x=\chi} \right) \right) . \tag{2.13}$$

To avoid convergence considerations, the circle of convergence about χ is assumed to include all the x_i Let $D_{d,m}[f]$ represent the truncated form of $D_d[f]$ with the j-summation terminated at degree $m-1$. For finite term truncations, the order of i and j summations can be exchanged

$$D_{d,m}[f] \; = \; \sum_{j=0}^{m-1} \left(\left(\sum_{i=1}^{n} w_i \frac{\alpha_i^j}{j!} \right) \left. \frac{\partial^j f}{\partial x^j} \right|_{x=\chi} \right) . \tag{2.14}$$

There are n weights w_i to be selected. The truncated operator $D_{d,n}[f]$ can be forced to represent the d'th derivative operator for $d < n$:

$$D_{d,n}[f] = \left. \frac{\partial^d f}{\partial x^d} \right|_{x=\chi} \tag{2 15}$$

With the standard notation for the Kronecker delta,

$$\delta_{ij} = \begin{cases} 1, & i = j , \\ 0, & i \neq j , \end{cases} \tag{2 16}$$

then the unit column vector $\left(\delta_{0d}, \delta_{1d}, \ldots, \delta_{(n-1)d}\right)^T$ represents the derivative to be approximated. The system to be solved can thus be written in matrix form as

$$\begin{pmatrix} 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_n \\ \frac{\alpha_1^2}{2} & \frac{\alpha_2^2}{2} & \cdots & \frac{\alpha_n^2}{2} \\ \vdots & \vdots & & \vdots \\ \frac{\alpha_1^{n-1}}{(n-1)!} & \frac{\alpha_2^{n-1}}{(n-1)!} & \cdots & \frac{\alpha_n^{n-1}}{(n-1)!} \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_n \end{pmatrix} = \begin{pmatrix} \delta_{0d} \\ \delta_{1d} \\ \delta_{2d} \\ \vdots \\ \delta_{(n-1)d} \end{pmatrix}. \tag{2.17}$$

Cramer's rule states that any system $Aw = b$ with non-zero $\det(A)$ has general solution for each component w_y of $w = (w_1, \ldots, w_n)$

$$w_y = \frac{-\det \begin{pmatrix} A & b \\ p(y) & 0 \end{pmatrix}}{\det(A)}, \tag{2.18}$$

where the unit row vector $p(y) = (\delta_{1y}, \delta_{2y}, \ldots, \delta_{ny})$ picks out the component w_y of the solution

In this form, the system (2 17), upon factoring out and cancelling factorials, has solution for each component

$$w_y = \frac{-\det\begin{pmatrix} 1 & 1 & \cdots & 1 & \delta_{0d} \\ \alpha_1 & \alpha_2 & \cdots & \alpha_n & \delta_{1d} \\ \alpha_1^2 & \alpha_2^2 & \cdots & \alpha_n^2 & 2\delta_{2d} \\ \vdots & \vdots & & \vdots & \vdots \\ \alpha_1^{n-1} & \alpha_2^{n-1} & \cdots & \alpha_n^{n-1} & (n-1)!\delta_{(n-1)d} \\ \delta_{1y} & \delta_{2y} & \cdots & \delta_{ny} & 0 \end{pmatrix}}{\det\begin{pmatrix} 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \cdots & \alpha_n^2 \\ \vdots & \vdots & & \vdots \\ \alpha_1^{n-1} & \alpha_2^{n-1} & \cdots & \alpha_n^{n-1} \end{pmatrix}}. \tag{2.19}$$

The denominator in (2.19) is a Vandermonde determinant (De Marchi 2001), hereafter denoted by $VDM(\alpha)$, in terms of the ordered set $\alpha = (\alpha_1, \dots, \alpha_n)$ It has value

$$VDM(\alpha) = \prod_{1 \leqslant j < k \leqslant n} (\alpha_k - \alpha_j) . \tag{2.20}$$

The matrix in the numerator has zero last column except for the value $d!$ at the position $(d+1, n+1)$ and it has zero last row except for 1 at the position $(n+1, y)$. As temporary notation within this section, let

$$\beta(y) = (\alpha_1, \dots, \alpha_n) \setminus (\alpha_y), \tag{2.21a}$$

a length $n-1$ ordered set of displacements that excludes α_y, and let

$$\gamma = (0, \dots, n-1) \setminus (d), \tag{2.21b}$$

a length $n-1$ ordered set of integers excluding d at position $d+1$ that arise as powers of the displacements By expansion down the last column then the last row, the numerator in (2 19) can be written

$$d!\,(-1)^{y-d-1}\det\left(\beta(y)_t^{\gamma_s}\right),\ 1\leqslant s,t\leqslant n-1. \tag{2.22}$$

The denominator can be evaluated in a way that involves $VDM(\beta(y))$:

$$\begin{aligned} VDM\,(\alpha) &= \prod_{1\leqslant j\neq y<k\neq y\leqslant n}(\alpha_k-\alpha_j)\prod_{1\leqslant j<y}(\alpha_y-\alpha_j)\prod_{y<k\leqslant n}(\alpha_k-\alpha_y)\\ &= VDM\,(\beta(y))\prod_{1\leqslant j<y}(\alpha_y-\alpha_j)\prod_{y<j\leqslant n}(\alpha_j-\alpha_y)\\ &= (-1)^{n-y}\,VDM\,(\beta(y))\prod_{1\leqslant j\neq y\leqslant n}(\alpha_y-\alpha_j). \end{aligned} \tag{2.23}$$

This is non-zero because the grid points x_j, and therefore the displacements α_j, are distinct. Then the quotient (2 19) takes the form

$$w_y=\frac{d!\,(-1)^{n-d-1}\,S_\lambda\,(\beta(y))}{\prod\limits_{1\leqslant j\neq y\leqslant n}(\alpha_y-\alpha_j)}, \tag{2.24}$$

where $S_\lambda\,(\beta)$ is a Schur function over $\beta(y)$ with partition λ (Baker 1994, MacDonald 1995):

$$S_\lambda\,(\beta(y))=\frac{\det\left(\beta(y)_t^{\gamma_s}\right)}{VDM\,(\beta(y))}. \tag{2.25}$$

Partitions can be calculated by taking the difference in the powers of the numerator and the denominator in (2.25), in reverse order (Baker 1994, MacDonald 1995) The powers in the numerator are $\gamma=(0,\ldots,n-1)\setminus(d)$ and those in the denominator are $(0,\ldots,n-2)$ so that the partition λ is given by

$$\lambda=(n-1,\ldots,0)\setminus(d)-(n-2,\ldots,0)=\left(1^{n-d-1}\right). \tag{2.26}$$

For convenience the notation a^b represents b occurrences of a e g. $(1^4) = (1,1,1,1)$. Trailing zeros in partitions are dropped as they are equivalent to multiplication of the Schur function by $e_0 = 1$. The conjugate of λ is obtained by transposing the diagram of λ to give $\lambda' = (n-d-1)$ (Baker 1994, MacDonald 1995)

The Jacobi-Trudi identity for elementary symmetric functions states (MacDonald 1995) that for an arbitrary partition λ of length ℓ:

$$S_\lambda = \det\left(e_{\lambda'_s - s + t}\right), \ 1 \leqslant s, t \leqslant \ell. \tag{2.27}$$

In this particular case with $\lambda' = (n-d-1)$ the Schur function has the simple form

$$S_\lambda(\beta) = e^{\beta(y)}_{n-d-1}. \tag{2.28}$$

This gives the explicit form of (2.24) as

$$w_y = \frac{d!\,(-1)^{n-d-1}\, e^{\beta(y)}_{n-d-1}}{\prod\limits_{1\leqslant j\neq y\leqslant n} (\alpha_y - \alpha_j)} \tag{2.29}$$

The weighted sum (2.12) over all n of the points gives the difference operator that approximates the d'th derivative

$$D_d[f] = d!\,(-1)^{n-d-1} \sum_{i=1}^{n} \frac{e^{\beta(i)}_{n-d-1}}{\prod\limits_{1\leqslant j\neq i\leqslant n} (\alpha_i - \alpha_j)} f(x_i). \tag{2.30}$$

Also,

$$\begin{aligned} \sum_{k=0}^{n} e_k^{\beta(i)} z^k &= \sum_{k=0}^{n-1} e_k^{\beta(i)} z^k + e_n^{\beta(i)} z^n \\ = \prod_{1\leqslant k\leqslant n-1} (1+\beta(i)_k z) &= \prod_{1\leqslant k\neq i\leqslant n} (1+\alpha_k z) \\ = \left(\sum_{k=0}^{n} e_k^{\alpha} z^k\right)\Bigg|_{\alpha_i=0} &= \sum_{k=0}^{n} e_k^{\alpha}|_{\alpha_i=0}\, z^k \end{aligned} \tag{2.31}$$

where the definition (2.1) of elementary symmetric functions and the result $e_n^{\beta(i)} = 0$ have been used i.e. $\beta(i)$ is only of length $n-1$. Equating powers of z gives

$$e_k^{\beta(i)} \equiv e_k^{\alpha}\big|_{\alpha_i=0} \, . \tag{2.32}$$

The temporary notation β can be replaced in (2.30), to give the result

$$D_d[f] = d!\,(-1)^{n-d-1} \sum_{i=1}^{n} \frac{e_{n-d-1}^{\alpha}\big|_{\alpha_i=0}}{\prod\limits_{1\leqslant j\neq i\leqslant n} (\alpha_i - \alpha_j)} f(x_i) \, . \tag{2.33}$$

The displacement differences $\alpha_i - \alpha_j$ can also be written as grid differences $x_i - x_j$. Thus, the denominators do not depend on χ.

$D_0[f](\chi)$ is a polynomial of degree $n-1$ in χ and can be recognised as n-point Lagrange interpolation of $f(\chi)$ (Abramowitz & Stegun 1965, 25.2.2). If a general function $f(\chi)$ is replaced by $D_0[f](\chi)$ then the grid-point values $f(x_i)$ and operators $D_d[f](\chi)$ are unchanged. That restriction to polynomials of degree $n-1$, permits $D_{d,n}$ to be replaced by D_d in the derivative matching (2.15). The freedom to vary χ implies that $D_d[f](\chi)$ is the d'th derivative with respect to χ of $D_0[f](\chi)$. Fornberg (1988) made that linkage the premise for an algorithm, rather than a consequence.

2.4 Derivation of error terms

At degree n and beyond, errors will arise. It is useful to be able to calculate the higher-order errors, for example to extend high-order numerical schemes to non-uniform grids (Smith 2000). The general difference operator can be written

$$\begin{aligned} D_d[f] &= \sum_{i=1}^{n} w_i f(x_i) = \sum_{i=1}^{n} \left(w_i \sum_{j=0}^{\infty} \frac{(x_i-\chi)^j}{j!} \frac{\partial^j f}{\partial x^j}\bigg|_{x=\chi} \right) \\ &= \sum_{i=1}^{n} \left(w_i \sum_{j=0}^{\infty} \frac{\alpha_i^j}{j!} \frac{\partial^j f}{\partial x^j}\bigg|_{x=\chi} \right) = \sum_{j=0}^{\infty} \frac{E(j)}{j!} \frac{\partial^j f}{\partial x^j}\bigg|_{x=\chi} \end{aligned} \tag{2.34}$$

where

$$E(j) = \sum_{i=1}^{n} w_i \alpha_i^j . \tag{2.35}$$

The derivation in §2 3 for the approximate derivatives ensures that with $0 \leqslant j < n$:

$$E(j) = j! \delta_{jd} \tag{2.36}$$

For $j \geqslant n$, the expression (2 19) for the weights w_i has the consequence

$$E(j) = \frac{-\det \begin{pmatrix} 1 & 1 & \cdots & 1 & \delta_{0d} \\ \alpha_1 & \alpha_2 & \cdots & \alpha_n & \delta_{1d} \\ \alpha_1^2 & \alpha_2^2 & \cdots & \alpha_n^2 & 2\delta_{2d} \\ \vdots & \vdots & & \vdots & \vdots \\ \alpha_1^{n-1} & \alpha_2^{n-1} & \cdots & \alpha_n^{n-1} & (n-1)!\delta_{(n-1)d} \\ \alpha_1^j & \alpha_2^j & \cdots & \alpha_n^j & 0 \end{pmatrix}}{\det \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \cdots & \alpha_n^2 \\ \vdots & \vdots & & \vdots \\ \alpha_1^{n-1} & \alpha_2^{n-1} & \cdots & \alpha_n^{n-1} \end{pmatrix}} . \tag{2 37}$$

The denominator is $VDM(\alpha)$ From (2 16) and by expansion down the last column, the numerator is

$$d! (-1)^{n-d-1} \det \left(\alpha_t^{\Gamma_s} \right), \ 1 \leqslant s, t \leqslant n \tag{2.38}$$

where $\Gamma = (0, 1, 2, \ldots, n-1, j) \setminus (d)$ (similar to γ but including j at position n).

Then

$$E(j) = \frac{d! (-1)^{n-d-1} \det \left(\alpha_t^{\Gamma_s} \right)}{VDM(\alpha)} = d! (-1)^{n-d-1} S_{\Lambda(j,d,n)}(\alpha) \tag{2.39}$$

where the partition is

$$\Lambda(j;d,n) = (j,n-1,\dots,0)\setminus(d) - (n-1,\dots,0) = \left(j-n+1,1^{n-d-1}\right). \tag{2.40}$$

The conjugate partition $\Lambda'(j;d,n) = \left(n-d,1^{j-n}\right)$ is of length $j-n+1$. Inserting the above expression into (2.34) gives the explicit form for the general difference operator in terms of Schur functions as

$$D_d[f] = d!\,(-1)^{n-d-1}\sum_{j=0}^{\infty}\left(\frac{S_{\Lambda(j,d,n)}(\alpha)}{j!}\left.\frac{\partial^j f}{\partial x^j}\right|_{x=\chi}\right) \tag{2.41}$$

With the initial $S_{\Lambda(j,d,n)}$ for $0 \leqslant j < n$ defined as

$$S_{\Lambda(j,d,n)} = \begin{cases} (-1)^{n-d-1}, & j = d, \\ 0, & j \neq d \end{cases} \tag{2.42}$$

then (2.41) can also be written as

$$D_d[f] - \left.\frac{\partial^d f}{\partial x^d}\right|_{x=\chi} = d!\,(-1)^{n-d-1}\sum_{j=n}^{\infty}\left(\frac{S_{\Lambda(j,d,n)}(\alpha)}{j!}\left.\frac{\partial^j f}{\partial x^j}\right|_{x=\chi}\right). \tag{2.43}$$

2.5 Preliminary results

Before the recurrence relation (2.9) is derived some preliminary results are first obtained. As used earlier, the Jacobi-Trudi identity for the conjugate partition gives the Schur functions in terms of elementary symmetric functions

$$S_{\Lambda(j,d,n)}(\alpha) = \det\left(e_{\Lambda'_s-s+t}\right),\ 1 \leqslant s,t \leqslant j-n+1. \tag{2.44}$$

where Λ'_s denotes element s of the conjugate partition $\Lambda'(j,d,n) = \left(n-d,1^{j-n}\right)$. The square matrix, of size $j-n+1$, which gives the subscripts for the elementary symmetric functions in (2 44) is

$$\left[\Lambda'_s - s + t\right]_{s,t} = \begin{pmatrix} n-d & n-d+1 & \cdots & j-d \\ 0 & 1 & \cdots & j-n \\ \vdots & \ddots & \ddots & \vdots \\ 1-j+n & \cdots & 0 & 1 \end{pmatrix}, \quad 1 \leqslant s,t \leqslant j-n+1. \qquad (2\,45)$$

By the definition (2.1), $e_i = 0$ when $i > n$ so the highest subscript that yields a non-zero elementary symmetric function is given when the subscript $i = n$. The first element $n-d$ of the conjugate partition gives the subscripts $n-d-s+t$ on the first row. So, with $s = 1$, the last non-zero elementary symmetric function e_n arises when $n-d-1+t = n$ i e. $t = d+1$ Since $j-n+1 \geqslant t$ then the first row consists of the elements $e_{n-d},\ldots,e_n$ padded with zeros for $j \geqslant n+d$ otherwise it consists of the elements $e_{n-d},\ldots,e_{j-d}$ Accordingly, the Schur function $S_{\Lambda(j,d,n)}(\alpha)$ is considered over two intervals

$$S_{\Lambda(j,d,n)}(\alpha) = \begin{cases} \det\begin{pmatrix} e_{n-d} & \cdots & e_{j-d} \\ & M^{(j-n+1)}_{j-n+1} & \end{pmatrix}, & n \leqslant j \leqslant n+d, \\ \det\begin{pmatrix} e_{n-d} & \cdots & e_n & 0 & \cdots & 0 \\ & & M^{(j-n+1)}_{j-n+1} & & & \end{pmatrix}, & j \geqslant n+d. \end{cases} \qquad (2.46)$$

For convenience the notation $M^{(x)}_i$ refers to the upper-triangular matrix M_i (of size i) with row x removed and the notation $M^{(x)}_i(y)$ refers to M_i with row x and column y removed The second row of (2 45), and hence the first row of M_{j-n+1}, has final element e_n when

$j-n=n$, so that $j=2n$, giving

$$M_{j-n+1} = \begin{pmatrix} 1 & e_1 & \cdot\ \cdot & e_n \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & e_1 \\ 0 & \cdot\cdot & 0 & 1 \end{pmatrix}, \; j=2n. \tag{2.47a}$$

The values of this matrix are a direct consequence of (2.45). The matrix is upper-triangular since for $s > t+1$ in (2.45), i.e. in the strictly lower triangular region of (2.47a), then the conjugate partition has elements $1+s-t<0$ and $e_i=0$ for $i<0$. For other values of $j \geqslant n$, (2.45) shows that the matrices M_{j-n+1} may be defined iteratively in terms of the above case (2.47a). The last rows of (2.47a) and in the second case below (2.47b) are chosen for compatibility in this iterative definition and as a result they preserve the upper-triangular nature of M_{j-n+1}.

$$M_{j-n+1} = \begin{cases} \{M_{j-n+2}\}_{k,\ell}, & 1 \leqslant k,\ell \leqslant j-n+1 < n+1, \\ \begin{pmatrix} & & & 0 \\ & & & \vdots \\ & & & 0 \\ & M_{j-n} & & e_n \\ & & & \vdots \\ & & & e_1 \\ 0 & \cdot & 0 & 1 \end{pmatrix}, & j > 2n. \end{cases} \tag{2.47b}$$

In the first case the final column goes up from 1 to e_{j-n}. In the second case the zeros at the start of the final column are a consequence of $e_i=0$ when $i>n$.

With the first row and column removed it is clear that for $i > 1$:

$$\det(M_i^{(1)}(1)) = 1\,. \tag{2.48}$$

For brevity in the following derivations this result is also assumed for the case $i = 1$. From the iterative definition it is clear that

$$M_i^{(i)}(i) = M_{i-1},\ i \geqslant 2\,. \tag{2.49}$$

Since M_i is upper triangular with unit diagonal elements then

$$\det(M_i) = 1,\ i \geqslant 1\,. \tag{2.50}$$

For $1 \leqslant k,\ \ell \leqslant i$ and $i \geqslant 2$

$$\det\left(M_i^{(k)}(\ell)\right) = \det\left(M_{\max(k,\ell)}^{(k)}(\ell)\right)\,. \tag{2.51}$$

This result is due to the trailing 1's on the leading diagonal of M_i. The determinant can be expanded up the leading diagonal until the first of either row k or column ℓ is reached when the trailing 1's end and the expansion of the determinant stops

By expansion up the leading diagonal in (2.47a,b), when $1 \leqslant k, \ell \leqslant j-n$ and $j-n \geqslant 2$,

$$\det\left(M_{j-n}^{(j-n-k+1)}(\ell)\right) = \begin{cases} \det\left(M_\ell^{(j-n-k+1)}(\ell)\right) = 0, & j-n-k+1 < \ell, \\ \det\left(M_{j-n-k+1}^{(j-n-k+1)}(\ell)\right), & j-n-k+1 \geqslant \ell \end{cases} \tag{2.52}$$

In the first case the matrix can be reduced to size $max(j-n-k+1,\ell) = \ell$ by (2.51) Since the row removed $j-n-k+1$ is less than the column removed ℓ, it can be seen by considering (2.47) that the last row is all zero, giving the zero determinant In the second case, when the row removed $j-n-k+1$ is greater than or equal to the column removed

ℓ, then the matrix can be reduced to size $max(j-n-k+1,\ell) = j-n-k+1$ by (2.51).

Expanding the determinant along the first row in (2.46) gives

$$S_{\Lambda(j,d,n)}(\alpha) = \begin{cases} \sum\limits_{\ell=1}^{j-n+1} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_{j-n+1}^{(j-n+1)}(\ell)\right), & n \leqslant j \leqslant n+d, \\ \sum\limits_{\ell=1}^{d+1} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_{j-n+1}^{(j-n+1)}(\ell)\right), & j \geqslant n+d. \end{cases} \tag{2.53}$$

By expansion of the determinant up the final column in (2 47a,b), when $\ell \leqslant j-n$ (i.e not removing the final column),

$$\det\left(M_{j-n+1}^{(j-n+1)}(\ell)\right) = \begin{cases} \sum\limits_{k=1}^{j-n} (-1)^{k+1} e_k \det\left(M_{j-n}^{(j-n-k+1)}(\ell)\right), & n < j \leqslant 2n, \\ \sum\limits_{k=1}^{n} (-1)^{k+1} e_k \det\left(M_{j-n}^{(j-n-k+1)}(\ell)\right), & j \geqslant 2n \end{cases} \tag{2.54}$$

Strictly speaking, with $j = n+1$, $\det\left(M_{j-n+1}^{(j-n+1)}(\ell)\right) = e_1$ so for compatibility with the first case above it is assumed that $\det\left(M_1^{(1)}(1)\right) = 1$

2.6 Construction of the recurrence relation

The results of the previous section form the building blocks used in deriving the recurrence relation. In accordance with the intervals over which these results are valid, $S_{\Lambda(j,d,n)}(\alpha)$ is considered for (a) low-order error terms $n \leqslant j \leqslant n+d$, (b) moderate-order error terms $n+d < j \leqslant 2n$ and (c) high-order error terms $j \geqslant 2n$ The initial values of $S_{\Lambda(j,d,n)}(\alpha)$ are defined on the interval $0 \leqslant j < n$ as in (2.42)

$$S_{\Lambda(j,d,n)}(\alpha) = \begin{cases} (-1)^{n-d-1}, & j = d, \\ 0, & j \neq d. \end{cases} \tag{2.55}$$

It is left to show that with these initial values the Schur functions $S_{\Lambda(j,d,n)}(\alpha)$ can be calculated for all $j \geqslant n$ through the recurrence relation

$$S_{\Lambda(j,d,n)}(\alpha) = \sum_{k=1}^{n} (-1)^{k+1} e_k S_{\Lambda(j-k,d,n)}(\alpha) \,. \tag{2.56}$$

2.6.1 Low-order error terms: $n \leqslant j \leqslant n+d$

For the interval $n \leqslant j \leqslant n+d$, the first case in (2.53) gives

$$\begin{aligned} S_{\Lambda(j,d,n)}(\alpha) &= \sum_{\ell=1}^{j-n+1} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_{j-n+1}^{(j-n+1)}(\ell)\right) \\ &= \sigma_1(j) + \sigma_2(j) + \sigma_3(j) \end{aligned} \tag{2.57}$$

where the summation is split up as

$$\begin{aligned} \sigma_1(j) + \sigma_2(j) &= \sum_{\ell=1}^{j-n} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_{j-n+1}^{(j-n+1)}(\ell)\right), \\ \sigma_3(j) &= (-1)^{j-n} e_{j-d}\,. \end{aligned} \tag{2.58}$$

The last case is with $\ell = j-n+1$ and (2.49) and (2.50) have been used to simplify the determinant When $j = n$ it is clear from the summation in (2.57) that $\sigma_1(j) + \sigma_2(j) = 0$ since these terms do not arise. For the remaining $j > n$, the first case of (2.54) is used to give

$$\sigma_1(j) + \sigma_2(j) = \sum_{\ell=1}^{j-n} (-1)^{\ell+1} e_{n-d+\ell-1} \left(\sum_{k=1}^{j-n} (-1)^{k+1} e_k \det\left(M_{j-n}^{(j-n-k+1)}(\ell)\right) \right). \tag{2.59}$$

The order of summation is exchanged to give

$$\sigma_1(j) + \sigma_2(j) = \sum_{k=1}^{j-n} (-1)^{k+1} e_k \left(\sum_{\ell=1}^{j-n} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_{j-n}^{(j-n-k+1)}(\ell)\right) \right). \tag{2.60}$$

The inner summation of the sum $\sigma_1(n) + \sigma_2(j)$ is split such that

$$\sigma_1(j) = \sum_{k=1}^{j-n} (-1)^{k+1} e_k \left(\sum_{\ell=1}^{j-n-k+1} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_{j-n}^{(j-n-k+1)}(\ell) \right) \right). \tag{2.61}$$

When $j = n+1$ then $\sigma_2(j) = 0$ as k only takes the value one in the outer summation hence ℓ takes all the values in the inner summation For $j > n+1$ the remaining part of the split is given by

$$\sigma_2(j) = \sum_{k=1}^{j-n} (-1)^{k+1} e_k \left(\sum_{\ell=j-n-k+2}^{j-n} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_{j-n-k+1}^{(j-n-k+1)}(\ell) \right) \right). \tag{2.62}$$

Using the second case of (2.52), since from the inner summation $j - n - k + 1 \geqslant \ell$,

$$\sigma_1(j) = \sum_{k=1}^{j-n} (-1)^{k+1} e_k \left(\sum_{\ell=1}^{j-n-k+1} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_{j-n-k+1}^{(j-n-k+1)}(\ell) \right) \right). \tag{2.63}$$

The outer summation implies that $n \leqslant j - k$. Since $k \geqslant 1$ and $j \leqslant n + d$, for this interval, then $j - k \leqslant n + d - 1$. Together, these inequalities imply that $n \leqslant j - k \leqslant n + d$ so the first case of (2.53) may be inserted with j replaced by $j - k$ to give

$$\sigma_1(j) = \sum_{k=1}^{j-n} (-1)^{k+1} e_k S_{\Lambda(j-k,d,n)}(\alpha) \tag{2.64}$$

Using the first case of (2.52), since from the inner summation $j - n - k + 1 < \ell$,

$$\sigma_2(j) = \sum_{k=1}^{j-n} (-1)^{k+1} e_k \left(\sum_{\ell=j-n-k+2}^{j-n} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_{\ell}^{(j-n-k+1)}(\ell) \right) \right) = 0. \tag{2.65}$$

The initial conditions (2.55) are used to rewrite $\sigma_3(j)$ as

$$\sigma_3(j) = (-1)^{j-n} e_{j-d} = \sum_{k=j-n+1}^{n} (-1)^{k+1} e_k S_{\Lambda(j-k,d,n)}(\alpha) \tag{2.66}$$

As $j \geqslant n$, for this interval, and from the upper limit $n \geqslant k$, then $j - k \geqslant 0$. Combining this with the lower limit gives $0 \leqslant j - k < n$ From the initial conditions (2 55), the only non-zero initial value for $S_{\Lambda(j-k,d,n)}$ arises when $j - k = d$ so that $(-1)^{k+1} e_k S_{\Lambda(j-k,d,n)}(\alpha) = (-1)^{j-n} e_{j-d}$ as required.

Finally, from (2.57), the recurrence relation over the interval $n \leqslant j \leqslant n + d$ is

$$\begin{aligned} S_{\Lambda(j,d,n)}(\alpha) &= \sum_{k=1}^{j-n} (-1)^{k+1} e_k S_{\Lambda(j-k,d,n)} + \sum_{k=j-n+1}^{n} (-1)^{k+1} e_k S_{\Lambda(j-k,d,n)}(\alpha) \\ &= \sum_{k=1}^{n} (-1)^{k+1} e_k S_{\Lambda(j-k,d,n)}(\alpha) . \end{aligned} \tag{2 67}$$

2.6.2 Moderate-order error terms: $n + d < j \leqslant 2n$

The proofs over the remaining intervals are much the same with differing summation indices For the interval $n + d < j \leqslant 2n$, the second case in (2 53) and the first case in (2 54) give

$$\begin{aligned} S_{\Lambda(j,d,n)}(\alpha) &= \sum_{\ell=1}^{d+1} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_{j-n+1}^{(j-n+1)}(\ell)\right) \\ &= \sum_{\ell=1}^{d+1} (-1)^{\ell+1} e_{n-d+\ell-1} \left(\sum_{k=1}^{j-n} (-1)^{k+1} e_k \det\left(M_{j-n}^{(j-n-k+1)}(\ell)\right)\right) . \end{aligned} \tag{2 68}$$

On exchanging the order of summation

$$\begin{aligned} S_{\Lambda(j,d,n)}(\alpha) &= \sum_{k=1}^{j-n} (-1)^{k+1} e_k \left(\sum_{\ell=1}^{d+1} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_{j-n}^{(j-n-k+1)}(\ell)\right)\right) \\ &= \sigma_1(j) + \sigma_2(j) + \sigma_3(j) , \end{aligned} \tag{2 69}$$

where the notation $\sigma_1(j)$, $\sigma_2(j)$ and $\sigma_3(j)$ is reused to again denote a split in the summation. The first part of the split is

$$\sigma_1(j) = \sum_{k=1}^{j-n-d} (-1)^{k+1} e_k \left(\sum_{\ell=1}^{d+1} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_{j-n}^{(j-n-k+1)}(\ell)\right)\right) . \tag{2 70}$$

When $j-n-d=j-n$ i e $d=0$ then the split doesn't arise hence then $\sigma_2(j)+\sigma_3(j)=0$ The remaining terms for $d>0$ are split in the inner summation to give

$$\sigma_2(j)=\sum_{k=j-n-d+1}^{j-n}(-1)^{k+1}e_k\left(\sum_{\ell=1}^{j-n-k+1}(-1)^{\ell+1}e_{n-d+\ell-1}\det\left(M_{j-n}^{(j-n-k+1)}(\ell)\right)\right)\quad(2.71)$$

and

$$\begin{aligned}\sigma_3(j) &= \sum_{k=j-n-d+1}^{j-n}(-1)^{k+1}e_k\\ &\left(\sum_{\ell=j-n-k+2}^{d+1}(-1)^{\ell+1}e_{n-d+\ell-1}\det\left(M_{j-n}^{(j-n-k+1)}(\ell)\right)\right).\end{aligned}\quad(2.72)$$

The last case of (2 53) with j replaced by $j-k$ gives

$$\sigma_1(j)=\sum_{k=1}^{j-n-d}(-1)^{k+1}e_kS_{\Lambda(j-k,d,n)}\quad(2.73)$$

since from the outer summation $j-k\geqslant n+d$. The second case of (2 52) is used on the inner summation since the limits give $j-n-k+1\geqslant\ell$ so that

$$\sigma_2(j)=\sum_{k=j-n-d+1}^{j-n}(-1)^{k+1}e_k\left(\sum_{\ell=1}^{j-n-k+1}(-1)^{\ell+1}e_{n-d+\ell-1}\det\left(M_{j-n-k+1}^{(j-n-k+1)}(\ell)\right)\right).\quad(2\ 74)$$

Then

$$\sigma_2(j)=\sum_{k=j-n-d+1}^{j-n}(-1)^{k+1}e_kS_{\Lambda(j-k,d,n)}\quad(2.75)$$

where the first case of (2 53) has been used with j replaced by $j-k$, since from the outer summation $k\leqslant j-n$ and $k\geqslant j-n-d+1$ so that $n\leqslant j-k\leqslant n+d-1$. The remaining part of the summation for $d \geqslant 1$ is

$$\sigma_3(j) = \sum_{k=j-n-d+1}^{j-n} (-1)^{k+1} e_k \left(\sum_{\ell=j-n-k+2}^{d+1} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_\ell^{(j-n-k+1)}(\ell)\right) \right) = 0, \tag{2.76}$$

by the first case in (2.52) as, from the inner summation, $\ell \geqslant j-n-k+2$ so that $j-n-k+1 < \ell$. When $j < 2n$ then the initial conditions (2.55) give

$$\sum_{k=j-n+1}^{n} (-1)^{k+1} e_k S_{\Lambda(j-k,d,n)}(\alpha) = 0. \tag{2.77}$$

This result is since $j > n+d$ for this interval and from the outer summation $n \geqslant k$, giving $j-k > d$ and so $S_{\Lambda(j-k,d,n)} = 0$ Then

$$S_{\Lambda(j,d,n)}(\alpha) = \sum_{k=1}^{j-n-d} (-1)^{k+1} e_k S_{\Lambda(j-k,d,n)}(\alpha) + \sum_{k=j-n-d+1}^{j-n} (-1)^{k+1} e_k S_{\Lambda(j-k,d,n)}. \tag{2.78}$$

With (2.77) used as required to extend the upper limit of the summation, the recurrence relation for $n+d < j \leqslant 2n$ is

$$S_{\Lambda(j,d,n)}(\alpha) = \sum_{k=1}^{n} (-1)^{k+1} e_k S_{\Lambda(j-k,d,n)}(\alpha). \tag{2.79}$$

2.6.3 High-order error terms: $j \geqslant 2n$

For the interval $j \geqslant 2n$, the second cases in (2.53) and (2.54) give

$$\begin{aligned} S_{\Lambda(j,d,n)}(\alpha) &= \sum_{\ell=1}^{d+1} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_{j-n+1}^{(j-n+1)}(\ell)\right) \\ &= \sum_{\ell=1}^{d+1} (-1)^{\ell+1} e_{n-d+\ell-1} \left(\sum_{k=1}^{n} (-1)^{k+1} e_k \det\left(M_{j-n}^{(j-n-k+1)}(\ell)\right)\right). \end{aligned} \tag{2.80}$$

Exchanging the order of summation and splitting the summations into three parts, with further reuse of the $\sigma_1(j)$, $\sigma_2(j)$ and $\sigma_3(j)$ notation, gives

$$\begin{aligned} S_{\Lambda(j,d,n)}(\alpha) &= \sum_{k=1}^{n}(-1)^{k+1}e_k\left(\sum_{\ell=1}^{d+1}(-1)^{\ell+1}e_{n-d+\ell-1}\det\left(M_{j-n}^{(j-n-k+1)}(\ell)\right)\right) \\ &= \sigma_1(j)+\sigma_2(j)+\sigma_3(j)\,. \end{aligned} \tag{2.81}$$

The first part of the split summation is

$$\begin{aligned} \sigma_1(j) &= \sum_{k=1}^{j-n-d}(-1)^{k+1}e_k\left(\sum_{\ell=1}^{d+1}(-1)^{\ell+1}e_{n-d+\ell-1}\det\left(M_{j-n-k+1}^{(j-n-k+1)}(\ell)\right)\right) \\ &= \sum_{k=1}^{j-n-d}(-1)^{k+1}e_kS_{\Lambda(j-k,d,n)}(\alpha)\,, \end{aligned} \tag{2.82}$$

where the second case of (2.52) has been used since from the summation limits $j-n-k+1\geqslant d+1\geqslant\ell$ and the second case of (2.53) has been used since from the outer summation $j-k\geqslant n+d$. For $j\geqslant 2n+d$, $\sigma_2(j)+\sigma_3(j)=0$ since $\sigma_2(j)$ and $\sigma_3(j)$ don't arise in this case. For $j<2n+d$:

$$\sigma_2(j) = \sum_{k=j-n-d+1}^{n}(-1)^{k+1}e_k\left(\sum_{\ell=1}^{j-n-k+1}(-1)^{\ell+1}e_{n-d+\ell-1}\det\left(M_{j-n-k+1}^{(j-n-k+1)}(\ell)\right)\right) \tag{2.83}$$

where the second case of (2.52) has been used since from the inner summation $j-n-k+1\geqslant\ell$. Then

$$\sigma_2(j)=\sum_{k=j-n-d+1}^{n}(-1)^{k+1}e_kS_{\Lambda(j-k,d,n)}(\alpha) \tag{2.84}$$

where the first case of (2.53) has been used since in this interval $j\geqslant 2n$ and from the upper limit $n\geqslant k$ so that $n\leqslant j-k$ and, from the lower limit, $j-k\leqslant n+d-1$ which combined give $n \leqslant j - k \leqslant n + d - 1$. The last split of the summation is

$$\sigma_3(j) = \sum_{k=j-n-d+1}^{n} (-1)^{k+1} e_k \left(\sum_{\ell=j-n-k+2}^{d+1} (-1)^{\ell+1} e_{n-d+\ell-1} \det\left(M_\ell^{(j-n-k+1)}(\ell)\right) \right) = 0\,, \tag{2.85}$$

where the first case of (2.52) has been used since from the inner summation $j-n-k+1 < \ell$. Finally, the recurrence relation for $j \geqslant 2n$ is

$$\begin{aligned} S_{\Lambda(j,d,n)}(\alpha) &= \sigma_1(j) + \sigma_2(j) \\ &= \sum_{k=1}^{n} (-1)^{k+1} e_k S_{\Lambda(j-k,d,n)}(\alpha)\,, \end{aligned} \tag{2.86}$$

completing the proof of the recurrence relation (2.56) with initial conditions (2.55).

2.7 Concluding remarks

Explicit one-dimensional difference operators D_d have been derived that mimic derivative operators $\partial^d/\partial x^d$ at a reference point χ for any number n of distinct points $x_1, \ldots, x_n$ over an irregular grid and for any derivative $d < n$. Along with these, a recurrence relation has been derived that allows calculation of Taylor series for the errors. The $n + j$'th derivative error terms are polynomials of order $j + 1$ in the elementary symmetric functions for the displacements $x_1 - \chi, \ldots, x_n - \chi$.

The Taylor series for the errors makes it simple to obtain the error from a linear sum of D_d terms e.g. when selecting coefficients in a finite difference scheme to mimic a differential equation. At all accuracy levels, the error coefficients involve polynomials in the n non-constant elementary symmetric functions $e_1, \ldots, e_n$ for the set of displacements. The difference operators D_d together with the elementary symmetric functions are a natural combination of tools with which to extend high-order numerical schemes from uniform to non-uniform grids.

Chapter 3

Linear damped Korteweg-de Vries equation[1]

3.1 Introduction

This chapter concerns the construction of high-accuracy compact finite difference schemes for a linear evolution equation that is first order in time:

$$\partial_t c + Lc = q \ . \tag{3.1}$$

The conventional approach to constructing a compact (few grid points) numerical scheme, amounts to a sum of compact numerical discretisations for ∂_t and for each of the x-derivative terms that comprise the linear operator L (Crank & Nicolson 1947) The accuracy of the sum is limited by the least accurate of the terms. High-accuracy schemes (Crandall 1955, Smith 2000, Spotz & Carey 2001) do better from consideration of the combined action for the sum of terms. The error in a low-accuracy term is compensated by small adjustments to the higher-accuracy terms.

Mitchell & Griffiths (1980) advocated the use of exact time-stepping, of infinite-order in x For discrete computational points in x, numerical schemes have the accuracy of the finite order approximations to the x structure. The present chapter gives a straightforward method for scheme construction, in which N-point difference formulae for the x-derivatives and for the errors (Bowen & Smith 2005a) lead to order $2N-2$ accuracy for the x structure.

[1]Submitted for publication (Smith & Bowen 2005)

Formally, exact time-stepping applies to vector c with variable-coefficient matrix L and vector x. For ease of exposition, the chosen test case is a single equation with one spatial-direction x and coefficients that do not vary with x:

$$L = \lambda + u\partial_x - \kappa\partial_x^2 + \tfrac{1}{6}h^2 u\partial_x^3 \quad \text{with} \quad \kappa,\, h \geqslant 0. \tag{3 2}$$

A third derivative augments the decay-advection-diffusion equation. The classical application (Korteweg & de Vries 1895) is the propagation of small amplitude surges from the sea into a shallow estuary $c(x,t)$ being the current associated with the surge, $q(x,t)$ the composite tidal and atmospheric forcing, λ the non-derivative damping, u the long-wave speed, $\kappa \geqslant 0$ diffusive or dispersion damping, and h the mean water depth. The depth and long-wave speed are related $u = (gh)^{1/2}$, where g is gravitational acceleration. It is implicit that $|c| << u$, otherwise the nonlinear term $\frac{3}{2}c\partial_x c$ should be added to the linear damped KdV (Korteweg & de Vries 1895) equation.

The many applications of KdV models and the widely-studied mathematical structure (Grimshaw 2005, Marchant & Smyth 2002), have led to a diversity of numerical schemes and to a wealth of experience in the use of the schemes (Feng & Wei 2002, Ma & Sun 2000, Soliman 2004, Yan & Shu 2002). The distinctive feature of the present work is the use of a smaller computational module than is usual. The high accuracy of the scheme allows the oscillations and skewness caused by the $\partial_x^3 c$ term to be modelled with only three points in x, even though direct numerical modelling of $\partial_x^3 c$ would have required at least four points

3.2 Exact time-stepping

As explored at length by Mitchell & Griffiths (1980, chapter 2), if the linear differential operator L is independent of time, then time-integration from one time-level t^n to the next

$t^{n+1} = t^n + \Delta t$ yields an exact time-stepping equation.

$$c(x, t^{n+1}) = \exp\left(-\Delta t\, L\right) c(x, t^n) + \int_0^{\Delta t} \exp\left(-[\Delta t - \tau]\, L\right) q(x, t^n + \tau) \mathrm{d}\tau \,. \tag{3.3}$$

Exponentials of linear differential operators have a series definition,

$$\exp\left(\tau\, L\right) = I + \sum_{n=1}^{\infty} \frac{\tau^n}{n!} L^n \,, \tag{3.4}$$

and are of infinite order in ∂_x. In the test case (3 2) the identity operator I is unity

If the forcing is non-zero and is only known at the discrete time-levels, then linear interpolation of the integrand,

$$\exp\left(-\left(\Delta t - \tau\right)\, L\right) q(x, t^n + \tau) \approx \left(1 - \frac{\tau}{\Delta t}\right) \exp\left(-\Delta t\, L\right) q(x, t^n) + \frac{\tau}{\Delta t} q(x, t^{n+1}) \,, \tag{3.5}$$

leads to an elegant approximation to the time-stepping equation:

$$c(x, t^{n+1}) - \tfrac{1}{2}\Delta t\, q(x, t^{n+1}) = \exp\left(-\Delta t\, L\right) \left\{c(x, t^n) + \tfrac{1}{2}\Delta t\, q(x, t^n)\right\} \tag{3 6}$$

Half of the forcing at time-level t^n is accounted for in the $[t^{n-1}, t^n]$ step and the other half in the subsequent $[t^n, t^{n+1}]$ step, which may be of different span. For time-dependent coefficients, it would suffice that L be replaced in equation (3 6) by its time-average over the $[t^n, t^{n+1}]$ step (see §1.3).

The variety of possible numerical schemes is associated with the selection of an operator M (non-normalised projection or viewpoint operator):

$$\begin{aligned} & M \exp\left(+\tfrac{1}{2}\Delta t\, L\right) \left\{c(x, t^{n+1}) - \tfrac{1}{2}\Delta t\, q(x, t^{n+1})\right\} \\ = \;& M \exp\left(-\tfrac{1}{2}\Delta t\, L\right) \left\{c(x, t^n) \quad + \tfrac{1}{2}\Delta t\, q(x, t^n) \quad \right\} \,. \end{aligned} \tag{3.7}$$

Explicit schemes correspond to $M = \exp\left(-\tfrac{1}{2}\Delta t\, L\right)$, while conventional two time-level implicit schemes correspond to the identity operator $M = I$. In this chapter it is asked· which viewpoint M can formally be discretised to greatest precision on compact computational modules of a given size?

3.3 Truncation of exponentials

With N-points in x, suitable viewpoint operators for the i'th module can be represented

$$M = I + \Delta t \sum_{p=1}^{2N-2} M_p \partial_x^p \,, \tag{3 8}$$

with $2N-3$ adjustable matrix or scalar constants M_p. For a constant-coefficient operator L with x-independent part L_0, the exact time-stepping equation (3 7) is re-written

$$\mathcal{E}_x^+ \left\{c(x,t^{n+1}) - \tfrac{1}{2}\Delta t\, q(x,t^{n+1})\right\} = \exp\left(-\Delta t\, L_0\right) \mathcal{E}_x^- \left\{c(x,t^n) + \tfrac{1}{2}\Delta t\, q(x,t^n)\right\} \,. \tag{3.9}$$

The operators $\mathcal{E}_x^\pm$ are defined and their finite-order truncations are denoted

$$\begin{aligned} \mathcal{E}_x^+ &\equiv M \exp\left(-\tfrac{1}{2}\Delta t\, L_0\right) \exp\left(\tfrac{1}{2}\Delta t\, L\right) = I + \tfrac{1}{2}\Delta t \sum_{p=1}^{2N-2} U_p^+ \partial_x^p + \dots \,, & \text{(3 10a)} \\ \mathcal{E}_x^- &\equiv M \exp\left(\tfrac{1}{2}\Delta t\, L_0\right) \exp\left(-\tfrac{1}{2}\Delta t\, L\right) = I - \tfrac{1}{2}\Delta t \sum_{p=1}^{2N-2} U_p^- \partial_x^p + \dots \,. & \text{(3 10b)} \end{aligned}$$

Faithfulness to the exact problem (3.1) is only possible if $2N-2$ is greater or equal to the order of the differential operator L For the third-order test case (3 2), the minimum number of grid points is $N = 3$

The coefficients $U_p^\pm$ are linear in M_q with $q \leqslant p$. For the scalar case (3.2), the first five scalar coefficients $U_p^{\pm}$ are:

$$
\begin{aligned}
U_1^{\pm} &= u \pm 2M_1\,, && \text{(3.11a)}\\
U_2^{\pm} &= -\kappa \pm \tfrac{1}{4}u^2\Delta t + M_1 u\,\Delta t \pm 2M_2\,, && \text{(3.11b)}\\
U_3^{\pm} &= \tfrac{1}{6}h^2u \mp \tfrac{1}{2}u\,\kappa\,\Delta t + \tfrac{1}{24}u^3\Delta t^2 + M_1\left(-\kappa\,\Delta t \pm \tfrac{1}{4}u^2\Delta t^2\right) + M_2 u\,\Delta t \pm 2M_3\,, && \text{(3.11c)}\\
U_4^{\pm} &= \pm\Delta t\left(\tfrac{1}{4}\kappa^2 + \tfrac{1}{12}h^2u^2\right) - \tfrac{1}{8}\kappa\,u^2\Delta t^2 \pm \tfrac{1}{192}u^4\Delta t^3 \\
&\quad + M_1\left(\tfrac{1}{6}h^2u\,\Delta t \mp \tfrac{1}{2}\kappa\,u\,\Delta t^2 + \tfrac{1}{24}u^3\Delta t^3\right) + M_2\left(-\kappa\,\Delta t \pm \tfrac{1}{4}u^2\Delta t^2\right) \\
&\quad + M_3\,u\,\Delta t \pm 2M_4\,, && \text{(3.11d)}\\
U_5^{\pm} &= \mp\tfrac{1}{12}u\,\kappa\,h^2\Delta t + \left(\tfrac{1}{48}h^2u^3 + \tfrac{1}{8}\kappa^2u\right)\Delta t^2 \mp \tfrac{1}{48}u^3\kappa\Delta t^3 + \tfrac{1}{1920}u^5\,\Delta t^4 \\
&\quad + M_1\left(\pm\Delta t^2\left(\tfrac{1}{4}\kappa^2 + \tfrac{1}{12}h^2u^2\right) - \tfrac{1}{8}\kappa\,u^2\Delta t^3 \pm \tfrac{1}{192}u^4\Delta t^4\right) \\
&\quad + M_2\left(\tfrac{1}{6}h^2u\,\Delta t \mp \tfrac{1}{2}\kappa\,u\,\Delta t^2 + \tfrac{1}{24}u^3\Delta t^3\right) + M_3\left(-\kappa\,\Delta t \pm \tfrac{1}{4}u^2\Delta t^2\right) \\
&\quad + M_4\,u\,\Delta t \pm 2M_5\,. && \text{(3.11e)}
\end{aligned}
$$

3.4 Difference counterparts to derivatives

Bickley (1941) derived N-point finite difference approximations to the derivatives at each of N uniformly spaced grid points. Chapter 2 gives the extension to non-uniform grids i.e. finite difference approximations D_x^p, with $p \leqslant N-1$, to the derivatives ∂_x^p at an arbitrary position χ. In particular, with three points x_{i-1}, x_i, x_{i+1}, the finite difference formulae are

$$
\begin{aligned}
D_x^0[f] &= \frac{(x_i-\chi)(x_{i+1}-\chi)}{(x_{i-1}-x_i)\,(x_{i-1}-x_{i+1})}f\left(x_{i-1}\right) + \frac{(x_{i-1}-\chi)(x_{i+1}-\chi)}{(x_i-x_{i-1})\,(x_i-x_{i+1})}f\left(x_i\right) \\
&\quad + \frac{(x_{i-1}-\chi)(x_i-\chi)}{(x_{i+1}-x_{i-1})\,(x_{i+1}-x_i)}f\left(x_{i+1}\right)\,, && \text{(3.12a)}\\
D_x^1[f] &= -\frac{(x_i+x_{i+1}-2\chi)}{(x_{i-1}-x_i)\,(x_{i-1}-x_{i+1})}f\left(x_{i-1}\right) - \frac{(x_{i-1}+x_{i+1}-2\chi)}{(x_i-x_{i-1})\,(x_i-x_{i+1})}f\left(x_i\right) \\
&\quad - \frac{(x_{i-1}+x_i-2\chi)}{(x_{i+1}-x_{i-1})\,(x_{i+1}-x_i)}f\left(x_{i+1}\right)\,, && \text{(3.12b)}\\
D_x^2[f] &= \frac{2}{(x_{i-1}-x_i)\,(x_{i-1}-x_{i+1})}f\left(x_{i-1}\right) + \frac{2}{(x_i-x_{i-1})\,(x_i-x_{i+1})}f\left(x_i\right) \\
&\quad + \frac{2}{(x_{i+1}-x_{i-1})\,(x_{i+1}-x_i)}f\left(x_{i+1}\right)\,. && \text{(3.12c)}
\end{aligned}
$$

The optimal scheme cannot depend upon the choice of reference position χ. The choice $\chi = \bar{x}_i \equiv \frac{1}{3}(x_{i-1} + x_i + x_{i+1})$ simplifies the representation in terms of D_x^p

The error expansions can be constructed to an arbitrary number of terms through (2.8) and (2.9)

$$D_x^p = \partial_x^p + \sum_{j=N}^{\infty} \varepsilon_j^p \partial_x^j \quad \text{i.e. } \varepsilon_j^p = 0 \text{ for } j \neq p \text{ and } \varepsilon_p^p = 1 \text{ with } 0 \leqslant j < N\,. \tag{3.13}$$

Error coefficients ε_j^p for $j \geqslant N$ are generated from the previous N coefficients, with the recurrence relation

$$\varepsilon_j^p = \sum_{\ell=1}^{N} (-1)^{j+1} \frac{(j-\ell)!}{j!} e_\ell \varepsilon_{j-\ell}^p\,, \tag{3.14}$$

where e_ℓ is the elementary symmetric function of homogeneous degree ℓ in the displacements $x_k - \chi$ for the computational module. In particular, $e_1 = \sum(x_k - \chi)$. Selecting $\chi = \frac{1}{N}\sum x_k$ gives $e_1 = 0$ and the recurrence relation (3 14) becomes

$$\varepsilon_j^p = \sum_{\ell=2}^{3} (-1)^{j+1} \frac{(j-\ell)!}{j!} e_\ell \varepsilon_{j-\ell}^p\,. \tag{3.15}$$

With $N = 3$ and $\chi = \bar{x}_i$, the quadratic and cubic elementary symmetric functions are:

$$\begin{aligned} e_2 &= (x_{i-1} - \bar{x}_i)(x_i - \bar{x}_i) + (x_{i-1} - \bar{x}_i)(x_{i+1} - \bar{x}_i) + (x_i - \bar{x}_i)(x_{i+1} - \bar{x}_i) \\ &= -\tfrac{1}{2}\left[(x_{i-1} - \bar{x}_i)^2 + (x_i - \bar{x}_i)^2 + (x_{i+1} - \bar{x}_i)^2\right] < 0\,, && \text{(3 16a)} \\ e_3 &= (x_{i-1} - \bar{x}_i)(x_i - \bar{x}_i)(x_{i+1} - \bar{x}_i)\,. && \text{(3.16b)} \end{aligned}$$

To the order of accuracy required in this chapter, the operator expansions are

$$\begin{aligned} D_x^0 &= I + \frac{e_3}{6}\partial_x^3 - \frac{e_2 e_3}{120}\partial_x^5 + \ldots\,, && \text{(3 17a)} \\ D_x^1 &= \partial_x - \frac{e_2}{6}\partial_x^3 + \frac{e_3}{24}\partial_x^4 + \frac{e_2^2}{120}\partial_x^5 + \ldots\,, && \text{(3.17b)} \\ D_x^2 &= \partial_x^2 - \frac{e_2}{12}\partial_x^4 + \frac{e_3}{60}\partial_x^5 + \ldots\,. && \text{(3 17c)} \end{aligned}$$

For uniform x-spacing $x_i = x_0 + i\Delta x$, then $\bar{x}_i = x_i$, $e_2 = -\Delta x^2$ and $e_3 = 0$. Thus, D_x^0 is precisely the computed value at the middle grid point and the errors in D_x^1, D_x^2 only involve even powers of Δx.

For ease of exposition, the computational points x_i and the grid properties e_2, e_3 are assumed to be the same at times t^n and t^{n+1} (Smith 2000). An extension to a moving grid is presented in chapter 4

3.5 Finite difference scheme

Finite difference counterparts to $\mathcal{E}_x^+$ and $\mathcal{E}_x^-$ are the combinations

$$E_x^+ = D_x^0 + \tfrac{1}{2}\Delta t \sum_{p=1}^{N-1} U_p^+ D_x^p, \tag{3.18a}$$

$$E_x^- = D_x^0 - \tfrac{1}{2}\Delta t \sum_{p=1}^{N-1} U_p^- D_x^p. \tag{3.18b}$$

The discrete counterpart to the exact time-stepping equation (3.9) is the implicit scheme

$$E_x^+ \left\{c(x,t^{n+1}) - \tfrac{1}{2}\Delta t\, q(x,t^{n+1})\right\} = \exp(-L_0\Delta t) E_x^- \left\{c(x,t^n) + \tfrac{1}{2}\Delta t\, q(x,t^n)\right\}. \tag{3.19}$$

The (scalar or matrix) coefficients for the numerical scheme are $U_1^{\pm} \ldots U_{N-1}^{\pm}$, and depend on the choice of the adjustable (scalar or matrix) constants $M_1 \ldots M_{N-1}$. The computational task remains essentially the same whatever the choice of those constants, whether the scheme be of modest accuracy or optimal.

For $N = 3$ the necessary computations are tri-diagonal, and solvable easily and efficiently with two opposite-direction computational sweeps in x (see appendix D). As N increases, so does the number of diagonals and the amount of computational processing for each of the sweeps (Sebben & Baliga 1995).

3.6 Near-optimal matching

The more accurate the matching $\mathcal{E}_x^{\pm} \approx E_x^{\pm}$ the more accurate the finite difference scheme. To assess the formal accuracy, $D_x^0 \dots D_x^{N-1}$ are replaced by their derivative expansions (3.13) up to order ∂_x^{2N-2}. Error terms first arise at order ∂_x^N. The matching conditions at order ∂_x^{N+r} are

$$\pm\tfrac{1}{2}\Delta t\, U_{N+r}^{\pm} = \varepsilon_{N+r}^{0} I \pm \tfrac{1}{2}\Delta t \sum_{p=1}^{N-1} \varepsilon_{N+r}^{p} U_p^{\pm} \quad \text{for} \quad r \geqslant 0\,, \tag{3.20}$$

which are linear in the adjustable constants $M_1, \dots, M_{N+r}$. As exemplified below, this pair of conditions (3 20) is associated with solutions for M_{N-1-r} and M_{N+r}. Starting with $r = 0$ and incrementing to $r = N - 3$ yields the span of adjustable constants $M_2 \dots M_{2N-3}$.

For $N = 3$ and $r = 0$, with error coefficients ε_3^p from equations (3 17a-c), the matching conditions (3 20) are:

$$\pm\tfrac{1}{2}\Delta t\, U_3^{\pm} = \tfrac{1}{6} e_3 \mp \tfrac{1}{12}\Delta t\, e_2 U_1^{\pm}\,. \tag{3.21}$$

Via the coefficients $U_1^{\pm}$ and $U_3^{\pm}$, there is linear dependence on M_2 and M_3 For the test case (3 2) the specific coefficients (3 11a,c) lead to the solutions:

$$M_2 = \frac{\kappa M_1}{u} - \frac{h^2 + e_2}{6\,\Delta t} - \tfrac{1}{24} u^2 \Delta t\,, \tag{3.22a}$$

$$M_3 = \frac{e_3}{6\,\Delta t} + \tfrac{1}{4}\kappa\, u\, \Delta t - M_1 \left(\tfrac{1}{6} e_2 + \tfrac{1}{8} u^2 \Delta t^2\right)\,. \tag{3 22b}$$

The possible singularity in M_2 as u tends to zero, can be removed if M_1 tends to zero with u, so it is written as

$$M_1 = -S u\,, \tag{3.23}$$

where S is an adjustable constant. A simple, but non-optimal, selection is $S = 0$.

In terms of adjustable S, the scheme coefficients are·

$$U_1^{\pm} = u(1 \mp 2S), \tag{3.24a}$$

$$U_2^{\pm} = -\kappa(1 \pm 2S) + \left(\pm\tfrac{1}{6} - S\right) u^2 \Delta t \mp \frac{e_2 + h^2}{3\,\Delta t}\,. \tag{3.24b}$$

The occurrence of h^2 in the formula (3 24b) demonstrates that account is being made for the KdV term. For $h = 0$, the scheme coefficients (3 24a,b) are equivalent to those derived by Smith (2000) for the decay-advection-diffusion equation.

3.7 Optimal matching

At $r = N-2$ the lowest index M_1 (via S) and highest index M_{2N-2} adjustable constants are determined. For $N = 3$ and $r = 1$, with ε_4^p from equations (3.17a-c), the pair of matching conditions (3 20) divided through by $\pm\frac{1}{2}\Delta t$ is

$$U_4^{\pm} = \tfrac{1}{24} e_3 U_1^{\pm} - \tfrac{1}{12} e_2 U_2^{\pm}\,. \tag{3 25}$$

For the test case (3 2), with the expressions (3.11a,b,d) for $U_1^{\pm}$, $U_2^{\pm}$, $U_4^{\pm}$ the non-changing terms are linear in S and the sign-changing terms are linear in M_4. The solution for S is:

$$S = -\frac{2\kappa(e_2 + 2h^2 + 2u^2\,\Delta t^2) + 3u\,e_3}{2\,\Delta t\,(12\kappa^2 + u^2(e_2 - 2h^2) + u^4 \Delta t^2)}\,. \tag{3 26}$$

Provided that $\kappa > 0$, there is not a singularity in S as u tends to zero The simple selection $S = 0$ is close to optimal if κ and e_3 are both small.

For $\lambda = 0$, $u = 0$, $h = 0$, $e_2 = -\Delta x^2$, $e_3 = 0$ (the diffusion equation with uniform x-spacing) then $S = \Delta x^2/(12\kappa\,\Delta t)$ and the optimal three-point scheme is that derived by Crandall (1955). The considerable improvement in computational accuracy, at negligible extra cost, over the better-known Crank & Nicolson (1947) implicit scheme is exemplified by Mitchell & Griffiths (1980, chapter 2, table 1).

With e_3 eliminated in favour of S, the selection for M_4 can be written

$$\begin{aligned}
144\,M_4\,\Delta t &= 8\,S\,\kappa\,h^2\,\Delta t - \left(6\,\kappa^2\left(3-8\,S^2\right)+u^2h^2\left(3+8\,S^2\right)\right)\Delta t^2 \\
&\quad + \left(2\,h^2+16S\,\kappa\,\Delta t+\left(2+4\,S^2\right)u^2\Delta t^2\right)e_2 - 10\,S\,u^2\kappa\,\Delta t^3 \\
&\quad + \left(\tfrac{3}{8}+4\,S^2\right)u^4\,\Delta t^4 + 2\,e_2^2\,. \qquad (3.27)
\end{aligned}$$

3.8 Exceptional case of yet more accuracy

Saul'ev (1958) noted that for the decay-diffusion equation $u=0$, $h=0$ with uniform spacing Δx, the optimal three-point implicit scheme gives yet more accuracy if the time-step Δt is tuned

$$\Delta t = \frac{\Delta x^2}{20^{1/2}\kappa}\,. \qquad (3.28)$$

This section investigates how $h \neq 0$ modifies the tuning.

To extend matching to $r = N-1$, there would be only one more adjustable constant M_{2N-1} but two more $\mathcal{E}_x^{\pm} \approx E_x^{\pm}$ matching conditions (3.20). For $N=3$ and $r=2$, with ε_5^p from equations (3.17a-c), the pair of matching conditions is

$$\pm\tfrac{1}{2}\Delta t\,U_5^{\pm} = -\tfrac{1}{120}e_2e_3 \pm \tfrac{1}{240}\Delta t\,e_2^2U_1^{\pm} \pm \tfrac{1}{120}\Delta t\,e_3U_2^{\pm}\,. \qquad (3.29)$$

For the one-variable test case (3.2) with the expressions (3.11a,b,e) for $U_1^{\pm}$, $U_2^{\pm}$, $U_5^{\pm}$, the sign-changing terms in the $\pm$ matching (3.29) do not involve M_5 and lead to a different selection for S from the previous selection (3.26).

The consistency condition for equality between the alternative S values, is

$$\begin{aligned}
0 &= u^3\left(u^2\Delta t^2+e_2-2h^2\right)\left(\left(u^2\Delta t^2+\tfrac{5}{2}e_2-5h^2\right)^2-\tfrac{9}{4}e_2^2+15h^2e_2-45h^4\right) \\
&\quad +24u^3\kappa^2\Delta t^2\left(3\,u^2\Delta t^2+5\,e_2+30\,h^2\right)+12\,u\,\kappa^2\left(9\,e_2^2+10\,h^2e_2-180\kappa^2\Delta t^2\right) \\
&\quad +27\,e_3^2u^3-108\,e_3\kappa\left(12\,\kappa^2-4\,h^2u^2-u^4\Delta t^2\right)\,. \qquad (3.30)
\end{aligned}$$

For non-uniform grids, variability of e_2 and e_3 between computational modules makes it impossible to satisfy this consistency condition

For uniform x-spacing, with e_2 constant and $e_3 = 0$, the consistency condition (3 30) can be divided by u and regarded as a tuning condition that is cubic in Δt^2 (or in $e_2 = -\Delta x^2$)

$$\begin{aligned} 0 &= u^2\left(u^2\Delta t^2 + e_2 - 2h^2\right)\left(\left(u^2\Delta t^2 + \tfrac{5}{2}e_2 - 5h^2\right)^2 - \tfrac{9}{4}e_2^2 + 15h^2 e_2 - 45h^4\right) \\ &\quad +24u^2\kappa^2\Delta t^2\left(3\,u^2\Delta t^2 + 5\,e_2 + 30\,h^2\right) + 12\,\kappa^2\left(9\,e_2^2 + 10\,h^2 e_2 - 180\kappa^2\Delta t^2\right)\,. \end{aligned} \tag{3 31}$$

There can be either one or three real roots for Δt^2 (or for $e_2 = -\Delta x^2$).

In the limit $u = 0$, the last group of terms in the tuning condition (3.31) leads to the single solution:

$$\Delta t = \frac{\Delta x^2}{20^{1/2}\kappa}\left(1 - \frac{10h^2}{9\Delta x^2}\right)^{1/2}\,. \tag{3 32}$$

For the time-step Δt to be real, this variant of the Saul'ev (1958) tuning (3.28) is restricted to $\Delta x > 1.054\,h$ i e. to grid spacing greater than water depth.

In the limit $\kappa = 0$, the first line of (3.31) leads to three real solutions for Δt^2.

$$u^2\Delta t^2 = \Delta x^2 + 2h^2\,, \tag{3.33a}$$

$$u^2\Delta t^2 = \tfrac{5}{2}\Delta x^2 + 5h^2 \pm \tfrac{1}{2}\left(9\Delta x^4 + 60\,\Delta x^2 h^2 + 180h^4\right)^{1/2}\,. \tag{3 33b}$$

Two of these tunings are beyond the classical CFL (Courant-Friedrichs-Lewy) condition ($|u|\Delta t \leqslant \Delta x$) that the distance moved in one time-step should be no more than one grid spacing, making numerical stability questionable. The third tuning, associated with the minus square root, has a restriction $\Delta x > 1\ 31h$ if Δt^2 is to be positive.

Equations (3 32, 3 33a,b), exemplify that there are circumstances in which one more order of scheme accuracy is achievable. Alas, such circumstances seem elusive and restricted to uniform grids and an interval of Δx^2 for which the cubic (3.31) has a real positive root Δt^2 for general (u, κ, h) has not been found.

3.9 Stability conditions

Formal high accuracy between time-steps on a single computational module need not coincide with computational stability (Mitchell & Griffiths 1980, §2 7). This section addresses computational stability for uniform x-spacing.

For a Fourier component of the error of amplitude a on a uniform grid

$$c(x, t^n) = a \exp(-\mathrm{i}\, k\, x)\,, \tag{3.34a}$$

the corresponding error at the next time-step can be written

$$c(x, t^{n+1}) = a\, R \exp(-\mathrm{i}\, k\, x - L_0 \Delta t)\,, \tag{3.34b}$$

where the complex multiplier R is the quotient

$$R = \frac{D_x^0[\exp(-\mathrm{i}\, k\, x)] - \frac{1}{2}\Delta t \sum_{p=1}^{N-1} U_p^- D_x^p[\exp(-\mathrm{i}\, k\, x)]}{D_x^0[\exp(-\mathrm{i}\, k\, x)] + \frac{1}{2}\Delta t \sum_{p=1}^{N-1} U_p^+ D_x^p[\exp(-\mathrm{i}\, k\, x)]}\,. \tag{3.34c}$$

The condition for stability, and avoiding relative growth of errors, is that $|R|^2 \leqslant 1$.

With $N = 3$ and a uniform grid, the difference operators D_x^0, D_x^1 and D_x^2 applied to $\exp(-\mathrm{i}\, k\, x)$ are equivalent to the multipliers on the right-hand sides

$$D_x^0[\exp(-\mathrm{i}\, k\, x)]/\exp(-\mathrm{i}\, k\, x) \;=\; 1\,, \tag{3 35a}$$

$$D_x^1[\exp(-\mathrm{i}\, k\, x)]/\exp(-\mathrm{i}\, k\, x) \;=\; -\mathrm{i}\, k \frac{\sin\left(\frac{1}{2}k\,\Delta x\right)\cos\left(\frac{1}{2}k\,\Delta x\right)}{\frac{1}{2}k\,\Delta x}\,, \tag{3 35b}$$

$$D_x^2[\exp(-\mathrm{i}\, k\, x)]/\exp(-\mathrm{i}\, k\, x) \;=\; -k^2 \frac{\sin\left(\frac{1}{2}k\,\Delta x\right)^2}{(\frac{1}{2}k\Delta x)^2}\,. \tag{3 35c}$$

Saw-tooth disturbances with $k\,\Delta x = \pi$ yield $D_x^1 = 0$ with R real and zero phase velocity, whatever the real coefficients $U_p^{\pm}$. For the one-variable KdV test case (3 2) the exact phase velocity is $u(1 - \frac{1}{6}h^2k^2)$. With $N = 3$ the numerical and exact zero phase velocities coincide provided that the grid spacing is chosen:

$$\Delta x = \frac{\pi}{6^{1/2}} h \approx 1.28255h. \tag{3.36}$$

Thus, there is reasonable accuracy in the phase velocity extending well away from $k = 0$. However, the grid spacing (3.36) would be too coarse if the focus of attention was the short-scale left-propagating oscillatory tail (Marchant & Smyth 2002)

For the KdV test case (3.2) the $U_p^{\pm}$ coefficients (3.24a,b) are reasonably simple. The outcome from equation (3.34c) is that the deviation of $|R|^2$ from unity can be factorised:

$$|R|^2 = 1 - \frac{24\, s^2\, \Delta t\, G}{F}, \tag{3.37a}$$

where

$$\begin{aligned} s &= \sin(\tfrac{1}{2}k\,\Delta x) \quad \text{with} \quad 0 \leqslant s^2 \leqslant 1, && (3.37b)\\ F &= \left[3\Delta x^2 - s^2\left(2\Delta x^2 - 2h^2 + u^2\Delta t^2(1-6S) - 6(1+2S)\kappa\,\Delta t\right)\right]^2 \\ &\quad + 9u^2\Delta x^2(1-2S)^2\Delta t^2(1-s^2)s^2 > 0, && (3.37c)\\ G &= 3\kappa\Delta x^2(1-s^2) + s^2(\kappa + u^2\,\Delta t\, S)(12\kappa\,\Delta t\, S + 2h^2 + \Delta x^2 - u^2\,\Delta t^2). && (3.37d) \end{aligned}$$

The non-negativity of the semi-sine-squared s^2 and of the sum of squares F reduce the condition for stability to the condition for non-negativity of G.

The linearity in s^2 of G requires the non-negativity at the two extremities $s^2 = 0$ (long waves) and $s^2 = 1$ (saw-teeth at successive grid points). At $s^2 = 0$ the non-negativity of the diffusivity κ suffices to imply non-negativity of G. At $s^2 = 1$ there are two factors for G, both linear in S. There is stability if both factors have the same sign. For positive signs, the stability condition is that S must satisfy the two inequalities.

$$u^2 S\Delta t \geqslant -\kappa, \tag{3.38a}$$

$$12\kappa\,\Delta t\, S \geqslant u^2\,\Delta t^2 - \Delta x^2 - 2h^2. \tag{3.38b}$$

There is instability should one, but not both, of the inequalities be violated

For the decay-diffusion equation (i.e. $u = 0$, $h = 0$ with $\kappa > 0$) the Crandall (1955) scheme yields $S = \Delta x^2/(12\kappa\,\Delta t) > 0$ With $u = 0$ the positivity of S is sufficient to satisfy both inequalities (3.38a,b) and to guarantee stability, whatever the value of Δx

The simple selection $S = 0$ is stable if $\kappa > 0$ and the time-step is restricted such that:

$$|u|\,\Delta t \leqslant \left(\Delta x^2 + 2h^2\right)^{1/2}. \tag{3.39}$$

This is marginally less stringent than the classical CFL condition.

3.10 Numerical results

The matching of the low to moderate-order derivatives ensure that the scheme gives the best possible results at long length scales. The severest type of numerical test would involve initial conditions at the shortest possible scale.

For a unit delta function starting condition at $x = 0$, $t = 0$ the exact solution of the linear damped KdV equation (3.2) can be written as a convolution in space of the Gaussian ($u = 0$, $h = 0$) and Airy ($\kappa = 0$) similarity solutions

$$\begin{aligned} c(x,t) &= \frac{\exp(-\lambda t)}{(4\pi\kappa t)^{1/2}}\left(\frac{2}{u h^2 t}\right)^{1/3} \\ &\times \int_{-\infty}^{\infty} \exp\left(-\frac{(x-\chi)^2}{4\kappa t}\right) Ai\left(\left(\frac{2}{u h^2 t}\right)^{1/3}(\chi - ut)\right) \mathrm{d}\chi. \end{aligned} \tag{3.40}$$

The Gaussian has strong decay at large distances in both directions. The Airy function has strong decay to the far right, but to the far left there is an increasingly oscillatory but decaying tail. The convolution $c(x,t)$ exhibits oscillations to the far left and non-oscillatory decay to the far right (continuous curves in figures 3.1-3 3). In the advection limit $\kappa = 0$, $h = 0$ the delta function would propagate to $x = ut$.

For the numerical scheme the grid points are taken to be uniformly spaced $x_i = i\,\Delta x$ The unit delta function initial condition is discretised as a kick-start·

$$c_0 = \frac{1}{\Delta x}, \quad c_i = 0 \text{ for } i \neq 0 \quad \text{at } t = 0 \tag{3.41}$$

Linear interpolation would be a ramp from zero at $x = -\Delta x$ rising to $1/\Delta x$ at $x = 0$, then a reversed ramp down to zero at $x = \Delta x$, with composite area unity. The subsequent numerical tests turn out to be more about sensitivity to triangular smoothing of the initial value than about errors from the numerical scheme.

The chosen numerical coefficients, with a non-trivial h are

$$\lambda = 0, \quad u = 1, \quad \kappa = 0\,01, \quad h = 1, \quad \Delta x = 1.28255, \quad \Delta t = 0.75, \quad S = 0\,00640. \tag{3 42}$$

The small κ has been chosen to give predominance to the Airy regime, because the effectiveness of three-point compact schemes in the Gaussian regime is well-established (Crandall 1955, Spotz & Carey 2001). The stability inequalities (3 38a,b) are both satisfied, so the numerical scheme is stable. Zero value $c = 0$ is imposed at distant end points (at $\pm 20\Delta x$).

Figure 3 1 compares the continuous exact solution with the discrete numerical solution at $t = \Delta t$. For $x < 0$ the three-point scheme fails to resolve the sub-grid oscillations with group (or energy) velocity arbitrarily large negative. In the numerical scheme, the choice (3 36) of Δx bounds the negative group velocity by that of the saw-tooth oscillations Those saw-teeth only propagate back to about $-\Delta x$. Triangular smoothing over $(-\Delta x, \Delta x)$ of the exact solution would almost eliminate the sub-grid oscillations and make the numerical scheme look less inadequate for $x < 0$. By contrast, for $x \geqslant 0$ the scheme succeeds in

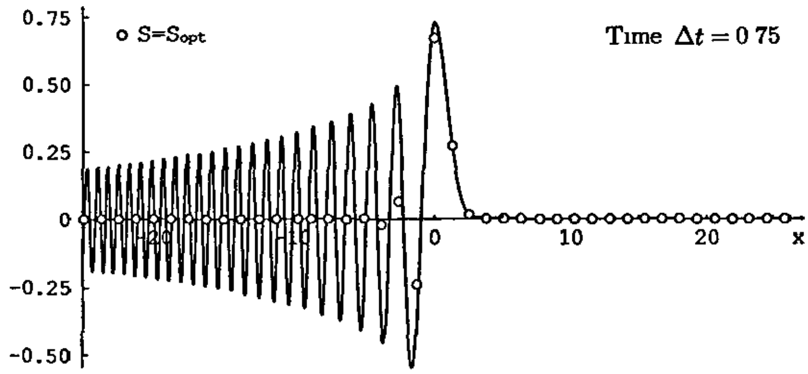

Figure 3 1: At one time-step after the delta-function start, the three-point numerical scheme fails to resolve the sub-grid oscillations that propagate rapidly to the left.

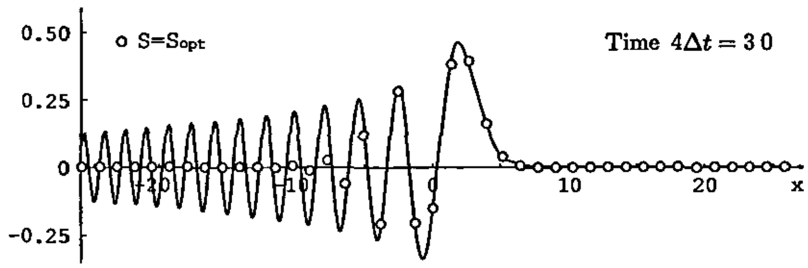

Figure 3 2 At four time-steps after the delta-function start, a few oscillations to the near left are long enough to be resolved and replicated by the three-point numerical scheme.

accurately replicating the height, position and shape of the right-propagating positive surge

Figure 3.2 compares the exact and numerical solutions at $4\Delta t$. To the far left the three-point scheme continues to fail in resolving the sub-grid oscillations. Again, triangular smoothing over $(-\Delta x, \Delta x)$ of the exact solution would almost remove those oscillations and remove the largest errors. The saw-teeth have propagated back to about $-4\,\Delta x$ To the right of figure 3 2, the solution length scale increases and the scheme accuracy improves The first zero-crossing has just advanced right of $x = 0$. The position of the leading peak lags behind the advection prediction $u\,t = 3$. Further to the right the forward skewness has become more apparent.

Figure 3.3 compares the exact and numerical solutions at $16\Delta t$. Now that the short-

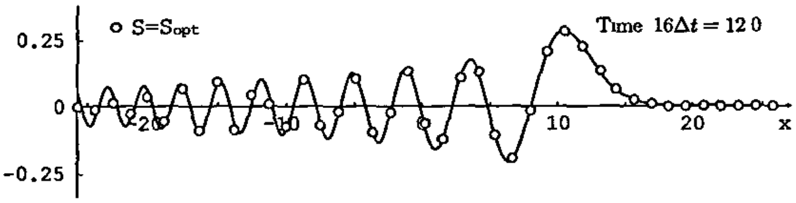

Figure 3 3 By sixteen time-steps after the delta-function start, several oscillations are long enough to be resolved and replicated by the three-point numerical scheme

scale transients have propagated away and the dominant features are longer than the grid spacing, the overall accuracy of the three-point scheme has improved. The saw-teeth and some accuracy have propagated back to about $-12\,\Delta x$. To the right of $x = 0$ there are now three zero-crossings The peak value is near $x = 10$, significantly behind the advection prediction $u\,t = 12$. The forward front remains noticeably skew.

In the context of water-wave surges from the sea into estuaries, the exact phase velocity and the corresponding KdV approximation can be written

$$u\left(\frac{\tanh(kh)}{kh}\right)^{1/2} \approx u\left(1 - \tfrac{1}{6}k^2h^2\right) \quad \text{with} \quad u = (gh)^{1/2}. \tag{3.43}$$

In the context of water waves, if the conversion between dimensionless lengths or depths and metres is multiplication by 10 metres, then the conversion between dimensionless times and seconds is multiplication by 1 01 seconds The numerical coefficients (3.42) would correspond to an estuary of depth 10 metres and diffusive damping of 1 m^2s^{-1}. The horizontal span of the figures would be from -275m to +275m. If the vertical range of the figures were to correspond to the free-surface elevation in metres, then the instantaneous forward displacement at $t = 0$, $x = 0$ would need to have been 1m.

The KdV approximation is only accurate for $kh < 1$. Zero phase velocity water waves have $kh = \infty$ not $kh = 6^{1/2}$. In the water-wave context, the left-propagating short-scale oscillations in figures 3.1-3.3 are shortcomings of the KdV model It is only the oscillation to the right of $x = 0$ that are physically relevant to undular bores.

With $N = 3$, a computational module has too few points in x for the direct numerical representation of the KdV $\partial_x^3 c$ term. The oscillations left of $x = ut$ and the skewness right of $x = ut$ would be absent but for that term While the KdV term cannot be represented with $N = 3$, the effects of the KdV term are modelled.

3.11 Concluding remarks

This chapter gives a straightforward method for the construction of compact schemes. It brings together exact time-stepping (Mitchell & Griffiths 1980) and expansions for the error in difference approximations to derivatives (Bowen & Smith 2005a). For the test case of the linear damped Korteweg-de Vries equation with computational modules spanning only three points in space, the order of truncation and numerical accuracy of the scheme at scales larger than grid spacing go beyond what would usually be expected. The suggestion implicit in this chapter is that scheme construction with accuracy beyond usual expectations should also be possible for compact computational modules of different sizes, for other linear operators, vector dependent variables, non-constant coefficients and several spatial dimensions.

Chapter 4

3D decay-advection-diffusion equation [1]

4.1 Introduction

In the early days of electronic computers, it was a major challenge to design a scheme capable of solving a multi-dimensional partial differential equation. A breakthrough was made by Peaceman & Rachford (1955) and Douglas (1955) with the development of compact alternating direction implicit (ADI) methods for the computation of isotropic diffusion with no flow or decay. Mitchell & Fairweather (1964) optimised the accuracy The methods use two time-levels and a compact computational module with three points in each spatial direction The time-stepping for the N-dimensional solution is factored into N one-dimensional (non-optimal Crank & Nicolson 1947, or optimal Crandall 1955) stages each of which involves solving implicit tri-diagonal systems Those systems can be solved by alternating direction forward and backward sweeps. The number of computations is proportional to $2N$ times the number of grid points. For moderate N, multi-dimensional computations are only difficult because of the large number of points at which the solution is required

Now, half a century later, compact ADI schemes are ideally suited for computation on parallel computers. For each pair of one-dimensional sweeps, there is an $N-1$ dimensional array of computations to be performed. Those numerous computations can be run in serial on a single processor or in parallel on separate processors.

Alas, the linear addition of decay, flow, or off-diagonal diffusivity terms to the partial

[1] Submitted for publication (Bowen & Smith 2005b)

differential equation and of corresponding linear additions to the numerical scheme, can lead to a collapse of accuracy Shortening the steps and increasing the number of grid points can recover accuracy, but sacrifices the speed advantage of ADI schemes.

Restoring the accuracy of ADI schemes restores their competitiveness. Beam & Warming (1978) used the method of approximate (spatial) factorisation to derive a second-order accurate compact ADI scheme, with three points in each spatial direction, for the compressible Navier-Stokes equations An equivalent method was used by McKee, Wall & Wilson (1996) to derive a second order accurate compact ADI scheme for the temperature or concentration distribution in flow with off-diagonal diffusivity. For that problem, Smith & Tang (2001) used Fourier methods to increase the accuracy to third order, but with restrictions to uniform grid spacing and to two dimensions. The increasing non-linearity from low to high order, of the optimal scheme coefficients in equations (2.4b,c,d) of Smith & Tang (2001) explains the inadequacy of linear addition of terms to the numerical scheme

As an alternative to Fourier methods, and without any restriction to uniform spacing, Bowen & Smith (2005a) give derivative expansions for the finite difference counterparts to spatial derivatives The purpose of the present chapter is to exemplify the ease with which derivative expansions lead to optimally accurate ADI schemes. Numerical comparisons with the McKee, Wall & Wilson (1996) scheme and other three-point compact methods, are conducted in serial with a $21 \times 21 \times 21$ grid for decay-advection-diffusion in three dimensions. Relative to non-ADI schemes there is a 20-fold speed up. For the chosen parameter values, there is also a 12-fold accuracy improvement

4.2 Exact free and approximate forced time-stepping

In operator notation, a forced linear evolution equation can be denoted·

$$\partial_t c + \ell\, c(\mathbf{x}, t) = q(\mathbf{x}, t)\,. \tag{4.1}$$

For the chosen illustrative example of the decay-advection-diffusion equation in three dimensions, $c(\mathbf{x},t)$ is the concentration at position $\mathbf{x} = (x_1, x_2, x_3)$ and time t. The linear operator is

$$\ell = \lambda + \mathbf{u} \cdot \nabla - \nabla \cdot \boldsymbol{\kappa} \cdot \nabla^T, \tag{4.2}$$

where λ is the decay rate, $\mathbf{u} = (u_1, u_2, u_3)$ is the flow vector,

$$\boldsymbol{\kappa} = \begin{pmatrix} \kappa_{11} & \kappa_{12} & \kappa_{13} \\ \kappa_{12} & \kappa_{22} & \kappa_{23} \\ \kappa_{13} & \kappa_{23} & \kappa_{33} \end{pmatrix} \tag{4.3}$$

is the symmetric diffusion matrix, and $\nabla = (\partial_{x_1}, \partial_{x_2}, \partial_{x_3})^T$ denotes the derivative column vector For the initial value problem to be well-posed, the diffusion matrix κ_{ij} must be positive definite Throughout this chapter, the subscript $_i$ indicates the spatial direction whereas superscripts ^{n} and $^{n+1}$ refer to time-levels.

For a two time-level compact computational module the reference location $\mathbf{x}_G^n$ (centroid) at which the spatial derivatives ∂_{x_i} are performed at time t^n, need not coincide with the reference location $\mathbf{x}_G^{n+1}$ (centroid) at time $t^{n+1} = t^n + \Delta t$. The vector displacement between those reference locations (centroids) can be used to define a velocity

$$\mathbf{u}_G \, \Delta t = \mathbf{x}_G^{n+1} - \mathbf{x}_G^n. \tag{4.4}$$

For each compact computational module, using a local coordinate system moving at velocity $\mathbf{u}_G$ eliminates the displacement in reference locations Computationally, $\mathbf{u} - \mathbf{u}_G$ replaces $\mathbf{u}$ in the operator ℓ. To avoid lengthening the expressions for scheme coefficients, henceforth where $\mathbf{u}$ is written $\mathbf{u} - \mathbf{u}_G$ is implied.

As explored by Mitchell & Griffiths (1980, chapter 2) and by Cox & Matthews (2002), the exact evolution between time-levels t^n and $t^{n+1} = t^n + \Delta t$ is given by:

$$c^{n+1}(\mathbf{x}) = \exp(-\Delta t\,\ell)\,c^n(\mathbf{x}) + \int_0^{\Delta t} \exp(-(\Delta t - \tau)\ell)\,q(\mathbf{x}, t^n + \tau)\mathrm{d}\tau\,. \tag{4.5}$$

The exponential of an operator is formally defined via Taylor series

$$\exp(-\Delta t\,\ell) = \sum_{n=0}^{\infty} \frac{(-\Delta t)^n}{n!}\ell^n\,, \tag{4.6}$$

and is of infinite order in the derivatives $\partial/\partial x_i$. It leads to exponential non-linearity in $-\lambda\,\Delta t$ and polynomial non-linearity in κ_{ij}, u_i for the numerical scheme. A two time-level interpolation model for the forcing integrand is

$$\exp(-(\Delta t - \tau)\ell)\,q(\mathbf{x}, t^n + \tau) = \left(1 - \frac{\tau}{\Delta t}\right)\exp(-\Delta t\,\ell)\,q^n(\mathbf{x}) + \frac{\tau}{\Delta t}q^{n+1}(\mathbf{x})\,. \tag{4.7}$$

The resulting exact free and approximate forced time-stepping equation is

$$c^{n+1}(\mathbf{x}) - \tfrac{1}{2}\Delta t\,q^{n+1}(\mathbf{x}) = \exp(-\Delta t\,\ell)\left\{c^n(\mathbf{x}) + \tfrac{1}{2}\Delta t\,q^n(\mathbf{x})\right\}\,. \tag{4.8}$$

Time dependence merely requires the replacement of ℓ by the average $\bar{\ell}$ between time-levels t^n and $t^{n+1} = t^n + \Delta t$. For ease of exposition, henceforth the operator ℓ is assumed to be independent of position (see §1.3).

4.3 Factorised spatial discretisation

The essence of ADI schemes is the spatial factorisation (Peaceman & Rachford 1955; Douglas 1955; Mitchell & Fairweather 1964; Beam & Warming 1978; Mitchell & Griffiths 1980, §2.12; McKee, Wall & Wilson 1996). For N arbitrary constant coefficient differential operators $M_i = I + \Delta t \sum m_{i,p}\partial^p_{x_i}$, with I the identity operator, and with ℓ specified by equation (4.2), the exact free and approximate forced time-stepping (4 8) is equivalent to

$$
\begin{aligned}
&\prod_{i=1}^{N} M_i \exp\left(\left[u_i\partial_{x_i} - \kappa_{ii}\partial^2_{x_i}\right]\tfrac{1}{2}\Delta t\right)\left\{c^{n+1}(\mathbf{x}) - \tfrac{1}{2}\Delta t\, q^{n+1}(\mathbf{x})\right\} \\
= \;& \exp(-\lambda\,\Delta t)\left\{\prod_{i=1}^{N} M_i \exp\left(\left[-u_i\partial_{x_i} + \kappa_{ii}\partial^2_{x_i}\right]\tfrac{1}{2}\Delta t\right)\right. \\
& + \prod_{i=1}^{N} M_i \exp\left(\left[-u_i\partial_{x_i} + \kappa_{ii}\partial^2_{x_i}\right]\tfrac{1}{2}\Delta t\right) \\
& \left.\times\left[\exp\left(2\Delta t \sum_{j=1}^{N}\sum_{k>j}^{N} \kappa_{jk}\partial_{x_j}\partial_{x_k}\right) - 1\right]\right\}\left\{c^{n}(\mathbf{x}) + \tfrac{1}{2}\Delta t\, q^{n}(\mathbf{x})\right\}. \qquad (4\,9)
\end{aligned}
$$

In the one-dimensional case, the off-diagonal diffusivity term would be absent and the product of exponentials would be restricted to a single x_i exponential. The product $\prod M_i$ is a (non-normalised) projection of the exact time-stepping

A general template for a derivative expansion is

$$
\begin{aligned}
&\prod_{i=1}^{N}\left(1 + \tfrac{1}{2}\Delta t \sum_{p=1} U^{+}_{i,p}\partial^{p}_{x_i}\right)\left\{c^{n+1}(\mathbf{x}) - \tfrac{1}{2}\Delta t\, q^{n+1}(\mathbf{x})\right\} \\
= \;& \exp(-\lambda\,\Delta t)\left\{\prod_{i=1}^{N}\left(1 - \tfrac{1}{2}\Delta t \sum_{p=1} U^{-}_{i,p}\partial^{p}_{x_i}\right)\right. \\
& \left. + 2\Delta t \sum_{p_1=0}\cdots\sum_{p_N=0} U(p_1,\ldots,p_N)\partial^{p_1}_{x_1}\ldots\partial^{p_N}_{x_N}\right\}\left\{c^{n}(\mathbf{x}) + \tfrac{1}{2}\Delta t\, q^{n}(\mathbf{x})\right\}. \qquad (4\,10)
\end{aligned}
$$

The decay term is exponential in $-\lambda\,\Delta t$. The single-direction coefficients $U^{\pm}_{i,p}$ are polynomial in u_i and in diagonal κ_{ii} but linear in the adjustable constants $m_{i,p}$

$$
\begin{aligned}
U^{\pm}_{i,1} &= u_i \pm 2m_{i,1}\,, && (4.11a)\\
U^{\pm}_{i,2} &= -\kappa_{ii} \pm \tfrac{1}{4}u_i^2\Delta t + m_{i,1}u_i\Delta t \pm 2m_{i,2}\,, && (4.11b)\\
U^{\pm}_{i,3} &= \mp\tfrac{1}{2}u_i\kappa_{ii}\Delta t + \tfrac{1}{24}u_i^3\Delta t^2 + m_{i,1}\Delta t\left(-\kappa_{ii} \pm \tfrac{1}{4}u_i^2\Delta t\right) + m_{i,2}u_i\Delta t \pm 2m_{i,3}\,, && (4.11c)\\
U^{\pm}_{i,4} &= \pm\tfrac{1}{4}\kappa_{ii}^2\Delta t - \tfrac{1}{8}u_i^2\kappa_{ii}\Delta t^2 \pm \tfrac{1}{192}u_i^4\Delta t^3 + m_{i,1}\,u_i\Delta t^2\left(\tfrac{1}{24}u_i^2\Delta t \mp \tfrac{1}{2}\kappa_{ii}\right)\\
&\quad + m_{i,2}\Delta t\left(-\kappa_{ii} \pm \tfrac{1}{4}u_i^2\Delta t\right) + m_{i,3}u_i\Delta t \pm 2m_{i,4}\,. && (4.11d)
\end{aligned}
$$

In §4.5 it is shown that to the requisite accuracy $U(p_1, \ldots, p_N)$ have elementary expressions involving $U^-_{i,1}$, $U^-_{i,2}$ and off-diagonal κ_{ij}

For compact finite differences with P points in all coordinate directions, there are finite difference counterparts D^p_i to derivatives $\partial^p_{x_i}$ at the reference location $\mathbf{x}^n_G$ with $0 \leqslant p \leqslant P-1$ (Fornberg 1988, Corless & Rokicki 1996). A compact ADI finite difference counterpart to the derivative expansion (4.10) is simply

$$
\begin{aligned}
&\prod_{i=1}^{N}\left(D^0_i + \tfrac{1}{2}\Delta t\sum_{p=1}^{P-1}U^+_{i,p}D^p_i\right)\left\{C^{n+1}(\mathbf{x}) - \tfrac{1}{2}\Delta t\,Q^{n+1}(\mathbf{x})\right\}\\
&= \exp(-\lambda\,\Delta t)\left\{\prod_{i=1}^{N}\left(D^0_i - \tfrac{1}{2}\Delta t\sum_{p=1}^{P-1}U^-_{i,p}D^p_i\right)\right.\\
&\quad\left. + 2\Delta t\sum_{p_1+\cdots+p_N=2}^{p_1+\cdots+p_N=P} U(p_1, \ldots, p_N)D_1^{p_1}\ldots D_N^{p_N}\right\}\left\{C^n(\mathbf{x}) + \tfrac{1}{2}\Delta t\,Q^n(\mathbf{x})\right\}. \qquad (4.12)
\end{aligned}
$$

Upper-case quantities $C^n(\mathbf{x})$, $Q^n(\mathbf{x})$ are used to distinguish the computed discrete numerical values from the lower-case continuous variables $c^n(\mathbf{x})$, $q^n(\mathbf{x})$ On the last line, the notation indicates that the summation over $p_1, \ldots, p_N$ is restricted to total derivative order up to P.

The accuracy of the scheme relates to the magnitude of the errors The next section illustrates that the absence of counterparts to $U^{\pm}_{i,P}\partial^P_x \ldots U^{\pm}_{i,2P-2}\partial^{2P-2}_x$ from the scheme (4.12) can be rectified with the selection of $m_{i,1} \ldots m_{i,2P-2}$

4.4 Three-point difference approximations to derivatives

For three points, the i-coordinates for the grid and reference points are denoted x_i^-, x_i, x_i^+ and χ_i. Three-point difference operators from appendix A that approximate the identity, first derivative and second derivative at χ_i are.

$$\begin{aligned}
D_i^0[C] &= \frac{(x_i-\chi_i)(x_i^+-\chi_i)C\left(x_i^-\right)}{\left(x_i^- - x_i\right)\left(x_i^- - x_i^+\right)} + \frac{(x_i^- -\chi_i)(x_i^+-\chi_i)C\left(x_i\right)}{\left(x_i - x_i^-\right)\left(x_i - x_i^+\right)} \\
&\quad + \frac{(x_i^- -\chi_i)(x_i-\chi_i)C\left(x_i^+\right)}{\left(x_i^+ - x_i^-\right)\left(x_i^+ - x_i\right)}, && (4.13\text{a}) \\
D_i^1[C] &= -\frac{\left(x_i+x_i^+-2\chi_i\right)C\left(x_i^-\right)}{\left(x_i^- - x_i\right)\left(x_i^- - x_i^+\right)} - \frac{\left(x_i^-+x_i^+-2\chi_i\right)C\left(x_i\right)}{\left(x_i - x_i^-\right)\left(x_i - x_i^+\right)} \\
&\quad - \frac{\left(x_i^-+x_i-2\chi_i\right)C\left(x_i^+\right)}{\left(x_i^+ - x_i^-\right)\left(x_i^+ - x_i\right)}, && (4.13\text{b}) \\
D_i^2[C] &= \frac{2C\left(x_i^-\right)}{\left(x_i^- - x_i\right)\left(x_i^- - x_i^+\right)} + \frac{2C\left(x_i\right)}{\left(x_i - x_i^-\right)\left(x_i - x_i^+\right)} \\
&\quad + \frac{2C\left(x_i^+\right)}{\left(x_i^+ - x_i^-\right)\left(x_i^+ - x_i\right)}. && (4.13\text{c})
\end{aligned}$$

The optimal scheme cannot depend upon χ_i. However, there is χ_i-dependence in the way that scheme is represented in terms of D_i^p.

Chapter 2 derives derivative expansions for the errors in terms of the P elementary symmetric functions in the displacements. For three points the displacements are denoted $\alpha_i^* = x_i^* - \chi_i$, with * denoting $^+$, null or $^-$. The linear, quadratic and cubic elementary symmetric functions are:

$$e_{i,1} = \alpha_i^- + \alpha_i + \alpha_i^+,\ e_{i,2} = \alpha_i^-\alpha_i + \alpha_i^-\alpha_i^+ + \alpha_i\alpha_i^+,\ e_{i,3} = \alpha_i^-\alpha_i\alpha_i^+ . \qquad (4.14)$$

In the error expansions in appendix B, $e_{i,1}$ occurs more frequently than the higher degree elementary symmetric functions To set $e_{i,1}=0$ and achieve the consequent simplifications, it is henceforth assumed that for each three-point computational module, the reference point is the centroid i e $\chi_i = \frac{1}{3}(x_i^- + x_i + x_i^+)$. With this assumption, $e_{i,2}$ is strictly negative and can be interpreted as minus the effective mean-square spacing. For regular spacing Δx, the centroid is $\chi_i = x_i$ and the elementary symmetric functions are $e_{i,2} = -\Delta x^2$ and $e_{i,3} = 0$

With the χ_i at the centroid, the derivative expansions from appendix B become

$$D_i^0 = I + \tfrac{1}{6}e_{i,3}\,\partial^3_{x_i} - \tfrac{1}{120}e_{i,2}e_{i,3}\,\partial^5_{x_i} + \dots , \tag{4 15a}$$

$$D_i^1 = \partial_{x_i} - \tfrac{1}{6}e_{i,2}\,\partial^3_{x_i} + \tfrac{1}{24}e_{i,3}\,\partial^4_{x_i} + \tfrac{1}{120}e^2_{i,2}\,\partial^5_{x_i} + \dots , \tag{4 15b}$$

$$D_i^2 = \partial^2_{x_i} - \tfrac{1}{12}e_{i,2}\,\partial^4_{x_i} + \tfrac{1}{60}e_{i,3}\,\partial^5_{x_i} + \dots \tag{4 15c}$$

Hence, term-by-term compact modelling of the partial differential equation would give spatial errors of third order. For D_i^0 the derivative order of error terms is the same as their polynomial power in the displacements This derivative and power dual meaning of 'order' transfers to the grid point accuracy of the numerical scheme.

The one-dimensional matching at order $\partial^3_{x_i}$ of the derivative expansion (4 10) to the difference scheme (4.12), yields the ${}^{n+1}$ and ^{n} pair of matching conditions

$$\tfrac{1}{2}\Delta t\, U^+_{i,3} = \tfrac{1}{6}e^{n+1}_{i,3} - \tfrac{1}{12}\Delta t\, U^+_{i,1}e^{n+1}_{i,2}\,, \tag{4.16a}$$

$$-\tfrac{1}{2}\Delta t\, U^-_{i,3} = \tfrac{1}{6}e^{n}_{i,3} + \tfrac{1}{12}\Delta t\, U^-_{i,1}e^{n}_{i,2}\,. \tag{4 16b}$$

At order $\partial^4_{x_i}$, the derivative and difference matching shifts one term along to·

$$\tfrac{1}{2}\Delta t\, U^+_{i,4} = \tfrac{1}{48}\Delta t\, U^+_{i,1}e^{n+1}_{i,3} - \tfrac{1}{24}\Delta t\, U^+_{i,2}e^{n+1}_{i,2}\,, \tag{4 16c}$$

$$-\tfrac{1}{2}\Delta t\, U^-_{i,4} = -\tfrac{1}{48}\Delta t\, U^-_{i,1}e^{n}_{i,3} + \tfrac{1}{24}\Delta t\, U^-_{i,2}e^{n}_{i,2}\,. \tag{4 16d}$$

The matching (4 16a-d) involves the elementary symmetric functions at two time-levels It is convenient to define time averages and semi-differences,

$$\bar{e}_{i,*} = \frac{e^{n+1}_{i,*} + e^{n}_{i,*}}{2}, \quad e'_{i,*} = \frac{e^{n+1}_{i,*} - e^{n}_{i,*}}{2}, \tag{4 17}$$

where $*$ denotes 1, 2 or 3. On a fixed grid $\bar{e}_{i,*} = e_{i,*}$, $e'_{i,*} = 0$

The formulae (4.11a-d) linking $U^{\pm}_{i,p}$ to $m_{i,p}$, convert the third-order matching (4.16a,b) to a pair of linear equations in $m_{i,2}$ and $m_{i,3}$. The solution for $m_{i,2}$,

$$m_{i,2} = -\tfrac{1}{24}u_i^2\Delta t - \frac{\bar{e}_{i,2}}{6\,\Delta t} + \frac{e'_{i,3} + m_{i,1}\,\Delta t\,(3\kappa_{ii}\,\Delta t - e'_{i,2})}{3\,u_i\Delta t^2}\,, \tag{4.18}$$

becomes singular in the pure-diffusion limit as u_i tends to zero. This failure can be rectified if $m_{i,1}$ is restricted to the one-parameter family

$$m_{i,1} = -F_i - S_i u_i\,, \tag{4.19a}$$

where S_i is an adjustable constant (possibly zero) and

$$F_i = \frac{e'_{i,3}}{\Delta t\left(3\kappa_{ii}\Delta t - e'_{i,2}\right)}\,. \tag{4.19b}$$

On a fixed grid F_i is zero

With the restricted structure (4.19a,b) for $m_{i,1}$, there is not a singularity in $m_{i,2}$. For arbitrary S_i, the third-order scheme coefficients are given by.

$$U^{\pm}_{i,1} = u_i \mp 2\,(F_i + S_i u_i)\,, \tag{4.20a}$$

$$U^{\pm}_{i,2} = -\kappa_{ii} - u_i F_i \Delta t - S_i u_i^2 \Delta t \pm \tfrac{1}{6}u_i^2\Delta t \mp \frac{\bar{e}_{i,2}}{3\,\Delta t} \mp 2\,S_i\left(\kappa_{ii} - \frac{e'_{i,2}}{3\,\Delta t}\right)\,. \tag{4.20b}$$

The third-order matching also determines $m_{i,3}$, $U^{\pm}_{i,3}$ but these are not needed directly in the finite difference scheme nor in the evaluation of the mixed-direction coefficients $U(p_1, \ldots, p_N)$ as performed in §4.5.

Fourth-order matching (4.16c,d) gives the optimal value of the parameter S_i:

$$S_{opt} = \frac{-\kappa_{ii}\left(2u_i^2\Delta t^2 + \bar{e}_{i,2}\right) - \frac{3}{2}\bar{e}_{i,3}u_i + f_i}{\Delta t\left(12\kappa_{ii}^2 + \bar{e}_{i,2}u_i^2 + u_i^4\Delta t^2\right) + g_i}\,, \tag{4.21a}$$

where

$$f_i = \left(\tfrac{5}{6}u_i^2\Delta t + \frac{\bar{e}_{i,2}}{3\,\Delta t}\right)e'_{i,2} - \left(u_i^3\Delta t^3 + \bar{e}_{i,2}u_i\Delta t + e'_{i,3}\right)F_i\,, \tag{4.21b}$$

$$g_i = \frac{2\,e'^2_{i,2}}{3\,\Delta t} + u_i e'_{i,3} - 6\kappa_{ii}e'_{i,2}\,. \tag{4.21c}$$

On a fixed grid f_i, g_i are both zero. With this optimal choice for S, the scheme is referred to as the $S = S_{opt}$ scheme. With a non-optimal choice $S_i = 0$, the scheme is referred to as the $S = S_0$ scheme. There is striking non-linearity of S_{opt} in u_i, κ_{ii} and $\bar{e}_{i,2}$. Fourth-order matching also determines $m_{i,4}$, $U^{\pm}_{i,4}$ but these are not needed. Smith (2000) gave a Fourier derivation of the results (4.20a,b, 4.21a-c).

For uniform spacing and zero flow, there is inverse dependence on κ_{ii},

$$S_{opt} = \frac{\Delta x^2}{12\,\kappa_{ii}\Delta t}\,, \tag{4.22}$$

and the scheme coefficients (4.20a,b, 4.21a-c) give the one-dimensional optimal scheme of Crandall (1955) or the Mitchell & Fairweather (1964) optimisation of the Peaceman & Rachford (1955) and Douglas (1955) ADI schemes.

4.5 Mixed-direction coefficients

The mixed-direction coefficients $U(p_1, \ldots, p_N)$ with $p_i \leqslant 2$ are sought, such that

$$\begin{aligned}
&\sum_{p_1+\cdots+p_N=2}^{p_1+\cdots+p_N=3} U(p_1,\ldots,p_N)\partial_{x_1}^{p_1}\ldots\partial_{x_N}^{p_N} + \text{fourth and higher order derivatives} \\
&= \frac{1}{2\,\Delta t}\prod_{i=1}^{N}\left\{1 + \Delta t\sum m_{i,p}\partial_{x_i}^{p}\right\}\exp\left(\left[-u_i\partial_{x_i} + \kappa_{ii}\partial_{x_i}^2\right]\tfrac{1}{2}\Delta t\right) \\
&\quad\times\left[\exp\left(2\Delta t\sum_{j=1}^{N}\sum_{k>j}^{N}\kappa_{jk}\partial_{x_j}\partial_{x_k}\right) - 1\right].
\end{aligned} \tag{4.23a}$$

The low level of truncation permits the replacement of the M_i-exponential products by low-order derivatives, and the expansion of the off-diagonal exponential

$$\approx \prod_{i=1}^{N}\left(1-\tfrac{1}{2}\Delta t\, U_{i,1}^{-}\partial_{x_i}-\tfrac{1}{2}\Delta t\, U_{i,2}^{-}\partial_{x_i}^{2}\right)\sum_{j=1}^{N}\sum_{k>j}^{N}\kappa_{jk}\partial_{x_j}\partial_{x_k}\,. \tag{4.23b}$$

The expansion can be written

$$\begin{aligned}\approx\ & \sum_{j=1}^{N}\sum_{k>j}^{N}\kappa_{jk}\left(\partial_{x_j}-\tfrac{1}{2}\Delta t\, U_{j,1}^{-}\partial_{x_j}^{2}\right)\left(\partial_{x_k}-\tfrac{1}{2}\Delta t\, U_{k,1}^{-}\partial_{x_k}^{2}\right)\\ & -\tfrac{1}{2}\Delta t\sum_{j=1}^{N}\sum_{k>j}^{N}\sum_{i\neq j,k}^{N}\kappa_{jk}U_{i,1}^{-}\partial_{x_i}\partial_{x_j}\partial_{x_k}\\ & +\text{ fourth and higher order derivatives}\,.\end{aligned} \tag{4.23c}$$

Corresponding to (4.23c), the $N=3$ compact ADI scheme can be written

$$\begin{aligned}&\prod_{i=1}^{3}\left(D_i^0+\tfrac{1}{2}\Delta t\, U_{i,1}^{+}D_i^1+\tfrac{1}{2}\Delta t\, U_{i,2}^{+}D_i^2\right)\left\{C^{n+1}(\mathrm{x})-\tfrac{1}{2}\Delta t\, Q^{n+1}(\mathrm{x})\right\}\\ =\ & \exp(-\lambda\Delta t)\left\{\prod_{i=1}^{3}\left(D_i^0-\tfrac{1}{2}\Delta t\, U_{i,1}^{-}D_i^1-\tfrac{1}{2}\Delta t\, U_{i,2}^{-}D_i^2\right)\right.\\ & +2\Delta t\left[\kappa_{12}\left(D_1^1-\tfrac{1}{2}\Delta t\, U_{1,1}^{-}D_1^2\right)\left(D_2^1-\tfrac{1}{2}\Delta t\, U_{2,1}^{-}D_2^2\right)D_3^0\right.\\ & \quad+\kappa_{13}\left(D_1^1-\tfrac{1}{2}\Delta t\, U_{1,1}^{-}D_1^2\right)D_2^0\left(D_3^1-\tfrac{1}{2}\Delta t\, U_{3,1}^{-}D_3^2\right)\\ & \quad+\kappa_{23}D_1^0\left(D_2^1-\tfrac{1}{2}\Delta t\, U_{2,1}^{-}D_2^2\right)\left(D_3^1-\tfrac{1}{2}\Delta t\, U_{3,1}^{-}D_3^2\right)\\ & \quad\left.\left.-\tfrac{1}{2}\Delta t\left(\kappa_{12}U_{3,1}^{-}+\kappa_{13}U_{2,1}^{-}+\kappa_{23}U_{1,1}^{-}\right)D_1^1D_2^1D_3^1\right]\right\}\\ & \times\left\{C^n(\mathrm{x})+\tfrac{1}{2}\Delta t\, Q^n(\mathrm{x})\right\}\,.\end{aligned} \tag{4.24}$$

Formally, the errors are of fourth order with S_i arbitrary

The $N = 3$ McKee, Wall & Wilson (1996) scheme, with decay included, is

$$\begin{aligned}
&\prod_{i=1}^{3}\left(D_i^0 + \tfrac{1}{2}\Delta t\, u_i D_i^1 - \tfrac{1}{2}\Delta t\, \kappa_{ii} D_i^2\right)\left\{C^{n+1}(\mathrm{x}) - \tfrac{1}{2}\Delta t\, Q^{n+1}(\mathrm{x})\right\} \\
= \;& \exp\left(-\lambda\Delta t\right)\left\{\prod_{i=1}^{3}\left(D_i^0 - \tfrac{1}{2}\Delta t\, u_i D_i^1 + \tfrac{1}{2}\Delta t\, \kappa_{ii} D_i^2\right)\right. \\
&\left. +2\,\Delta t\,\left[\kappa_{12} D_1^1 D_2^1 D_3^0 + \kappa_{13} D_1^1 D_2^0 D_3^1 + \kappa_{23} D_1^0 D_2^1 D_3^1\right]\right\} \\
&\times\left\{C^n(\mathrm{x}) + \tfrac{1}{2}\Delta t\, Q^n(\mathrm{x})\right\}. \qquad (4.25)
\end{aligned}$$

The simplicity of the coefficients, as compared with the optimal ADI scheme (4.24), comes with a loss of accuracy that is quantified in §4.8.

4.6 ADI solution

The right-hand side of the scheme (4.24) consists of known values from the ^{n} time-step. As elaborated by Mitchell & Griffiths (1980, §2.12), the factorised structure of the left-hand side of the scheme (4.24) allows for fast solution, by solving sets of tri-diagonal systems The scheme is solved in three alternating-direction implicit (ADI) stages. Assuming n_i points along each dimension, there is a total of $n_1 n_2 + n_1 n_3 + n_2 n_3$ tri-diagonal systems to be solved, either in serial or parallel.

For definiteness, the x_1 sweeps are performed first. The quasi-concentration $\widetilde{C}^{n+1}$, associated with the central grid point of the computational module, is the solution over all the grid points of the $n_2 \times n_3$ tri-diagonal systems:

$$
\begin{aligned}
&\left(D_1^0 + \tfrac{1}{2}\Delta t\, U_{1,1}^+ D_1^1 + \tfrac{1}{2}\Delta t\, U_{1,2}^+ D_1^2\right) \widetilde{C}^{n+1} \\
=\; & \exp(-\lambda\Delta t) \left\{ \prod_{i=1}^{3} \left(D_i^0 - \tfrac{1}{2}\Delta t\, U_{i,1}^- D_i^1 - \tfrac{1}{2}\Delta t\, U_{i,2}^- D_i^2\right) \right. \\
& +2\,\Delta t \left[\kappa_{12} \left(D_1^1 - \tfrac{1}{2}\Delta t\, U_{1,1}^- D_1^2\right) \left(D_2^1 - \tfrac{1}{2}\Delta t\, U_{2,1}^- D_2^2\right) D_3^0 \right. \\
& \quad +\kappa_{13} \left(D_1^1 - \tfrac{1}{2}\Delta t\, U_{1,1}^- D_1^2\right) D_2^0 \left(D_3^1 - \tfrac{1}{2}\Delta t\, U_{3,1}^- D_3^2\right) \\
& \quad +\kappa_{23} D_1^0 \left(D_2^1 - \tfrac{1}{2}\Delta t\, U_{2,1}^- D_2^2\right) \left(D_3^1 - \tfrac{1}{2}\Delta t\, U_{3,1}^- D_3^2\right) \\
& \quad \left.\left. -\tfrac{1}{2}\Delta t \left(\kappa_{12} U_{3,1}^- + \kappa_{13} U_{2,1}^- + \kappa_{23} U_{1,1}^-\right) D_1^1 D_2^1 D_3^1 \right] \right\} \\
& \times \left\{ C^n(\mathbf{x}) + \tfrac{1}{2}\Delta t\, Q^n(\mathbf{x}) \right\} . \qquad (4\,26a)
\end{aligned}
$$

For definiteness, the x_2 sweeps are performed next. Another quasi-concentration $\widehat{C}^{n+1}$, associated with the central grid point of the computational module, is the solution over all the grid points of the $n_1 \times n_3$ tri-diagonal systems:

$$
\left(D_2^0 + \tfrac{1}{2}\Delta t\, U_{2,1}^+ D_2^1 + \tfrac{1}{2}\Delta t\, U_{2,2}^+ D_2^2\right) \widehat{C}^{n+1} = \widetilde{C}^{n+1} . \qquad (4.26b)
$$

Finally, the x_3 sweeps give the actual concentration C^{n+1}, by solving the $n_1 \times n_2$ tri-diagonal systems:

$$
\left(D_3^0 + \tfrac{1}{2}\Delta t\, U_{3,1}^+ D_3^1 + \tfrac{1}{2}\Delta t\, U_{3,2}^+ D_3^2\right) \left\{C^{n+1}(\mathbf{x}) - \tfrac{1}{2}\Delta t\, Q^{n+1}(\mathbf{x})\right\} = \widehat{C}^{n+1} . \qquad (4\,26c)
$$

Tri-diagonal systems can be solved very quickly using standard methods (Mitchell & Griffiths 1980, §2 5; Richtmyer & Morton 1967, §8 5). The structure ensures no coupling between systems. At each of the N stages, the left-hand side operates in just one dimension Thus, the systems can be solved in parallel and split amongst processors. To advance the solution by a time-step, the computational running time is proportional to $2\,N$ times the total number of points, and is inversely proportional to the number of processors involved.

4.7 Stability conditions

With a fixed grid and zero off-diagonal diffusion, the growth factor for the 3D case is of the form (1.32) and so the stability calculation involves finding sufficient conditions such that the inequality (1.34) is satisfied. For this case (1.34) is a lengthy expression of the form (C 6), and hence the inequalities (C 8) are applied to split the solution into smaller parts. After removing constant factors and factors that are non-negative by definition (such as κ_{ii}, Δx_i and Δt), the inequalities (C.8) for the 3D case become·

$$
\begin{aligned}
&3 : \alpha_1 \geqslant 0, \quad 7 : \alpha_2 \geqslant 0, \quad 19 : \alpha_3 \geqslant 0,\\
&5 : \kappa_{11}\beta_2 + \kappa_{22}\beta_1 \geqslant 0, \qquad 11 : \kappa_{11}\beta_3 + \kappa_{33}\beta_1 \geqslant 0, \qquad 13 : \kappa_{22}\beta_3 + \kappa_{33}\beta_2 \geqslant 0,\\
&6 : \alpha_1\beta_2 + \kappa_{22}\gamma_1 \geqslant 0, \qquad 8 : \alpha_2\beta_1 + \kappa_{11}\gamma_2 \geqslant 0, \qquad 12 : \alpha_1\beta_3 + \kappa_{33}\gamma_1 \geqslant 0,\\
&16 : \alpha_2\beta_3 + \kappa_{33}\gamma_2 \geqslant 0, \qquad 20 : \alpha_3\beta_1 + \kappa_{11}\gamma_3 \geqslant 0, \qquad 22 : \alpha_3\beta_2 + \kappa_{22}\gamma_3 \geqslant 0,\\
&9 : \alpha_1\gamma_2 + \alpha_2\gamma_1 \geqslant 0, \qquad 21 : \alpha_1\gamma_3 + \alpha_3\gamma_1 \geqslant 0, \qquad 25 : \alpha_2\gamma_3 + \alpha_3\gamma_2 \geqslant 0,\\
&14 : \kappa_{11}\beta_2\beta_3 + \kappa_{22}\beta_1\beta_3 + \kappa_{33}\beta_1\beta_2 + 144\Delta t^2\kappa_{11}\kappa_{22}\kappa_{33} \geqslant 0,\\
&15 : \gamma_1\left(\kappa_{22}\beta_3 + \kappa_{33}\beta_2\right) + \alpha_1\left(\beta_2\beta_3 + 144\Delta t^2\kappa_{22}\kappa_{33}\right) \geqslant 0,\\
&17 : \gamma_2\left(\kappa_{11}\beta_3 + \kappa_{33}\beta_1\right) + \alpha_2\left(\beta_1\beta_3 + 144\Delta t^2\kappa_{11}\kappa_{33}\right) \geqslant 0,\\
&23 : \gamma_3\left(\kappa_{11}\beta_2 + \kappa_{22}\beta_1\right) + \alpha_3\left(\beta_1\beta_2 + 144\Delta t^2\kappa_{11}\kappa_{22}\right) \geqslant 0,\\
&18 : \beta_3\left(\alpha_1\gamma_2 + \alpha_2\gamma_1\right) + \kappa_{33}\left(\gamma_1\gamma_2 + 144\Delta t^2\alpha_1\alpha_2\right) \geqslant 0,\\
&24 : \beta_2\left(\alpha_1\gamma_3 + \alpha_3\gamma_1\right) + \kappa_{22}\left(\gamma_1\gamma_3 + 144\Delta t^2\alpha_1\alpha_3\right) \geqslant 0,\\
&26 : \beta_1\left(\alpha_2\gamma_3 + \alpha_3\gamma_2\right) + \kappa_{11}\left(\gamma_2\gamma_3 + 144\Delta t^2\alpha_2\alpha_3\right) \geqslant 0,\\
&27 : \alpha_1\gamma_2\gamma_3 + \alpha_2\gamma_1\gamma_3 + \alpha_3\gamma_1\gamma_2 + 144\Delta t^2\alpha_1\alpha_2\alpha_3 \geqslant 0.
\end{aligned} \tag{4.27}
$$

Inequalities 1, 2, 4 and 10 are immediately satisfied and have been omitted The inequalities (4 27) are satisfied if the following quantities are non-negative.

$$\begin{aligned}
\alpha_i &= \left(S_i u_i^2 \Delta t + \kappa_{ii}\right)\left(12 S_i \kappa_{ii} \Delta t + \Delta x_i^2 - u_i^2 \Delta t^2\right), && (4\ 28a)\\
\beta_i &= 12 S_i^2 u_i^2 \Delta t^2 + 24 S_i \kappa_{ii} \Delta t + 2\Delta x_i^2 + u_i^2 \Delta t^2, && (4\ 28b)\\
\gamma_i &= 12\Delta t\left(3S_i^2\left(u_i^4 \Delta t^3 + 4\kappa_{ii}^2 \Delta t\right) + 2S_i\kappa_{ii}\left(2u_i^2\Delta t^2 + \Delta x_i^2\right) + 3\kappa_{ii}^2 \Delta t\right)\\
&\quad + \left(\Delta x_i^2 - u_i^2 \Delta t^2\right)^2 . && (4\ 28c)
\end{aligned}$$

In the absence of off-diagonal diffusion terms, the conditions for numerical stability are inherited from the N one-dimensional cases. In particular, for the 3D case with uniform spacing, (4 28a-c) show that for each direction, $S_i \geqslant 0$ together with the classic CFL (Courant-Friedrichs-Lewy) condition $|u_i|\Delta t \leqslant \Delta x_i$ are sufficient conditions for stability For zero flow and uniform spacing, the well-posed requirement that the diffusion matrix κ_{ij} be positive definite is also a sufficient condition for numerical stability (McKee & Mitchell 1970, Smith & Tang 2001).

4.8 Numerical results

To compare the $S = S_{opt}$ and $S = S_0$ schemes to other schemes, a restriction is made to uniformly spaced grids with $D_i^0 = 1$. Standard error norms are introduced

$$l_1 = \frac{\sum^p |C^n - c(t^n)|}{\sum^p |c(t^n)|}, \quad l_2 = \frac{\left(\sum^p (C^n - c(t^n))^2\right)^{\frac{1}{2}}}{\left(\sum^p c(t^n)^2\right)^{\frac{1}{2}}}, \quad l_\infty = \frac{max\,|C^n - c(t^n)|}{max\,|c(t^n)|}, \qquad (4\ 29)$$

where $p = n_1 n_2 n_3$ denotes the total number of grid points and summation or maximum is over all of the grid points Error norms near zero are desirable and above unity are extremely bad.

For uniform spacing, the $S = S_{opt}$ and $S = S_0$ schemes and the McKee, Wall & Wilson

(1996) scheme in (4 25) are tested against a 3D forward-time θ-method time-averaged spatial derivatives scheme, written in difference operator notation as

$$\frac{C^{n+1}-\mathrm{e}^{-\lambda\Delta t}C^{n}}{\Delta t}+\sum_{i=1}^{3}\Big[u_i\left\{\mathrm{e}^{-\lambda\Delta t}\left(1-\theta\right)D_i^1[C^n]+\theta D_i^1[C^{n+1}]\right\}$$
$$-\kappa_{ii}\left\{\mathrm{e}^{-\lambda\Delta t}\left(1-\theta\right)D_i^2[C^n]+\theta D_i^2[C^{n+1}]\right\}\Big]=0\,. \qquad (4.30)$$

The values used are $\theta = 0$ (explicit), $\theta = \frac{1}{2}$ (Crank-Nicolson) and $\theta = 1$ (fully implicit). For large grids the Crank & Nicolson (1947) and fully implicit schemes, as written, are unsuitable for general use due to the matrix system that has to be solved. Here, the modest number of grid points along with Mathematica's sparse array routines make the run-time of these comparisons bearable. Methods to directly convert the $\theta-$methods into a faster ADI structure would only reduce their accuracy yet further

Point source tests are used due to the severe strain they cause numerical schemes. An initial point source of unit strength is placed in the centre grid point and left to advect and diffuse The boundary values are held at zero

The first test is of pure diffusion with the parameters (making all schemes stable):

$$\Delta t = 0\,2\,,\quad \Delta x = 1\,,\quad \lambda = 0\,,\quad u = v = w = 0\,,$$
$$\kappa_{11} = \kappa_{22} = \kappa_{33} = 0.8\,,\quad \kappa_{12} = \kappa_{13} = \kappa_{23} = 0\,. \qquad (4\,31a)$$

The grid size is $21 \times 21 \times 21$ i e 9261 points The error norms are shown in table 4 1. The $\theta = 0$ explicit scheme is always less accurate than the other schemes. For zero flow and zero off-diagonal diffusion, odd derivatives in any direction are absent. Instead of the designed third order errors, the $\theta = \frac{1}{2}$ and McKee, Wall & Wilson (1996) schemes, have errors of fourth order. Their accuracy matches that of the $S = S_0$ scheme, that is designed to have fourth-order errors. In the absence of off-diagonal diffusion, the $S = S_{opt}$ scheme is designed to have fifth order errors. However, for zero flow it has errors of sixth order and

Time	Scheme	l_1	l_2	l_∞
Δt	$S = S_{opt}$	0.0926	0.1051	0.1163
	$S = S_0$	0.6893	0.4736	0.4936
	$\theta = 0$	1.0373	0.9547	0.8860
	$\theta = \frac{1}{2}$	0.1824	0.0935	0.0724
	$\theta = 1$	0.5017	0.4920	0.5195
	McKee	0.2181	0.1672	0.1713
$4\Delta t$	$S = S_{opt}$	0.0229	0.0262	0.0484
	$S = S_0$	0.1544	0.1305	0.2110
	$\theta = 0$	0.7451	0.6490	0.4515
	$\theta = \frac{1}{2}$	0.1698	0.2011	0.3951
	$\theta = 1$	0.3278	0.5825	1.4831
	McKee	0.1721	0.2210	0.4782
$16\Delta t$	$S = S_{opt}$	0.0011	0.0010	0.0020
	$S = S_0$	0.0395	0.0371	0.0663
	$\theta = 0$	0.2643	0.2457	0.2371
	$\theta = \frac{1}{2}$	0.0395	0.0407	0.0806
	$\theta = 1$	0.0870	0.1080	0.2539
	McKee	0.0397	0.0416	0.0839

Table 4.1: Diffusion test (4.31a)

Time	Scheme	l_1	l_2	l_∞
Δt	$S = S_{opt}$	0.0644	0.0632	0.0720
	$S = S_0$	0.6461	0.4091	0.4467
	$\theta = 0$	1.0409	1.0095	0.8624
	$\theta = \frac{1}{2}$	0.3200	0.2495	0.2593
	$\theta = 1$	0.6464	0.6844	0.8002
	McKee	0.3375	0.2119	0.1443
$4\Delta t$	$S = S_{opt}$	0.0347	0.0359	0.0442
	$S = S_0$	0.1483	0.1235	0.1897
	$\theta = 0$	0.8268	0.8140	0.7320
	$\theta = \frac{1}{2}$	0.2969	0.3159	0.4186
	$\theta = 1$	0.4678	0.7002	1.6243
	McKee	0.3968	0.3445	0.4797
$16\Delta t$	$S = S_{opt}$	0.0092	0.0100	0.0110
	$S = S_0$	0.0395	0.0356	0.0611
	$\theta = 0$	0.3422	0.3597	0.4697
	$\theta = \frac{1}{2}$	0.1262	0.1223	0.1765
	$\theta = 1$	0.2340	0.2498	0.3259
	McKee	0.3850	0.3759	0.4884

Table 4.2. Advection-diffusion test (4.31b)

coincides with the Mitchell & Fairweather (1964) optimal scheme for pure diffusion. After sixteen time-steps the optimal scheme is about a factor of forty superior to the alternatives.

Table 4.2 contains the error norms. For the $\theta = 0$ and $S = S_0$ schemes the error norms are similar to those in the zero flow case. The other schemes suffer substantial drops in accuracy. The $S = S_{opt}$ scheme remains the most accurate, followed by $S = S_0$. At sixteen time-steps the $\theta = \frac{1}{2}$ scheme has error norms between 12 and 16 times optimal. The error norms for the remaining schemes are larger. The second test includes advection

$$\Delta t = 0.2\,, \quad \Delta x = 1\,, \quad \lambda = 0\,, \quad u = v = w = 1\,,$$
$$\kappa_{11} = \kappa_{22} = \kappa_{33} = 0.8\,, \quad \kappa_{12} = \kappa_{13} = \kappa_{23} = 0\,. \tag{4.31b}$$

Figure 4.1 shows the solution after sixteen time-steps for the second test, along the arbitrar-

Time	Scheme	l_1	l_2	l_∞
Δt	$S = S_{opt}$	0.2286	0 1552	0.0944
	$S = S_0$	0.9702	0.4971	0 2560
	McKee	0.4872	0.3012	0 2283
$4\Delta t$	$S = S_{opt}$	0 0485	0.0540	0 0893
	$S = S_0$	0 4550	0 3081	0 2642
	McKee	0 4458	0 3136	0.4004
$16\Delta t$	$S = S_{opt}$	0 0154	0 0164	0 0271
	$S = S_0$	0.0546	0 0365	0 0483
	McKee	0.3844	0.3158	0 3584

Table 4 3: Off-diagonal test (4 31)

ily chosen slice $x = 8, \ldots, 20$ with $y = 11$, $z = 11$. The $S = S_{opt}$ scheme is indistinguishable from the continuous exact curve. For clarity, the plots for the other schemes are joined by dotted lines. The jaggedness of the $\theta = 0$ explicit scheme is a reminder of the vulnerability of that scheme to numerical instability. The relative proximity of the numerical results to the continuous exact curve does not fully conform with the error norms. In figure 4 1, the McKee, Wall & Wilson (1996) results looks better than either the $\theta = 0$ or $\theta = 1$ results, although the error norms would suggest the opposite.

The third test incorporates both advection and off-diagonal diffusion:

$$\Delta t = 0.2\,, \qquad \Delta x = 1\,, \quad u = v = w = 1\,,$$
$$\kappa_{11} = 0.8\,, \quad \kappa_{22} = 1\,, \quad \kappa_{33} = 0\,8\,, \quad \kappa_{12} = 0\,4\,, \quad \kappa_{13} = \kappa_{23} = 0\,. \tag{4.31c}$$

Table 4 3 contains the error norms for the suitably versatile $S = S_{opt}$, $S = S_0$ and McKee, Wall & Wilson (1996) schemes. After sixteen time-steps, the respective error norms are approximately in the ratio $1 \cdot 3 \quad 20$.

The tests were carried out in serial using Mathematica The $S = S_{opt}$, $S = S_0$ and McKee, Wall & Wilson (1996) ADI-schemes were approximately 20 times faster per time-step than non-ADI schemes, not taking into account the speed increase that would be

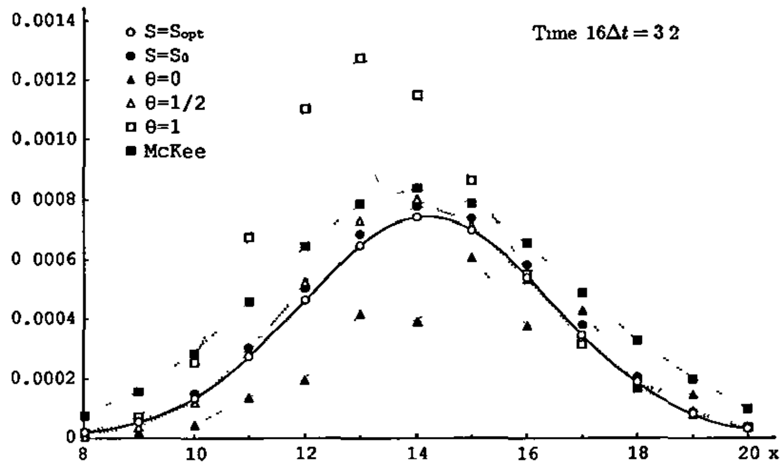

Figure 4.1 Numerical and exact results with flow and diffusion

attained by running the ADI schemes in parallel.

Formally the $S = S_{opt}$ and $S = S_0$ schemes both have mixed-direction errors at fourth order. The above tests are suggestive that by eliminating uni-directional $\partial^4_{x_i}$ errors, the $S = S_{opt}$ scheme is consistently more accurate than the $S = S_0$ scheme

4.9 Concluding remarks

An accurate compact finite difference scheme has been presented, that is structured in such a way that it has the potential to be exploited on parallel computers. Through the use of three-point difference operators and derivative expansions for the error, the scheme is straightforward to derive and simple to program. The high speed at which the tri-diagonal scheme can be solved, even on a serial computer, should not be underestimated The running time scales linearly with the total number of grid points and inversely to the number of processors used.

Chapter 5

Conclusion

A toolkit for deriving high-order parallelisable schemes has been presented along with numerical schemes, and results, for the 1D and 3D decay-advection-diffusion equations and the 1D linear KdV equation. The toolkit allows for the simple derivation of multi-dimensional schemes in terms of 1D difference operators. A simple recurrence relation gives the errors of these operators and it is the knowledge of these errors that allows high-order schemes to be derived in a straightforward manner. The benefits of high-order schemes are clear, the results are significantly more accurate offering the possibility of increasing time-steps and/or decreasing grid resolution whilst retaining results as good as, or better, than those given by other schemes

Along with accuracy, a major area of concern with any scheme is the solution time. For this reason the schemes involve solving tri-diagonal systems and, in higher dimensions, an ADI structure is enforced. With this structure in place the compact module still provides enough degrees of freedom to derive high-order schemes and the speed benefit of such a structure should not be underestimated Even when running in serial, the solution in any number of dimensions involves solving repeated tri-diagonal systems, an operation that is very fast and leads to a solution time that scales linearly with the total number of grid points, however many dimensions are involved. When running in parallel, the process of solving these schemes can be shared amongst multiple processors and thus the speed of solution increases proportionally With parallel computers becoming more commonplace, even available as desktop systems, schemes that can take advantage of this situation will become increasingly desirable

Conditions sufficient for stability in certain cases have been derived. These are typically related to the classical CFL condition along with a condition on the high-order parameter/s of the scheme. If the high-order condition is not met, the schemes can still be used by setting the relevant parameter to a given value, such as zero. As the results show this still provides highly accurate results

There are many ways in which this work can be continued. The methods can be applied to different equations and in various dimensions. The module size can be experimented with, since the toolkit works with any number of points. Non-trivial boundary conditions can be used to model different problems and research into applying the methods to non-linear problems, perhaps even on completely irregular grids, can take place

In practice it is the time reductions available that will ensure the work presented replaces the standard finite difference methods in the future. It is a stark reality that in business, time costs money, and in medicine and weather predictions, time costs lives so any methods that can improve on those currently used, such as those presented here, should be studied and applied

Appendix A

Finite difference formulae for derivatives

A.1 Introduction

Formulae are listed for n-point finite difference operators D_d that mimic the d'th derivative of a function f at some position χ, expressed in terms of the displacements $\alpha_i = x_i - \chi$. In the denominators, displacement differences $\alpha_i - \alpha_j$ can also be written as grid differences $x_i - x_j$. Errors for the finite difference representations are presented in appendix B

A.2 One-point formula

$$D_0[f] = f(x_1) . \tag{A.1}$$

A.3 Two-point formulae

$$D_0[f] = -\frac{\alpha_2 f(x_1)}{\alpha_1 - \alpha_2} - \frac{\alpha_1 f(x_2)}{\alpha_2 - \alpha_1} , \tag{A 2a}$$

$$D_1[f] = \frac{f(x_1)}{\alpha_1 - \alpha_2} + \frac{f(x_2)}{\alpha_2 - \alpha_1} . \tag{A 2b}$$

A.4 Three-point formulae

$$
\begin{aligned}
D_0[f] &= \frac{\alpha_2\alpha_3 f(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)}+\frac{\alpha_1\alpha_3 f(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)}+\frac{\alpha_1\alpha_2 f(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)}, && \text{(A 3a)}\\
D_1[f] &= -\frac{(\alpha_2+\alpha_3) f(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)}-\frac{(\alpha_1+\alpha_3) f(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)}-\frac{(\alpha_1+\alpha_2) f(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)}, && \text{(A 3b)}\\
D_2[f] &= \frac{2 f(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)}+\frac{2 f(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)}+\frac{2 f(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)} && \text{(A 3c)}
\end{aligned}
$$

A.5 Four-point formulae

$$
\begin{aligned}
D_0[f] = &-\frac{\alpha_2\alpha_3\alpha_4 f(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)(\alpha_1-\alpha_4)}-\frac{\alpha_1\alpha_3\alpha_4 f(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)(\alpha_2-\alpha_4)}\\
&-\frac{\alpha_1\alpha_2\alpha_4 f(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)(\alpha_3-\alpha_4)}-\frac{\alpha_1\alpha_2\alpha_3 f(x_4)}{(\alpha_4-\alpha_1)(\alpha_4-\alpha_2)(\alpha_4-\alpha_3)}, \qquad \text{(A 4a)}
\end{aligned}
$$

$$
\begin{aligned}
D_1[f] = &\frac{(\alpha_2\alpha_3+\alpha_2\alpha_4+\alpha_3\alpha_4) f(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)(\alpha_1-\alpha_4)}+\frac{(\alpha_1\alpha_3+\alpha_1\alpha_4+\alpha_3\alpha_4) f(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)(\alpha_2-\alpha_4)}\\
&+\frac{(\alpha_1\alpha_2+\alpha_1\alpha_4+\alpha_2\alpha_4) f(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)(\alpha_3-\alpha_4)}+\frac{(\alpha_1\alpha_2+\alpha_1\alpha_3+\alpha_2\alpha_3) f(x_4)}{(\alpha_4-\alpha_1)(\alpha_4-\alpha_2)(\alpha_4-\alpha_3)}, \qquad \text{(A.4b)}
\end{aligned}
$$

$$
\begin{aligned}
D_2[f] = &-\frac{2(\alpha_2+\alpha_3+\alpha_4) f(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)(\alpha_1-\alpha_4)}-\frac{2(\alpha_1+\alpha_3+\alpha_4) f(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)(\alpha_2-\alpha_4)}\\
&-\frac{2(\alpha_1+\alpha_2+\alpha_4) f(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)(\alpha_3-\alpha_4)}-\frac{2(\alpha_1+\alpha_2+\alpha_3) f(x_4)}{(\alpha_4-\alpha_1)(\alpha_4-\alpha_2)(\alpha_4-\alpha_3)}, \qquad \text{(A 4c)}
\end{aligned}
$$

$$
\begin{aligned}
D_3[f] = &\frac{6 f(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)(\alpha_1-\alpha_4)}+\frac{6 f(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)(\alpha_2-\alpha_4)}\\
&+\frac{6 f(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)(\alpha_3-\alpha_4)}+\frac{6 f(x_4)}{(\alpha_4-\alpha_1)(\alpha_4-\alpha_2)(\alpha_4-\alpha_3)}. \qquad \text{(A.4d)}
\end{aligned}
$$

A.6 Five-point formulae

$$\begin{aligned}
D_0[f] \;=\;& \frac{\alpha_2\alpha_3\alpha_4\alpha_5 f(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)(\alpha_1-\alpha_4)(\alpha_1-\alpha_5)} \\
&+\frac{\alpha_1\alpha_3\alpha_4\alpha_5 f(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)(\alpha_2-\alpha_4)(\alpha_2-\alpha_5)} \\
&+\frac{\alpha_1\alpha_2\alpha_4\alpha_5 f(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)(\alpha_3-\alpha_4)(\alpha_3-\alpha_5)} \\
&+\frac{\alpha_1\alpha_2\alpha_3\alpha_5 f(x_4)}{(\alpha_4-\alpha_1)(\alpha_4-\alpha_2)(\alpha_4-\alpha_3)(\alpha_4-\alpha_5)} \\
&+\frac{\alpha_1\alpha_2\alpha_3\alpha_4 f(x_5)}{(\alpha_5-\alpha_1)(\alpha_5-\alpha_2)(\alpha_5-\alpha_3)(\alpha_5-\alpha_4)},
\end{aligned} \tag{A.5a}$$

$$\begin{aligned}
D_1[f] \;=\;& -\frac{(\alpha_2\alpha_3\alpha_4+\alpha_2\alpha_3\alpha_5+\alpha_2\alpha_4\alpha_5+\alpha_3\alpha_4\alpha_5) f(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)(\alpha_1-\alpha_4)(\alpha_1-\alpha_5)} \\
&-\frac{(\alpha_1\alpha_3\alpha_4+\alpha_1\alpha_3\alpha_5+\alpha_1\alpha_4\alpha_5+\alpha_3\alpha_4\alpha_5) f(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)(\alpha_2-\alpha_4)(\alpha_2-\alpha_5)} \\
&-\frac{(\alpha_1\alpha_2\alpha_4+\alpha_1\alpha_2\alpha_5+\alpha_1\alpha_4\alpha_5+\alpha_2\alpha_4\alpha_5) f(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)(\alpha_3-\alpha_4)(\alpha_3-\alpha_5)} \\
&-\frac{(\alpha_1\alpha_2\alpha_3+\alpha_1\alpha_2\alpha_5+\alpha_1\alpha_3\alpha_5+\alpha_2\alpha_3\alpha_5) f(x_4)}{(\alpha_4-\alpha_1)(\alpha_4-\alpha_2)(\alpha_4-\alpha_3)(\alpha_4-\alpha_5)} \\
&-\frac{(\alpha_1\alpha_2\alpha_3+\alpha_1\alpha_2\alpha_4+\alpha_1\alpha_3\alpha_4+\alpha_2\alpha_3\alpha_4) f(x_5)}{(\alpha_5-\alpha_1)(\alpha_5-\alpha_2)(\alpha_5-\alpha_3)(\alpha_5-\alpha_4)},
\end{aligned} \tag{A.5b}$$

$$\begin{aligned}
D_2[f] \;=\;& \frac{2(\alpha_2\alpha_3+\alpha_2\alpha_4+\alpha_2\alpha_5+\alpha_3\alpha_4+\alpha_3\alpha_5+\alpha_4\alpha_5) f(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)(\alpha_1-\alpha_4)(\alpha_1-\alpha_5)} \\
&+\frac{2(\alpha_1\alpha_3+\alpha_1\alpha_4+\alpha_1\alpha_5+\alpha_3\alpha_4+\alpha_3\alpha_5+\alpha_4\alpha_5) f(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)(\alpha_2-\alpha_4)(\alpha_2-\alpha_5)} \\
&+\frac{2(\alpha_1\alpha_2+\alpha_1\alpha_4+\alpha_1\alpha_5+\alpha_2\alpha_4+\alpha_2\alpha_5+\alpha_4\alpha_5) f(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)(\alpha_3-\alpha_4)(\alpha_3-\alpha_5)} \\
&+\frac{2(\alpha_1\alpha_2+\alpha_1\alpha_3+\alpha_1\alpha_5+\alpha_2\alpha_3+\alpha_2\alpha_5+\alpha_3\alpha_5) f(x_4)}{(\alpha_4-\alpha_1)(\alpha_4-\alpha_2)(\alpha_4-\alpha_3)(\alpha_4-\alpha_5)} \\
&+\frac{2(\alpha_1\alpha_2+\alpha_1\alpha_3+\alpha_1\alpha_4+\alpha_2\alpha_3+\alpha_2\alpha_4+\alpha_3\alpha_4) f(x_5)}{(\alpha_5-\alpha_1)(\alpha_5-\alpha_2)(\alpha_5-\alpha_3)(\alpha_5-\alpha_4)},
\end{aligned} \tag{A.5c}$$

$$
\begin{aligned}
D_3[f] = & -\frac{6(\alpha_2+\alpha_3+\alpha_4+\alpha_5)f(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)(\alpha_1-\alpha_4)(\alpha_1-\alpha_5)} \\
& -\frac{6(\alpha_1+\alpha_3+\alpha_4+\alpha_5)f(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)(\alpha_2-\alpha_4)(\alpha_2-\alpha_5)} \\
& -\frac{6(\alpha_1+\alpha_2+\alpha_4+\alpha_5)f(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)(\alpha_3-\alpha_4)(\alpha_3-\alpha_5)} \\
& -\frac{6(\alpha_1+\alpha_2+\alpha_3+\alpha_5)f(x_4)}{(\alpha_4-\alpha_1)(\alpha_4-\alpha_2)(\alpha_4-\alpha_3)(\alpha_4-\alpha_5)} \\
& -\frac{6(\alpha_1+\alpha_2+\alpha_3+\alpha_4)f(x_5)}{(\alpha_5-\alpha_1)(\alpha_5-\alpha_2)(\alpha_5-\alpha_3)(\alpha_5-\alpha_4)},
\end{aligned}
\tag{A.5d}
$$

$$
\begin{aligned}
D_4[f] = & \frac{24f(x_1)}{(\alpha_1-\alpha_2)(\alpha_1-\alpha_3)(\alpha_1-\alpha_4)(\alpha_1-\alpha_5)} \\
& +\frac{24f(x_2)}{(\alpha_2-\alpha_1)(\alpha_2-\alpha_3)(\alpha_2-\alpha_4)(\alpha_2-\alpha_5)} \\
& +\frac{24f(x_3)}{(\alpha_3-\alpha_1)(\alpha_3-\alpha_2)(\alpha_3-\alpha_4)(\alpha_3-\alpha_5)} \\
& +\frac{24f(x_4)}{(\alpha_4-\alpha_1)(\alpha_4-\alpha_2)(\alpha_4-\alpha_3)(\alpha_4-\alpha_5)} \\
& +\frac{24f(x_5)}{(\alpha_5-\alpha_1)(\alpha_5-\alpha_2)(\alpha_5-\alpha_3)(\alpha_5-\alpha_4)}
\end{aligned}
\tag{A 5e}
$$

Appendix B

Finite difference errors

B.1 Introduction

Formulae are listed for the n-point elementary symmetric functions e_i in terms of the displacements $\alpha_i = x_i - \chi$, and for the first four error terms in the finite difference operators $D_d[f]$ that mimic the d'th derivative of a function $f(x)$ at a reference position χ (see appendix A)

B.2 One-point formula

Elementary symmetric functions

$$e_1 = \alpha_1 . \tag{B.1}$$

Error terms (Taylor series):

$$D_0[f] - f(\chi) = e_1 f'(\chi) + \frac{e_1^2}{2} f''(\chi) + \frac{e_1^3}{6} f^{(3)}(\chi) + \frac{e_1^4}{24} f^{(4)}(\chi) + \dots . \tag{B.2}$$

B.3 Two-point formulae

Elementary symmetric functions·

$$e_1 = \alpha_1 + \alpha_2 , \quad e_2 = \alpha_1 \alpha_2 . \tag{B 3}$$

Error terms.

$$
\begin{aligned}
D_0[f] - f(\chi) &= -\frac{e_2}{2} f''(\chi) - \frac{e_1 e_2}{6} f^{(3)}(\chi) - \frac{(e_1^2 - e_2) e_2}{24} f^{(4)}(\chi) \\
&\quad - \frac{(e_1^2 - 2\, e_2) e_1 e_2}{120} f^{(5)}(\chi) - \dots , \qquad \text{(B 4a)} \\
D_1[f] - f'(\chi) &= \frac{e_1}{2} f''(\chi) + \frac{e_1^2 - e_2}{6} f^{(3)}(\chi) + \frac{e_1^3 - 2 e_1 e_2}{24} f^{(4)}(\chi) \\
&\quad + \frac{e_1^4 - 3\, e_1^2 e_2 + e_2^2}{120} f^{(5)}(\chi) + \dots \qquad \text{(B.4b)}
\end{aligned}
$$

B.4 Three-point formulae

Elementary symmetric functions:

$$
e_1 = \alpha_1 + \alpha_2 + \alpha_3 , \quad e_2 = \alpha_1\alpha_2 + \alpha_1\alpha_3 + \alpha_2\alpha_3 , \quad e_3 = \alpha_1\alpha_2\alpha_3 . \qquad \text{(B 5)}
$$

Error terms.

$$
\begin{aligned}
D_0[f] - f(\chi) &= \frac{e_3}{6} f^{(3)}(\chi) + \frac{e_1 e_3}{24} f^{(4)}(\chi) + \frac{(e_1^2 - e_2) e_3}{120} f^{(5)}(\chi) \\
&\quad + \frac{(e_1^3 - 2\, e_1 e_2 + e_3) e_3}{720} f^{(6)}(\chi) + \dots , \qquad \text{(B.6a)} \\
D_1[f] - f'(\chi) &= -\frac{e_2}{6} f^{(3)}(\chi) - \frac{e_1 e_2 - e_3}{24} f^{(4)}(\chi) - \frac{e_1^2 e_2 - e_1 e_3 - e_2^2}{120} f^{(5)}(\chi) \\
&\quad - \frac{e_1^3 e_2 - e_1^2 e_3 - 2 e_1 e_2^2 + 2\, e_2 e_3}{720} f^{(6)}(\chi) - \dots , \qquad \text{(B 6b)} \\
D_2[f] - f''(\chi) &= \frac{e_1}{3} f^{(3)}(\chi) + \frac{e_1^2 - e_2}{12} f^{(4)}(\chi) + \frac{e_1^3 - 2\, e_1 e_2 + e_3}{60} f^{(5)}(\chi) \\
&\quad + \frac{e_1^4 - 3\, e_1^2 e_2 + 2\, e_1 e_3 + e_2^2}{360} f^{(6)}(\chi) + \dots \qquad \text{(B 6c)}
\end{aligned}
$$

B.5 Four-point formulae

Elementary symmetric functions:

$$e_1 = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 , \tag{B.7a}$$

$$e_2 = \alpha_1\alpha_2 + \alpha_1\alpha_3 + \alpha_1\alpha_4 + \alpha_2\alpha_3 + \alpha_2\alpha_4 + \alpha_3\alpha_4 , \tag{B.7b}$$

$$e_3 = \alpha_1\alpha_2\alpha_3 + \alpha_1\alpha_2\alpha_4 + \alpha_1\alpha_3\alpha_4 + \alpha_2\alpha_3\alpha_4 , \tag{B.7c}$$

$$e_4 = \alpha_1\alpha_2\alpha_3\alpha_4 . \tag{B.7d}$$

Error terms:

$$\begin{aligned} D_0[f] - f(\chi) &= -\frac{e_4}{24} f^{(4)}(\chi) - \frac{e_1 e_4}{120} f^{(5)}(\chi) - \frac{(e_1^2 - e_2)e_4}{720} f^{(6)}(\chi) \\ &\quad - \frac{({e_1}^3 - 2\,e_1 e_2 + e_3)e_4}{5040} f^{(7)}(\chi) - \ldots , \end{aligned} \tag{B 8a}$$

$$\begin{aligned} D_1[f] - f'(\chi) &= \frac{e_3}{24} f^{(4)}(\chi) + \frac{e_1 e_3 - e_4}{120} f^{(5)}(\chi) + \frac{e_1^2 e_3 - e_1 e_4 - e_2 e_3}{720} f^{(6)}(\chi) \\ &\quad + \frac{e_1^3 e_3 - e_1^2 e_4 - 2\,e_1 e_2 e_3 + e_2 e_4 + e_3^2}{5040} f^{(7)}(\chi)) + \ldots , \end{aligned} \tag{B.8b}$$

$$\begin{aligned} D_2[f] - f''(\chi) &= -\frac{e_2}{12} f^{(4)}(\chi) - \frac{e_1 e_2 - e_3}{60} f^{(5)}(\chi) - \frac{e_1^2 e_2 - e_1 e_3 - e_2^2 + e_4}{360} f^{(6)}(\chi) \\ &\quad - \frac{e_2 e_1^3 - e_1^2 e_3 - 2\,e_2^2 e_1 + e_1 e_4 + 2\,e_2 e_3}{2520} f^{(7)}(\chi) - \ldots , \end{aligned} \tag{B.8c}$$

$$\begin{aligned} D_3[f] - f^{(3)}(\chi) &= \frac{e_1}{4} f^{(4)}(\chi) + \frac{e_1^2 - e_2}{20} f^{(5)}(\chi) + \frac{e_1^3 - 2e_1 e_2 + e_3}{120} f^{(6)}(\chi) \\ &\quad + \frac{e_1^4 - 3\,e_1^2 e_2 + 2\,e_1 e_3 + e_2^2 - e_4}{840} f^{(7)}(\chi) + \ldots . \end{aligned} \tag{B.8d}$$

B.6 Five-point formulae

Elementary symmetric functions.

$$e_1 = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5\,, \tag{B 9a}$$

$$\begin{aligned} e_2 = {} & \alpha_1\alpha_2 + \alpha_1\alpha_3 + \alpha_1\alpha_4 + \alpha_1\alpha_5 + \alpha_2\alpha_3 + \alpha_2\alpha_4 + \alpha_2\alpha_5 + \alpha_3\alpha_4 \\ & + \alpha_3\alpha_5 + \alpha_4\alpha_5\,, \end{aligned} \tag{B 9b}$$

$$\begin{aligned} e_3 = {} & \alpha_1\alpha_2\alpha_3 + \alpha_1\alpha_2\alpha_4 + \alpha_1\alpha_2\alpha_5 + \alpha_1\alpha_3\alpha_4 + \alpha_1\alpha_3\alpha_5 + \alpha_1\alpha_4\alpha_5 \\ & + \alpha_2\alpha_3\alpha_4 + \alpha_2\alpha_3\alpha_5 + \alpha_2\alpha_4\alpha_5 + \alpha_3\alpha_4\alpha_5\,, \end{aligned} \tag{B.9c}$$

$$e_4 = \alpha_1\alpha_2\alpha_3\alpha_4 + \alpha_1\alpha_2\alpha_3\alpha_5 + \alpha_1\alpha_2\alpha_4\alpha_5 + \alpha_1\alpha_3\alpha_4\alpha_5 + \alpha_2\alpha_3\alpha_4\alpha_5\,, \tag{B.9d}$$

$$e_5 = \alpha_1\alpha_2\alpha_3\alpha_4\alpha_5\,. \tag{B.9e}$$

Error terms

$$\begin{aligned} D_0[f] - f(\chi) = {} & \frac{e_5}{120} f^{(5)}(\chi) + \frac{e_1 e_5}{720} f^{(6)}(\chi) + \frac{(e_1^2 - e_2)e_5}{5040} f^{(7)}(\chi) \\ & + \frac{\left(e_1^3 - 2\,e_1 e_2 + e_3\right) e_5}{40320} f^{(8)}(\chi) + \ldots\,, \end{aligned} \tag{B.10a}$$

$$\begin{aligned} D_1[f] - f'(\chi) = {} & -\frac{e_4}{120} f^{(5)}(\chi) - \frac{e_1 e_4 - e_5}{720} f^{(6)}(\chi) - \frac{e_1^2 e_4 - e_1 e_5 - e_2 e_4}{5040} f^{(7)}(\chi) \\ & - \frac{e_1^3 e_4 - e_1^2 e_5 - 2\,e_1 e_2 e_4 + e_2 e_5 + e_3 e_4}{40320} f^{(8)}(\chi) - \ldots\,, \end{aligned} \tag{B 10b}$$

$$\begin{aligned} D_2[f] - f''(\chi) = {} & \frac{e_3}{60} f^{(5)}(\chi) + \frac{e_1 e_3 - e_4}{360} f^{(6)}(\chi) + \frac{e_1^2 e_3 - e_1 e_4 - e_2 e_3 + e_5}{2520} f^{(7)}(\chi) \\ & + \frac{e_1^3 e_3 - e_1^2 e_4 - 2\,e_1 e_2 e_3 + e_1 e_5 + e_2 e_4 + e_3^2}{20160} f^{(8)}(\chi) + \ldots\,, \end{aligned} \tag{B.10c}$$

$$\begin{aligned} D_3[f] - f^{(3)}(\chi) = {} & -\frac{e_2}{20} f^{(5)}(\chi) - \frac{e_1 e_2 - e_3}{120} f^{(6)}(\chi) - \frac{e_1^2 e_2 - e_1 e_3 - e_2^2 + e_4}{840} f^{(7)}(\chi) \\ & - \frac{e_1^3 e_2 - e_1^2 e_3 - 2\,e_1 e_2^2 + e_1 e_4 + 2\,e_2 e_3 - e_5}{6720} f^{(8)}(\chi) - \ldots\,, \end{aligned} \tag{B.10d}$$

$$\begin{aligned} D_4[f] - f^{(4)}(\chi) = {} & \frac{e_1}{5} f^{(5)}(\chi) + \frac{e_1^2 - e_2}{30} f^{(6)}(\chi) + \frac{e_1^3 - 2e_1 e_2 + e_3}{210} f^{(7)}(\chi) \\ & + \frac{e_1^4 - 3\,e_1^2 e_2 + 2\,e_1 e_3 + e_2^2 - e_4}{1680} f^{(8)}(\chi) + \ldots \end{aligned} \tag{B 10e}$$

Appendix C

Stability proofs for quadratic inequalities

C.1 Introduction

In deriving stability calculations it is often necessary to find sufficient conditions such that a quadratic inequality holds true over a bounded (e g. sine-squared) variable. §C.2 states the problem in 1D and derives equivalent conditions, one set of which serves as sufficient conditions, linear in the coefficients of the quadratic inequality. A simple geometrical interpretation of these results is made in §C 3. By repeated application of these results, a table of sufficient conditions is generated for the 2D and 3D cases in §C.4.

C.2 Derivation in one dimension

Let $0 \leqslant \zeta(k) \leqslant 1$, with $\zeta(k_1) = 0$ and $\zeta(k_2) = 1$ for some k_1, k_2 and consider the quadratic inequality $\alpha + \beta\zeta + \gamma\zeta^2 \geqslant 0$, with α, β, γ, $\zeta \in \mathbb{R}$ and $\gamma \neq 0$. The problem is to find inequalities as functions of α, β, γ (independent of ζ) that satisfy the quadratic inequality with given constraints. A trivial solution is to require all coefficients α, β and γ to be non-negative, and, along with the non-negativity of ζ, these are sufficient conditions to satisfy the quadratic inequality. In practice these restrictions are not flexible enough to yield useful stability criteria. This section proves that the problem is equivalent to the conditions.

$$(\alpha \geqslant 0 \text{ and } 2\alpha + \beta \leqslant 0 \text{ and } \beta^2 - 4\alpha\gamma \leqslant 0) \tag{C.1a}$$

$$\text{or} \quad (\alpha \geqslant 0 \text{ and } 2\alpha + \beta \geqslant 0 \text{ and } \alpha + \beta + \gamma \geqslant 0). \tag{C.1b}$$

The veracity of the inequalities (C.1b) provides relaxed conditions that are sufficient in solving the problem, whilst being linear in the coefficients (and therefore simple to apply). This makes (C.1b) particularly suitable for stability calculations of the form described.

Proof. A quadratic equation has three degrees of freedom, that as well as being interpreted as the coefficients α, β and γ multiplying increasing powers of ζ, can also be interpreted in a geometrical manner by writing the equation in the form $\gamma(\zeta - x_1)(\zeta - x_2)$. Thus it is clear that γ denotes a scalar factor/orientation along with two, possibly equal, real or complex roots $x_{1,2}$. The roots are given by $x_{1,2} = \frac{-\beta \pm \sqrt{\beta^2 - 4\alpha\gamma}}{2\gamma}$, with x_1 taking the negative sign. For real roots, and when $\gamma \geqslant 0$, then $x_1 \leqslant x_2$ so that x_1 is the left-most root. With $\gamma \leqslant 0$ the situation is reversed and x_1 is to the right. Since the two quadratic representations are equivalent, the problem is solved by consideration of the second form of the quadratic for its ease of geometrical interpretation.

Figure C.1 shows generic examples of all four cases of quadratic equation that satisfy the problem. These are found by consideration of the degrees of freedom that affect the solution, i.e. the orientation and position of the roots relative to the interval $0 \leqslant \zeta \leqslant 1$. The proof is thus reduced to enumerating these cases and showing their equivalence to the conditions (C.1a,b).

Consider the four cases in turn:

Case A consists of two imaginary roots, so that the discriminant $\beta^2 - 4\alpha\gamma \leqslant 0$ (for ease of proof this also includes the case of zero discriminant with two equal real roots). Since the quadratic is restricted to the upper half of the plane, the quadratic is non-negative for all ζ and thus in particular for $\zeta = 0$ so that $\alpha \geqslant 0$. Conversely, with the quadratic restricted

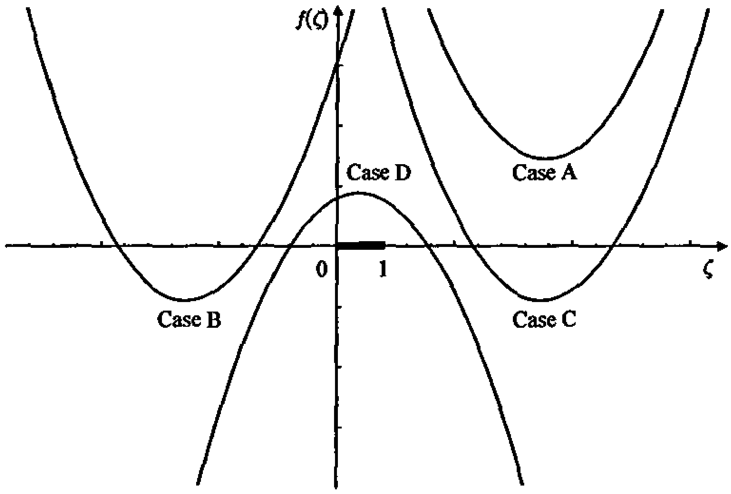

Figure C 1· Quadratic cases

to one half plane, if $\alpha \geqslant 0$ then the quadratic is non-negative at $\zeta = 0$ and hence must be case A

Case B consists of two (possibly equal) real roots, with discriminant $\beta^2 - 4\alpha\gamma \geqslant 0$ That the quadratic points upwards is equivalent to $\gamma \geqslant 0$ The final characteristic of case B is that the right-most root x_2 is before the left of the interval $0 \leqslant \zeta \leqslant 1$ at $\zeta(k_1) = 0$ so that $x_2 \leqslant \zeta(k_1) = 0$.

Case C consists of two (possibly equal) real roots, with discriminant $\beta^2 - 4\alpha\gamma \geqslant 0$ The quadratic points upwards and as before this is equivalent to $\gamma \geqslant 0$. The left-most root x_1 is beyond the right of the interval $0 \leqslant \zeta \leqslant 1$ at $\zeta(k_2) = 1$ so that $x_1 \geqslant \zeta(k_2) = 1$

Case D consists of two (possibly equal) real roots, with discriminant $\beta^2 - 4\alpha\gamma \geqslant 0$. The quadratic points downwards so that $\gamma \leqslant 0$. The roots straddle the interval $0 \leqslant \zeta \leqslant 1$ so that the left root $x_2 \leqslant \zeta(k_1) = 0$ and the right root $x_1 \geqslant \zeta(k_2) = 1$.

The converses of all cases follow simply so that equality exists. To summarise.

$$\text{Case A} \Leftrightarrow \beta^2 - 4\alpha\gamma \leqslant 0 \text{ and } \alpha \geqslant 0\,, \tag{C.2a}$$

$$\text{Case B} \Leftrightarrow \beta^2 - 4\alpha\gamma \geqslant 0 \text{ and } \gamma \geqslant 0 \text{ and } x_2 = \frac{-\beta + \sqrt{\beta^2 - 4\alpha\gamma}}{2\gamma} \leqslant 0\,, \tag{C.2b}$$

$$\text{Case C} \Leftrightarrow \beta^2 - 4\alpha\gamma \geqslant 0 \text{ and } \gamma \geqslant 0 \text{ and } x_1 = \frac{-\beta - \sqrt{\beta^2 - 4\alpha\gamma}}{2\gamma} \geqslant 1\,, \tag{C 2c}$$

$$\text{Case D} \Leftrightarrow \beta^2 - 4\alpha\gamma \geqslant 0 \text{ and } \gamma \leqslant 0 \text{ and } x_2 = \frac{-\beta + \sqrt{\beta^2 - 4\alpha\gamma}}{2\gamma} \leqslant 0$$

$$\text{and } x_1 = \frac{-\beta - \sqrt{\beta^2 - 4\alpha\gamma}}{2\gamma} \geqslant 1\,. \tag{C 2d}$$

With mathematical descriptions of the four cases in place, in terms of the coefficients α, β and γ, it is left to show the equivalence of the inequalities (C 2a-d) to the conditions (C.1a,b).

Proposition It will be shown that each of the four cases A, B, C and D implies the inequalities (C.1a,b).

Case A (C.2a) It is given that $\alpha \geqslant 0$ and $\beta^2 - 4\alpha\gamma \leqslant 0$.

Consider the case when $2\alpha + \beta \leqslant 0$ Then (C.1a) is immediately satisfied

Consider the case when $2\alpha + \beta \geqslant 0$. Then $\alpha \geqslant 0$ and $\beta^2 - 4\alpha\gamma \leqslant 0$ imply $\gamma \geqslant 0$ Adding to each side of the discriminant condition yields $\beta^2 - 4\alpha\gamma + 4\beta\gamma + 4\gamma^2 \leqslant 4\beta\gamma + 4\gamma^2$ and rearranging to complete the square gives $(\beta + 2\gamma)^2 \leqslant 4\gamma(\alpha + \beta + \gamma)$ The left side is non-negative, so the right side must also be non-negative. Since $\gamma \geqslant 0$ then $\alpha + \beta + \gamma \geqslant 0$, which implies (C.1b)

Case B (C.2b). It is given that $\gamma \geqslant 0$, $\beta^2 - 4\alpha\gamma \geqslant 0$ and $x_2 = \frac{-\beta + \sqrt{\beta^2 - 4\alpha\gamma}}{2\gamma} \leqslant 0$

The denominator of x_2 is positive so that the numerator must be negative. Then $\sqrt{\beta^2 - 4\alpha\gamma} \leqslant \beta$ so that $\beta \geqslant 0$. The conditions $\gamma \geqslant 0$ and $\beta^2 - 4\alpha\gamma \geqslant 0$ imply that $\alpha \geqslant 0$. With all coefficients non-negative then $2\alpha + \beta \geqslant 0$ and $\alpha + \beta + \gamma \geqslant 0$ which gives (C 1b).

Case C (C 2c): It is given that $\gamma \geqslant 0$, $\beta^2 - 4\alpha\gamma \geqslant 0$ and $x_1 = \frac{-\beta - \sqrt{\beta^2 - 4\alpha\gamma}}{2\gamma} \geqslant 1$.

The denominator of x_1 is positive so that $2\gamma+\beta \leqslant -\sqrt{\beta^2-4\alpha\gamma}$ and then $2\gamma+\beta \leqslant 0$ Since $\gamma \geqslant 0$ then $\beta \leqslant 0$. Squaring both negative sides of $2\gamma+\beta \leqslant -\sqrt{\beta^2-4\alpha\gamma}$ gives $4\gamma^2+4\beta\gamma+\beta^2 = 4\gamma(\alpha+\beta+\gamma) \geqslant \beta^2-4\alpha\gamma \geqslant 0$ Since $\gamma \geqslant 0$ then $\alpha+\beta+\gamma \geqslant 0$ With $2\gamma+\beta \leqslant 0$ then $-2\gamma-\beta \geqslant 0$ Adding this to $2\alpha+2\beta+2\gamma \geqslant 0$ gives $2\alpha+\beta \geqslant 0$ which yields (C.1b).

Case D (C 2d): It is given that $\gamma \leqslant 0$, $\beta^2-4\alpha\gamma \geqslant 0$, $x_1 = \frac{-\beta+\sqrt{\beta^2-4\alpha\gamma}}{2\gamma} \leqslant 0$ and $x_2 = \frac{-\beta-\sqrt{\beta^2-4\alpha\gamma}}{2\gamma} \geqslant 1$

The denominator of x_1 is negative so that $\sqrt{\beta^2-4\alpha\gamma} \geqslant \beta$. Both sides are non-negative so that squaring them gives $\beta^2-4\alpha\gamma \geqslant \beta^2$. Thus $4\alpha\gamma \leqslant 0$ so that $\alpha \geqslant 0$. The denominator of x_2 is negative so that $\beta+2\gamma \geqslant -\sqrt{\beta^2-4\alpha\gamma}$. Since $\gamma \leqslant 0$ then trivially $\beta \geqslant \beta+2\gamma$ Combining these inequalities gives $\sqrt{\beta^2-4\alpha\gamma} \geqslant \beta \geqslant \beta+2\gamma \geqslant -\sqrt{\beta^2-4\alpha\gamma}$ so that $|\beta+2\gamma| \leqslant \sqrt{\beta^2-4\alpha\gamma}$. Both sides are positive so that upon squaring $\beta^2+4\gamma^2+4\beta\gamma \leqslant \beta^2-4\alpha\gamma$. Rearranging gives $4\gamma(\alpha+\beta+\gamma) \leqslant 0$ so that $\alpha+\beta+\gamma \geqslant 0$ since $\gamma \leqslant 0$ Then $\alpha+\beta+\gamma \geqslant \gamma$ so that $a+\beta \geqslant 0$ and finally $2\alpha+\beta \geqslant 0$ since $\alpha \geqslant 0$ which yields (C.1b).

Converse The final part of the proof is to show that (C 1a,b) implies one of the cases A, B, C or D.

The inequality (C.1b) is split into two cases so that (C.1a,b) become:

$$(\alpha \geqslant 0 \text{ and } 2\alpha+\beta \leqslant 0 \text{ and } \beta^2-4\alpha\gamma \leqslant 0) \tag{C.3a}$$

$$\text{or} \quad (\alpha \geqslant 0 \text{ and } 2\alpha+\beta \geqslant 0 \text{ and } \alpha+\beta+\gamma \geqslant 0 \text{ and } \beta^2-4\alpha\gamma \leqslant 0) \tag{C.3b}$$

$$\text{or} \quad (\alpha \geqslant 0 \text{ and } 2\alpha+\beta \geqslant 0 \text{ and } \alpha+\beta+\gamma \geqslant 0 \text{ and } \beta^2-4\alpha\gamma \geqslant 0)\,. \tag{C.3c}$$

The two inequalities (C 3a,b) both imply case A (C.2a). This leaves the inequality (C.3c) which is split into three cases.

$$((\text{C.3c}) \text{ and } \beta \geqslant 0 \text{ and } \gamma \geqslant 0) \tag{C.4a}$$

$$\text{or} \quad ((\text{C.3c}) \text{ and } \beta \leqslant 0 \text{ and } \gamma \geqslant 0) \tag{C.4b}$$

$$\text{or} \quad ((\text{C 3c}) \text{ and } \gamma \leqslant 0)\,. \tag{C.4c}$$

These conditions (C 4a-c) are examined in turn in the following three cases:

Case 1 (C 4a): $\alpha \geqslant 0$ and $\gamma \geqslant 0$ so that $4\alpha\gamma \geqslant 0$ and thus $\beta^2 - 4\alpha\gamma \leqslant \beta^2$. Both sides are the squares of positive quantities so, on taking the square-root, $\sqrt{\beta^2 - 4\alpha\gamma} \leqslant \beta$ Rearranging and dividing by the positive value 2γ gives $x_2 = \frac{-\beta+\sqrt{\beta^2-4\alpha\gamma}}{2\gamma} \leqslant 0$ so that all the conditions for case B (C 2b) are satisfied.

Case 2 (C.4b): $\beta^2 - 4\alpha\gamma \geqslant 0$ immediately gives $4\alpha\gamma \leqslant \beta^2$. $2\alpha + \beta \geqslant 0$ and $\beta \leqslant 0$ imply that $2\alpha\beta + \beta^2 \leqslant 0$ so that $\beta^2 \leqslant -2\alpha\beta$. Combining these inequalities gives $4\alpha\gamma \leqslant \beta^2 \leqslant -2\alpha\beta$ so that $4\alpha\gamma \leqslant -2\alpha\beta$. Dividing through by $2\alpha \geqslant 0$ yields $2\gamma + \beta \leqslant 0$.

The inequalities $\alpha + \beta + \gamma \geqslant 0$ and $\gamma \geqslant 0$ imply $4\alpha\gamma + 4\beta\gamma + 4\gamma^2 + \beta^2 \geqslant \beta^2$ so that after rearranging and completing the square then $\beta^2 - 4\alpha\gamma \leqslant (2\gamma + \beta)^2$. Since $2\gamma + \beta \leqslant 0$ then taking the square-root implies $\sqrt{\beta^2 - 4\alpha\gamma} \leqslant |2\gamma + \beta| = -(2\gamma + \beta)$ Rearranging and dividing by the positive value 2γ gives $x_1 = \frac{-\beta-\sqrt{\beta^2-4\alpha\gamma}}{2\gamma} \geqslant 1$ so that the conditions for case C (C.2c) are satisfied

Case 3 (C.4c) $\alpha \geqslant 0$ and $\gamma \leqslant 0$ imply $4\alpha\gamma \leqslant 0$ so that $\beta^2 \leqslant \beta^2 - 4\alpha\gamma$. On taking the square-root then $\sqrt{\beta^2 - 4\alpha\gamma} \geqslant |\beta|$ and since $|\beta| \geqslant \beta$ then $\sqrt{\beta^2 - 4\alpha\gamma} \geqslant \beta$ After rearranging and dividing by $2\gamma \leqslant 0$ then $x_2 = \frac{-\beta+\sqrt{\beta^2-4\alpha\gamma}}{2\gamma} \leqslant 0$.

With $\alpha+\beta+\gamma \geqslant 0$ and $\gamma \leqslant 0$ then $4\alpha\gamma+4\beta\gamma+4\gamma^2+\beta^2 \leqslant \beta^2$ and, after rearranging and completing the square, $(2\gamma + \beta)^2 \leqslant \beta^2 - 4\alpha\gamma$. Taking the square-root gives $\sqrt{\beta^2 - 4\alpha\gamma} \geqslant |2\gamma + \beta| \geqslant -(2\gamma + \beta)$ Rearranging and dividing by $2\gamma \leqslant 0$ gives $x_1 = \frac{-\beta-\sqrt{\beta^2-4\alpha\gamma}}{2\gamma} \geqslant 1$, completing the conditions for case D (C 2d). □

C.3 Geometrical interpretation

Geometrically the condition (C 1b) can be interpreted as shown in figure (C.2) The value of the quadratic at $\zeta = 0$ (point A) and at $\zeta = 1$ (point C) must both be non-negative and the tangent through $\zeta = 0$ and $\zeta = 1$ evaluated at their intersection at $\zeta = \frac{1}{2}$ (point B) must also be non-negative. The remaining condition (C 1a) consists of those quadratics with imaginary roots that have negative values at point B.

Proof The value of the quadratic $\alpha + \beta\zeta + \gamma\zeta^2$ evaluated at $\zeta = 0$ is α so the inequality $\alpha \geqslant 0$ states that this value must be non-negative. Similarly, evaluating the quadratic at $\zeta = 1$ gives the value $\alpha + \beta + \gamma$ which also must be non-negative by (C 1b)

The tangent to the quadratic $\alpha + \beta\zeta + \gamma\zeta^2$ at $\zeta = \zeta_0$ is given by

$$(\beta + 2\gamma\zeta_0)\zeta + \alpha - \gamma\zeta_0^2. \tag{C 5}$$

This intersects with the tangent at $\zeta = \zeta_1$ when $(\beta+2\gamma\zeta_0)\zeta+\alpha-\gamma\zeta_0^2 = (\beta+2\gamma\zeta_1)\zeta+\alpha-\gamma\zeta_1^2$ i e at the midpoint $\zeta = \frac{1}{2}(\zeta_0 + \zeta_1)$. The condition $2\alpha + \beta \geqslant 0 \Leftrightarrow \alpha + \frac{\beta}{2} \geqslant 0$ is of the form (C 5) with $\zeta = \frac{1}{2}$ and $\zeta_0 = 0$ and thus implies that the tangent through the points $\zeta = 0$ and $\zeta = 1$, meeting at $\zeta = \frac{1}{2}$, must be non-negative.

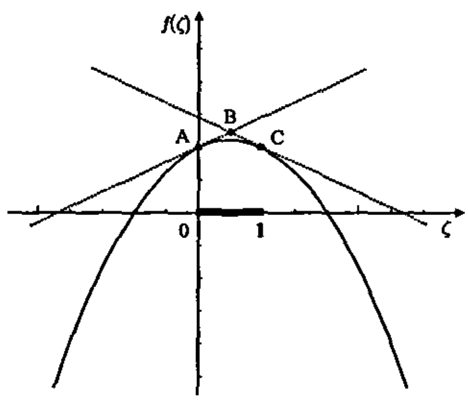

Figure C.2· Geometrical interpretation of (C.1b)

For (C 1a), the discriminant implies that the quadratics have imaginary roots. With a similar argument as above, the condition $2\alpha + \beta \leqslant 0$ further reduces the set of quadratics to those that have a negative value at the intersection at $\zeta = \frac{1}{2}$ formed by the tangents to the quadratic at $\zeta = 0$ and $\zeta = 1$. □

C.4 Application to two and three dimensions

Consider a quadratic inequality over three dimensions,

$$\begin{aligned} & a_1 + a_2\zeta_1 + a_3\zeta_1^2 + \left(a_4 + a_5\zeta_1 + a_6\zeta_1^2\right)\zeta_2 + \left(a_7 + a_8\zeta_1 + a_9\zeta_1^2\right)\zeta_2^2 \\ + & \left[a_{10} + a_{11}\zeta_1 + a_{12}\zeta_1^2 + \left(a_{13} + a_{14}\zeta_1 + a_{15}\zeta_1^2\right)\zeta_2 + \left(a_{16} + a_{17}\zeta_1 + a_{18}\zeta_1^2\right)\zeta_2^2\right]\zeta_3 \\ + & \left[a_{19} + a_{20}\zeta_1 + a_{21}\zeta_1^2 + \left(a_{22} + a_{23}\zeta_1 + a_{24}\zeta_1^2\right)\zeta_2 + \left(a_{25} + a_{26}\zeta_1 + a_{27}\zeta_1^2\right)\zeta_2^2\right]\zeta_3^2 \geqslant 0\,, \end{aligned} \tag{C 6}$$

with $0 \leqslant \zeta_1, \zeta_2, \zeta_3 \leqslant 1$. A trivial sufficient solution is to require all the coefficients a_j, $j = 1, \ldots, 27$ to be non-negative, but in practice this is too severe a requirement that does not yield useful conditions for stability.

Instead, (C.1b) is applied repeatedly to give a set of 27 inequalities, which if all satisfied prove sufficiency of the inequality (C.6). The non-linear condition (C 1a) is not required in proving sufficiency

First, (C 6) is written in the form $\alpha + \beta\zeta_3 + \gamma\zeta_3^2 \geqslant 0$ where

$$\alpha = a_1 + a_2\zeta_1 + a_3\zeta_1^2 + \left(a_4 + a_5\zeta_1 + a_6\zeta_1^2\right)\zeta_2 + \left(a_7 + a_8\zeta_1 + a_9\zeta_1^2\right)\zeta_2^2\,, \tag{C.7a}$$

$$\beta = a_{10} + a_{11}\zeta_1 + a_{12}\zeta_1^2 + \left(a_{13} + a_{14}\zeta_1 + a_{15}\zeta_1^2\right)\zeta_2 + \left(a_{16} + a_{17}\zeta_1 + a_{18}\zeta_1^2\right)\zeta_2^2\,, \tag{C 7b}$$

$$\gamma = a_{19} + a_{20}\zeta_1 + a_{21}\zeta_1^2 + \left(a_{22} + a_{23}\zeta_1 + a_{24}\zeta_1^2\right)\zeta_2 + \left(a_{25} + a_{26}\zeta_1 + a_{27}\zeta_1^2\right)\zeta_2^2\,, \tag{C.7c}$$

so that (C.1b) is directly applicable. This gives three inequalities, each of which is again quadratic, of the form $\alpha + \beta\zeta_2 + \gamma\zeta_2^2 \geqslant 0$ (reusing the notation α, β and γ). Thus (C.1b) can be applied again directly to yield nine inequalities. The final step involves inequalities of the form $\alpha+\beta\zeta_1+\gamma\zeta_1^2 \geqslant 0$ so that (C.1b) is again applicable, resulting in 27 inequalities. Thus a complicated non-linear inequality is reduced to a series of 27 inequalities Each inequality is linear and of the form·

$$\sum_{j=1}^{27} r_j a_j \geqslant 0. \tag{C 8}$$

The coefficients r_j are listed in table C.1 (with dashes corresponding to zero)

The first three inequalities are sufficient conditions for the 1D inequality $a_1 + a_2\zeta_1 + a_3\zeta_1^2 \geqslant 0$ (with $\zeta_2 = \zeta_3 = 0$)

$$a_1 \geqslant 0, \tag{C.9a}$$

$$2a_1 + a_2 \geqslant 0, \tag{C 9b}$$

$$a_1 + a_2 + a_3 \geqslant 0. \tag{C 9c}$$

These are the sufficient conditions (C.1b) Similarly, the first nine inequalities correspond to the 2D case with $\zeta_3 = 0$

	Index j of coefficients r_j																										
No	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2	2	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3	1	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4	2	-	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
5	4	2	-	2	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
6	2	2	2	1	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
7	1	-	-	1	-	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
8	2	1	-	2	1	-	2	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
9	1	1	1	1	1	1	1	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
10	2	-	-	-	-	-	-	-	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
11	4	2	-	-	-	-	-	-	-	2	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
12	2	2	2	-	-	-	-	-	-	1	1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
13	4	-	-	2	-	-	-	-	-	2	-	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-
14	8	4	-	4	2	-	-	-	-	4	2	-	2	1	-	-	-	-	-	-	-	-	-	-	-	-	-
15	4	4	4	2	2	2	-	-	-	2	2	2	1	1	1	-	-	-	-	-	-	-	-	-	-	-	-
16	2	-	-	2	-	-	2	-	-	1	-	-	1	-	-	1	-	-	-	-	-	-	-	-	-	-	-
17	4	2	-	4	2	-	4	2	-	2	1	-	2	1	-	2	1	-	-	-	-	-	-	-	-	-	-
18	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	-	-	-	-	-	-	-	-	-
19	1	-	-	-	-	-	-	-	-	1	-	-	-	-	-	-	-	-	1	-	-	-	-	-	-	-	-
20	2	1	-	-	-	-	-	-	-	2	1	-	-	-	-	-	-	-	2	1	-	-	-	-	-	-	-
21	1	1	1	-	-	-	-	-	-	1	1	1	-	-	-	-	-	-	1	1	1	-	-	-	-	-	-
22	2	-	-	1	-	-	-	-	-	2	-	-	1	-	-	-	-	-	2	-	-	1	-	-	-	-	-
23	4	2	-	2	1	-	-	-	-	4	2	-	2	1	-	-	-	-	4	2	-	2	1	-	-	-	-
24	2	2	2	1	1	1	-	-	-	2	2	2	1	1	1	-	-	-	2	2	2	1	1	1	-	-	-
25	1	-	-	1	-	-	1	-	-	1	-	-	1	-	-	1	-	-	1	-	-	1	-	-	1	-	-
26	2	1	-	2	1	-	2	1	-	2	1	-	2	1	-	2	1	-	2	1	-	2	1	-	2	1	-
27	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Table C.1: Coefficients of the inequalities (C 8)

Appendix D

Solution of tri-diagonal systems

D.1 Introduction

The speed at which a scheme can be solved is an important factor to consider when choosing or designing a scheme Finite difference schemes involve solving banded systems, and in particular tri-diagonal systems when a module with three spatial points is used. For the particular case of solving such a 1D scheme in parallel, specific methods exist to take advantage of such capabilities, such as recursive doubling (Stone 1973) and recursive striding (Evans 1997). To solve a 1D scheme in serial, or a higher dimensional ADI scheme in either serial or parallel, a fast method such as the Thomas algorithm (Richtmyer & Morton 1967 §8 5, Mitchell & Griffiths 1980 §2.5, Sebben & Baliga 1995) as described here is applicable

D.2 Derivation

The solution is sought to a tri-diagonal system of N equations

$$\begin{pmatrix} b_1 & c_1 & 0 & 0 & 0 & & \\ a_2 & b_2 & c_2 & 0 & 0 & & \\ 0 & a_3 & b_3 & c_3 & 0 & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & 0 & a_{N-2} & b_{N-2} & c_{N-2} & 0 \\ & & 0 & 0 & a_{N-1} & b_{N-1} & c_{N-1} \\ & & 0 & 0 & 0 & a_N & b_N \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \\ C_3 \\ \vdots \\ C_{N-2} \\ C_{N-1} \\ C_N \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_{N-2} \\ d_{N-1} \\ d_N \end{pmatrix}. \qquad \text{(D.1)}$$

Each equation is of the form

$$a_i C_{i-1} + b_i C_i + c_i C_{i+1} = d_i \,, \quad i = 1, \ldots, N \,, \tag{D.2}$$

with $a_1 = c_N = 0$ The first equation is written as

$$C_1 + \frac{c_1}{b_1} C_2 = \frac{d_1}{b_1} \,. \tag{D 3}$$

Substituting this into the next equation gives a relation in terms of C_2 and C_3 and so on, yielding, in general, the relation

$$C_i + \alpha_i C_{i+1} = \beta_i \,, \quad i = 1, \ldots, N - 1 \,, \tag{D.4}$$

or, equivalently,

$$C_{i-1} + \alpha_{i-1} C_i = \beta_{i-1} \,, \quad i = 2, \ldots, N. \tag{D 5}$$

Thus knowledge of α_i, β_i and C_i allows previous values C_{i-1} to be calculated in a sweep back through the system. Inserting the form (D 5) into (D.2) removes C_{i-1} to give

$$C_i + \frac{c_i}{b_i - a_i \alpha_{i-1}} C_{i+1} = \frac{d_i - a_i \beta_{i-1}}{b_i - a_i \alpha_{i-1}} \,, \tag{D 6}$$

for $i = 2, \ldots, N$. Comparing (D 4) with (D 6) gives the relations

$$\alpha_i = \frac{c_i}{b_i - a_i \alpha_{i-1}} \,, \qquad \beta_i = \frac{d_i - a_i \beta_{i-1}}{b_i - a_i \alpha_{i-1}} \,, \tag{D.7}$$

so that α_i and β_i can be calculated iteratively in a sweep forwards through the system. The initial values α_1 and β_1 are found by comparing (D 3) to (D.4) with $i = 1$ so that

$$\alpha_1 = \frac{c_1}{b_1} \,, \qquad \beta_1 = \frac{d_1}{b_1} \,. \tag{D.8}$$

When $i = N$, (D.1) and (D 7) give $\alpha_N = 0$ so that

$$C_N = \beta_N \,, \tag{D 9}$$

and thus the final value C_N seeds the backwards sweep

D.3 Summary

The process of solving a tri-diagonal system involves the steps·

- Calculate α_1 and β_1 using (D 8).
- Sweep forwards, calculating α_i and β_i for $i = 2, \ldots, N$ with (D 7).
- Calculate C_N using (D.9)
- Sweep backwards, calculating C_i for $i = N-1, \ldots, 1$ with the relation (D.4).

Bibliography

[1] Abramowitz, M. & Stegun, I. A. 1965 *Handbook of Mathematical Functions*, New York Dover.

[2] Baker, T H 1994 *Symmetric Functions and Infinite Dimensional Algebras*, Ph D. Thesis, Univ. of Tasmania

[3] Beam, R M & Warming, R. F. 1978 An implicit factored scheme for the compressible Navier-Stokes equations. *AIAA J.* **16**, 393–402

[4] Bickley, W G. 1941 Formulae for numerical differentiation. *Math Gaz* **25**, 19–27.

[5] Bowen, M. K. & Smith, R. 2005a Derivative formulas and errors for non-uniformly spaced points *Proc. Roy. Soc. Lond A In the press*

[6] Bowen, M. K. & Smith, R. 2005b Structure and accuracy of alternating direction implicit schemes *Unpublished*

[7] Corless, R. M. & Rokicki, J. 1996 The symbolic generation of finite-difference formulas *Z A M M.* **76**, 381–382

[8] Cox, S. M & Matthews, P. C. 2002 Exponential time differencing for stiff systems. *J. Comput. Phys.* **176**, 430–455

[9] Crandall, S. H. 1955 An optimum implicit recurrence formula for the heat conduction equation. *Quart. Appl. Math* **13, 3**, 318–320

[10] Crank, J. & Nicolson, P. 1947 A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type *Proc. Camb. Philos Soc.* **43**, 50–67.

[11] De Marchi, S 2001 Polynomials arising in factoring generalized Vandermonde determinants an algorithm for computing their coefficients. *Math. Comput. Modelling* **34**, 271–281

[12] Douglas, J. 1955 On the numerical integration of $u_{xx} + u_{yy} = u_t$ by implicit methods. *J. Soc. Indust. Appl. Math.* **3**, 42–65.

[13] Evans, D. J. 1997 The parallel solution of tridiagonal systems by recursive striding. *Para. Alg. & App.* **10**, 161–164

[14] Feng, B F. & Wei, G. W 2002 A comparison of the spectral and the discrete singular convolution schemes for the KdV-type equations. *J. Comput. Appl. Math* **145**, 183–188.

[15] Fornberg, B. 1988 Generation of finite difference formulas on arbitrarily spaced grids *Math. Comp* **51, 184**, 699–706

[16] Fornberg, B. 1998 Calculation of weights in finite difference formulas *SIAM* **40, 3**, 685–691.

[17] Grimshaw, R. 2005 Korteweg-de Vries equation *Encyclopedia on Nonlinear Science*, edited by A C. Scott. (to appear).

[18] Korteweg, D. J. & de Vries, G. 1895 On the change of form of long waves advancing in a rectangular canal and on a new type of long stationary wave *Philosophical Magazine* **39**, 422–443

[19] Ma, H P. & Sun, W. W. 2000 A Legendre-Petrov-Galerkin and Chebyshev collocation method for third-order differential equations. *SIAM J. Num. Anal.* **38**, 1425–1438.

[20] MacDonald, I G 1995 *Symmetric Functions and Hall Polynomials*, 2nd ed Oxford Science Publications

[21] Marchant, T. R & Smyth, N. F. 2002 The initial boundary problem for the Korteweg-de Vries equation on the negative quarter-plane. *Proc. Roy Soc. Lond A* **458**, 857–871.

[22] McKee, S & Mitchell, A. R. 1970 Alternating direction methods for parabolic equations in two space dimensions with a mixed derivative *The Computer Journal* **13**, 81–86.

[23] McKee, S , Wall, D. P & Wilson, S K. 1996 An alternating direction implicit scheme for parabolic equations with mixed derivative and convective terms. *J Comput. Phys.* **126**, 64–76

[24] Mitchell, A R. & Fairweather, G 1964 Improved forms of the alternating direction methods of Douglas, Peaceman and Rachford for solving parabolic and elliptic equations. *Numer. Math.* **6**, 285–292

[25] Mitchell, A. R. & Griffiths, D. F 1980 *The Finite Difference Method in Partial Differential Equations*, New York: Wiley

[26] Peaceman, D W & Rachford, H. H 1955 The numerical solution of parabolic and elliptic differential equations. *J. Soc. Indust Appl Math* **3, 1**, 28–41.

[27] Richtmyer, R. D. & Morton, K. W. 1967 *Difference methods for initial-value problems*, 2nd ed. New York. Wiley.

[28] Saul'ev, V. K. 1958 On methods of increased accuracy in two-sided approximations to the solution of parabolic equations. *Doklady Akad Nauk USSR* **118**, 1088–1090.

[29] Sebben, S. & Baliga, B. R 1995 Some extensions of tridiagonal and pentadiagonal matrix algorithms, *Numerical Heat Transfer, Part B* **28**, 323–351.

[30] Smith, R. 1999 Optimal and near-optimal advection-diffusion finite-difference schemes I. Constant coefficient in one dimension. *Proc. Roy. Soc. Lond A* **455**, 2371–2387

[31] Smith, R 2000 Optimal and near-optimal advection-diffusion finite-difference schemes II. Unsteadiness and non-uniform grid *Proc Roy. Soc. Lond. A* **456**, 489–502

[32] Smith, R. & Bowen, M. K. 2003 Optimal and near-optimal advection-diffusion finite-difference schemes. VIII Kay benchmark problem *Unpublished.*

[33] Smith, R & Bowen, M. K 2005 Compact schemes for evolution equations. *Unpublished*

[34] Smith, R. & Tang, Y. 2001 Optimal and near-optimal advection-diffusion finite-difference schemes VI. 2-D alternating directions. *Proc Roy Soc. Lond. A* **457**, 2379–2396.

[35] Soliman, A A. 2004 Collocation solution of the Korteweg-de Vries equation using septic splines *Int. J. Comput. Math.* **81**, **3**, 325–331.

[36] Spotz, W. F. & Carey, G F. 2001 Extension of high order compact schemes to time dependent problems. *Numer. Meth. PDEs* **17**, 657–672.

[37] Stone, H. S. 1973 An efficient parallel algorithm for the solution of a tridiagonal linear system of equations *J. ACM* **20**, **1**, 27–38.

[38] Yan, J & Shu, C W. 2002 A local discontinuous Galerkin method for KdV type equations. *SIAM J. Num Anal.* **40**, 769–791.