Finite differences for the convection-diffusion equation:

On stability and boundary conditions

Ercília Sousa
St. John's College

Thesis submitted for the degree of Doctor of Philosophy
at the University of Oxford

Trinity Term 2001

Abstract

Finite differences for the convection-diffusion equation:
On stability and boundary conditions

Ercília Sousa
St John's College

Doctor of Philosophy
Trinity Term 2001

The solution of convection-diffusion problems is a challenging task for numerical methods because of the nature of the governing equation, which includes a non-dissipative component and a dissipative component. Once the convection-diffusion equation is discretised, it is usual to observe oscillations in the computed solution regardless of whether these might be expected in the original physical situation. Mostly these oscillations are the result of numerical instability. This thesis centres on this fundamental difficulty: the numerical stability of finite difference discretisation of a convection-diffusion equation.

The existence of an exact evolution operator for the constant coefficient convection diffusion problem is the framework we use to derive new finite difference schemes in one and two dimensions and also, when a high-order scheme is considered, to derive numerical boundary conditions. The influence of numerical boundary conditions on the stability of a general scheme is one of the main themes. The stability analysis is done mostly by using the von Neumann method and the matrix method. The Godunov-Ryabenkii theory is also applied to the one dimensional case.

In two dimensions we deduce different forms of second-order (Lax-Wendroff) schemes and third-order (Quickest) schemes. We apply some of those schemes to a Navier-Stokes problem by running experiments to illustrate the practical stability region, showing how results from a simpler case presented in previous chapters carry over to the more complex case.

Contents

Acknowledgements

The writing of this dissertation was accomplished with the help and inspiration of a number of people to which I am deeply indebted.

My deepest gratitude is to my supervisor Ian Sobey, for his many valuable insights into a great variety of problems and for his stimulation, encouragement and permanent support since my first day in Oxford.

I would like to thank Professor Nick Trefethen and Professor Mike Giles. Nick Trefethen directed my attention to the Godunov-Ryabenkii theory, giving me many valuable comments on the fourth chapter. Mike Giles was of an enormous help in the development of the algorithm in the same chapter. I am very grateful to both of them for the extensive and enjoyable discussions on normal mode analysis, GKS theory and Godunov-Ryabenkii theory that all three of us had together.

Many thanks go to my colleagues at the Oxford University Computing Laboratory, to Álvaro Meseguer for the exciting discussions about Navier-Stokes equations and bifurcations, to Nick Birkett for his unlimited patience with me and my laptop and to Pierre Moinier and all the others.

Special thanks go to my friend Guy Kahane for saving me from numerous English mistakes.

I would also like to acknowledge my debt to the person with whom I first started to explore the fascinating field of Numerical Analysis, Professor Paula Oliveira from Coimbra University. She was one of the persons who made my research in Oxford possible.

Lastly, I am grateful to Professor Endre Süli and Professor Ian Castro for agreeing to be my examiners and for their comments regarding this dissertation.

My work on this dissertation was financially supported by Coimbra University and Fundação para a Ciência e Tecnologia in Portugal.

Chapter 1

Introduction

1.1 Purpose

Many physical problems involve the combination of convective and diffusive processes. They occur in fields where mathematical modelling is important such as physics, engineering and particularly in fluid dynamics and transport problems. A very large literature has been built up describing numerical approximations, as well as the various techniques for analysing and overcoming the difficulties that each numerical method presents. Extensive descriptions of different numerical methods for the convection-diffusion equation can be found for instance in Morton [50] and Roos *et al* [67].

The solution of convection-diffusion problems is a challenging task for numerical methods because of the nature of the governing equation, which includes a non-dissipative component and a dissipative component whose effects run over very different length and time scales. The relative influence of the two effects is described by a Péclet number. The Péclet number also determines the nature of the equation. For diffusion-dominated processes, where the Péclet number is low, the equation is best characterised as elliptic or parabolic. For convective dominated problems, where the Péclet number is large, the equation is best characterised as hyperbolic.

In the case of convection-diffusion of a dissolved material, the physical solutions usually decay monotonically to a uniform state. In the case of a fluid flow, where a non-uniform convection-diffusion equation describes conservation of momentum, solutions can be largely decaying towards a steady state. However, it is possible for real physical oscillations to occur for a set of initial and boundary conditions that might indicate a steady solution.

Once the convection-diffusion equation is discretised, it is common to observe oscillations in the computed solution regardless of whether these might be expected in the original physical situation. These oscillations are often the result of either inaccurate convection of different Fourier modes or numerical instability.

Numerical solutions of the convection-diffusion equation often show spatial numerical oscillations which persist in steady state. In practical calculations, many authors have observed spurious oscillations in the solution as the Péclet number is increased, a phenomenon common to finite difference methods when central rather than upwind differencing of the convective term is used (see e.g. Gresho and Lee [23] and Roache [65]). On the other hand, upwind differencing usually introduces unpleasant artificial numerical diffusion.

This dilemma is central to numerical solutions of convection-diffusion problems. Oscillations may or may not have a physical basis, oscillatory computations may or may not reflect properties of a discretisation rather than a physical situation. Artificial damping of numerical oscillations is inevitably dispersive, destroying the ability of the numerical solution to represent the original physical process. This thesis centres on this fundamental difficulty: the numerical stability of finite difference discretisation of a convection-diffusion equation.

In order to compute approximate solutions for evolutionary partial differential equations with either explicit or implicit schemes, it is necessary to use some form of local approximation; local in the sense that solution values at local nodes are used to generate an approximate solution value at a new time level. In finite differences it is usual to try to make the local domain as compact as possible, for instance using only adjacent nodes when updating at a node. The domain of local approximation may need to be large because the degree of the equation is high or it may need to be enlarged to accommodate a higher order local approximation for a low order differential equation. In either case schemes are usually derived for infinite space domains; when space boundaries occur they prevent such high order schemes from being applied directly. One method to deal with the second of these situations, a high order local approximation to a low order equation, is simply to use a lower order scheme immediately near boundaries and use a high order scheme for the major part of the interior of the domain. Whether this is useful will depend on the nature of the problem being approximated. If interest is centred on dynamics in the interior and on time scales where boundary effects have not propagated to the region of interest, then this will be a reasonable approximation. If the boundary influences the interior quickly, little may be gained by using a high order scheme to accurately propa-

gate low order errors from the boundaries to the interior. There will always be special problem formulations where boundary conditions can be used to provide a modified scheme at a boundary which retains the accuracy of the scheme used in the interior of the domain; in the majority of cases such techniques will either not be possible or not be satisfactory at nodes close to a boundary.

If we consider the approximation of the unsteady convection-diffusion problem on a uniform mesh, we can observe that much of the literature is concerned with choices between three-point schemes that focus attention on the nodal value at j and the two neighbouring values at $j-1$ and $j+1$; however the emphasis on what happens in the cell $[x_{j-1}, x_j]$ or the cell $[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$, which is at the heart of some upwind finite difference schemes, leads naturally, for the convection with velocity in the x-direction, to four-point schemes centred on the cell and using the nodal values at $j-2$, $j-1$, j and $j+1$. Recent work on finite difference schemes for the convection-diffusion equation can be found in papers such as Bruneau *et al* [8], Douglas *et al* [14], and Kurganov and Tadmor [36].

Finite difference schemes typically consist of replacement of the individual derivative terms in the partial differential equation by a set of discretised approximations (see e.g. Smith [72]). However, recently different techniques were suggested for deriving finite differences for the unsteady convection-diffusion equation (see e.g. Morton and Sobey [49] and Xu *et al* [94]).

Morton and Sobey [49] studied discretisation of a convection-diffusion equation on the whole line by using an exact evolution solution. If the function $u(x,t)$ evolves according to

$$\frac{\partial u}{\partial t} + \mathcal{L}[u] = 0,$$

then the exact solution can be written

$$u(.,t+\Delta t) = E(\Delta t)u(.,t),$$

where $E(\Delta t)$ is an evolution operator. If U^n represents an approximate solution at $t = n\Delta t$ and P a projection operator onto the approximation space, then an approximate solution is obtained from

$$U^{n+1} = PE(\Delta t)U^n.$$

This also allowed a very natural error analysis. In the case of finite differences, let $U^n = I_p\{U_j^n\}$ where U_j^n are nodal values and I_p a local approximation based on nodal values; suppose R is a restriction operator onto the nodes, then as the exact solution satisfies $u^{n+1} = E(\Delta t)u^n$, the local nodal error satisfies

$$Ru^{n+1} - RU^{n+1} = RE(\Delta t)u^n - RE(\Delta t)U^n,$$

which could be rewritten,

$$Ru^{n+1} - RU^{n+1} = RE(\Delta t)(u^n - I_p\{u_j^n\}) + RE(\Delta t)I_p[Ru^n - RU^n].$$

The first term is the evolution of a local approximation error based on exact initial data, the second term is the evolution at the local approximation for the local nodal error. In the case of finite elements Morton and Sobey [49] introduced an approximate evolutionary operator E_Δ such as $U^{n+1} = E_\Delta U^n$ and

$$Pu^{n+1} - U^{n+1} = [PE(\Delta t) - E_\Delta P]u^n + E_\Delta[Pu^n - U^n],$$

so that the error could be analysed as the sum of an error in the evolution operator and the evolution of a local approximation error. This general framework centres around a knowledge of the exact evolution operator E and it is of course applicable regardless of the number of space dimensions or the operator $\mathcal{L}$.

In this dissertation we use the framework described above to deduce new finite difference schemes and numerical boundary conditions for high order schemes. Although such a procedure cannot easily be generalised to partial differential equations with variable coefficients, the finite difference schemes obtained in this dissertation using the evolution operator can still be applied to those cases as we show in the last chapters.

Related with the convergence of a finite difference scheme we encounter questions about stability and accuracy. Here, we focus our attention on the study of the stability of the schemes considered either in an unbounded domain or in a bounded domain. The influence of numerical boundary conditions on the stability of the general scheme is also one of the main themes of this dissertation.

Stability of finite difference schemes has been widely described in the literature. Two important books on stability analysis of difference methods are the classical book by Richtmyer and Morton [62] and the more recent book by Gustafsson *et al* [27]. Some of the work on stability analysis for finite difference schemes for the convection-diffusion equation using von Neumann method and matrix analysis was done by Beckers [5], Chan [11], Griffiths *et al* [26], Hindmarsh *et al* [29], Hirt [31], Kwok and Tan [37], Leonard [42, 45], Morton [48], Siemieniuch and Gladwell [70], Verwer and Sommeijer [88], Warming and Hyett [91], and Wesseling [92].

The motivation for our study of the convection-diffusion equation also has to do with the two-dimensional Navier-Stokes equations. The alternative form of the Navier-Stokes equations that is obtained with the vorticity and the stream function formulation has been applied extensively. We are studying the convec-

tion-diffusion equation not only because of its own importance but also in order to understand some aspects of the vorticity equation.

1.2 Outline of the dissertation

We now give a description of the remaining chapters of this dissertation.

Chapter 2. We begin with a study of a one-dimensional unsteady convection-diffusion problem. We describe how in Morton and Sobey [49] Lax-Wendroff and Quickest methods were re-derived using an evolution operator. The Lax-Wendroff scheme is a scheme due to Lax and Wendroff [38, 39], which is a second order accurate scheme and consists of the combination of time and space-centred discretisations. The Quickest scheme was introduced as an alternative to central differencing convection and to upwinding differencing convection. This scheme was first proposed by Leonard [41], using control volume arguments.

In this chapter we use the same framework as in Morton and Sobey [49] to derive, in a new way, the Allen-Southwell operator. One section of this chapter is devoted to the stability analysis of the Lax-Wendroff scheme and Quickest scheme using the von Neumann method. In the last section we consider the truncation error of both Lax-Wendroff and Quickest schemes.

Chapter 3. We consider a model problem for the one-dimensional convection-diffusion equation with a left physical boundary condition and discuss different choices of numerical boundary conditions and their effects on the stability and accuracy of the resulting numerical schemes. These results were presented in Sousa and Sobey [76]. The numerical boundary conditions are derived by using the evolutionary operator for the unsteady convection-diffusion equation in an unbounded domain.

The schemes are written as a system of equations. The iterative matrix contains the numerical boundary treatment, and the stability is investigated through its eigenvalue spectrum and norm, without disregarding the stability condition given by the von Neumann method. Furthermore, this framework is used in subsequent chapters to investigate the stability of finite difference schemes for model problems in two dimensions with constant flow velocity and non-uniform flow velocity.

Chapter 4. We give a brief description of the Godunov-Ryabenkii theory that

was initially presented by Godunov and Ryabenkii [21] and later developed by Kreiss [35] and Osher [55]. The original Godunov-Ryabenkii theory provided a necessary condition for stability. A necessary and sufficient condition for stability was later developed by Gustafsson, Kreiss and Sundstrom [28], henceforth called GKS theory. This theory covers linear, first order hyperbolic systems in one space dimension. Since 1971, when the GKS theory was first presented, related work has been done by Varah [90] for parabolic problems, by Strikwerda [78] for semi-discretised equations, by Michelson [47] for multidimensional problems and by Trefethen [84, 85] where a relation between the GKS theory and group velocity is established.

We are interested in parabolic problems with convective and diffusive coefficients. The GKS theory that leads to necessary and sufficient conditions for stability was proven for hyperbolic problems. For parabolic problems we can apply the Godunov-Ryabenkii method which theoretically will give us only necessary conditions for stability although in a number of cases these appear to be also sufficient conditions.

We analytically apply the Godunov-Ryabenkii theory to the Quickest scheme with a particular numerical boundary condition which was also described in Sousa [75]. We also prove some new properties related to this theory and develop a new algorithm that can be applied successfully to the Quickest scheme with some numerical boundary conditions.

Chapter 5. We derive new numerical schemes using an exact solution for an initial value problem with an inflow boundary in one dimension and consider the numerical boundary conditions which are required. These numerical boundary conditions are deduced using the evolutionary operator associated with the problem in a domain with a left physical boundary instead of using the evolutionary operator in the unbounded domain as in the third chapter. We also discuss if there are advantages for stability and accuracy in using these new schemes, whether when compared with the Lax-Wendroff scheme and the Quickest scheme discussed previously. Some of these results are also in Sousa and Sobey [74].

Chapter 6. We develop a family of Lax-Wendroff schemes and Quickest schemes using the evolutionary operator for the two dimensional convection-diffusion problem in an unbounded domain. Although the Lax-Wendroff scheme has only one variant for the one-dimensional case, it has many variants in the two-dimensional case. The ambiguity in two dimensions is connected with the fact that different combinations of local nodal values are equally able to model

local behaviour with the same order of accuracy. In this chapter, we deduce the analytical solution for a two dimensional convection-diffusion problem and use it as the source for obtaining Lax-Wendroff and Quickest schemes not yet studied in the literature. The Quickest scheme was first generalised in two dimensions by Davis and Moore [13]. When generalising the method they ignored some of the cross-derivatives and that reduced the temporal accuracy of the scheme. The new Quickest schemes are expected to be more accurate in time than the Quickest scheme deduced by Davis and Moore [13], since we take into account the cross-derivatives. Additionally we study in detail the stability of those Lax-Wendroff and Quickest schemes, a crucial property for convergence of numerical schemes.

To analyse the practical stability of the numerical schemes we use von Neumann analysis since we are considering a problem in an unbounded domain. We observe that interesting differences occur between the stability regions of the different numerical schemes. For a clear visualisation of the stability regions we plot the sufficient and necessary stability conditions in a three-dimensional space, in which the coordinates involve convection and diffusion coefficients.

Chapter 7. In the previous chapter we applied the von Neumann method to analyse the stability of the Lax-Wendroff schemes and Quickest schemes in a two-dimensional unbounded domain. In this chapter, similarly to what we have done in the third chapter for a one-dimensional problem, we provide numerical boundary conditions for high order Quickest schemes and examine their effect on the stability of the general scheme. We investigate the stability regions for the different Lax-Wendroff schemes and Quickest schemes with the numerical boundary conditions when applied to two problems: first, we consider a convection-diffusion problem in a quarter plane where we assume both convective coefficients to be positive and constant and secondly, we consider a problem defined in a square with non-uniform flow velocity.

Chapter 8. In this chapter a numerical simulation of the unsteady incompressible flow in the unit cavity is performed. We consider the formulation of the Navier-Stokes equations in terms of the vorticity and the streamline function and approximate the vorticity equation using what we called the Taylor Lax-Wendroff scheme and the Taylor Quickest scheme. For the high order Taylor Quickest scheme near the boundary we consider the numerical boundary conditions discussed in the foregoing chapter. We perform experiments to illustrate the practical stability for the Navier-Stokes problem and show how results from

the simpler case presented in the previous chapter carry over to a more complex case.

We close this chapter by giving global iteration matrices for the stream-function vorticity formulation for the Navier-Stokes equations, showing that the true time marching iteration matrix is more complicated than the iteration matrix for the convection-diffusion equation that is part of the Navier-Stokes equations. Indeed, by examining a one-dimensional test problem it is shown that the full system has much tighter stability constraints than would have been predicted from the convection-diffusion equation alone.

Chapter 9. We draw some conclusions and describe some open problems which originated from our research.

Chapter 2

Convection-diffusion in one dimension

We consider in this chapter a one-dimensional unsteady convection-diffusion problem on the whole real line. For this problem, we first demonstrate how the Lax-Wendroff scheme and the Quickest scheme are derived in Morton and Sobey [49] using an evolution operator. We describe this in detail, since the same framework is used later in this chapter to derive the Allen and Southwell operator, and in subsequent chapters to derive numerical boundary conditions and new finite difference schemes.

Next, we present the stability results and the truncation errors for the aforementioned schemes. Since this is a convection-diffusion problem in an unbounded domain, we can use the von Neumann method to investigate stability. Although the necessary and sufficient stability conditions for the Lax-Wendroff scheme are quite well known, we derive them here to illustrate the general idea of the von Neumann stability analysis. On the other hand, the analytical von Neumann necessary and sufficient conditions that we derive for the Quickest scheme have not been previously stated in the literature, although they have been computed numerically.

2.1 Introduction

Consider the one-dimensional problem of convection with constant velocity V in the positive x direction and constant diffusion with coefficient $D > 0$:

$$\frac{\partial u}{\partial t} + V\frac{\partial u}{\partial x} = D\frac{\partial^2 u}{\partial x^2}, \quad t > 0, \quad x \in \mathbb{R}. \tag{2.1}$$

If we choose a uniform space step Δx and time step Δt let:

$$\mu = \frac{D\Delta t}{(\Delta x)^2}, \quad \nu = \frac{V\Delta t}{\Delta x};$$

ν is called the Courant (CFL) number.

When convection is dominant it is natural to make use of the method, introduced by Lax and Wendroff [39], which considers a Taylor expansion

$$u(x, t + \Delta t) \approx u(x, t) + \Delta t\frac{\partial u}{\partial t} + \frac{\Delta t^2}{2}\frac{\partial^2 u}{\partial t^2} + \ldots \tag{2.2}$$

and then uses the equation (2.1) to replace the temporal derivatives with spatial ones. Whereas simple upwinding results in a scheme which is only first order accurate the Lax-Wendroff scheme is second order accurate. Leonard [41], using control volume arguments, proposed a scheme called Quickest (Quadratic Upstream Interpolation for Convective Kinematics with Estimated Streaming Terms) which is explicit and third order accurate in time in the limit $D \to 0$; Davis and Moore [13] have shown that Quickest can also be derived by considering the Δt^3 term in expansion (2.2) and making some subsequent approximations.

Quickest uses an explicit, Leith-type differencing and third-order upwinding on the convective derivatives yielding a four-point scheme. The use of third-order upwind differencing for convection greatly reduces the numerical diffusion associated with first-order upwinding. This is illustrated in a paper by Baum *et al* [4]. The major difficulties associated with the use of Quickest scheme in multidimensions are in the application of boundary conditions.

Quickest was introduced as an alternative to central differencing convection and to upwinding differencing convection. The first one is usually associated with high-convection instability problems and wiggles, and the second one with the fact that stability is reached by introducing unpleasant artificial numerical diffusion terms. Quickest is an improvement of these two standard methods for unsteady primarily convective flows.

Some of the literature about Quickest used in a flow simulation can be found in Baum *et al* [4], Davis and Moore [13] and Leonard [41]. Other references concerning the quasi-steady flow and the method called Quick which is the equivalent method to Quickest for quasi-steady flows can be found for instance in Johnson and Mackinnon [34], Leonard [43] and Leonard and Mokhtari [44].

Our attention, in this dissertation, is concentrated in a high-order scheme derived using the evolution operator associated with the convection-diffusion problem considered, although there are other high order schemes in literature that could be considered, see for instance Castro and Jones [10] which compares the Quick scheme with a scheme they called HODS. The HODS scheme is considered briefly in chapter 8.

2.2 Lax-Wendroff scheme and Quickest scheme

In this section we describe the derivation in Morton and Sobey [49] of Lax-Wendroff and Quickest methods, for the one-dimensional convection-diffusion equation in an infinite domain, using an evolution operator.

Consider the convection-diffusion equation (2.1), subject to the initial condition

$$u(x,0) = u_0(x) \tag{2.3}$$

and the boundary condition

$$u(x,t) = 0 \quad \text{as} \quad |x| \to \infty. \tag{2.4}$$

This initial value problem can be solved in closed form using Fourier transforms in x to obtain the exact solution,

$$u(x,t) = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{+\infty} u_0(x - Vt + 2\sqrt{Dt}\xi)\mathrm{e}^{-\xi^2} d\xi. \tag{2.5}$$

Applying the result to evolution over one time step, from time t_n to $t_{n+1} = t_n + \Delta t$ write

$$u(x, t_n + \Delta t) = \int_{-\infty}^{+\infty} u(\eta, t_n) G(x - \eta; \Delta t) d\eta,$$

where the Green's function is given by

$$G(z;\tau) = \frac{1}{\sqrt{D\pi\tau}} \mathrm{e}^{-(z-V\tau)^2/4D\tau}.$$

As showed by Morton and Sobey [49] to derive either finite difference or finite element approximations we substitute an approximation to $u(\eta, t_n)$ in this integral, and exploit the fact that the integration of a global polynomial can be carried out exactly.

We now focus on the finite difference case. Suppose we have approximations U_j^n to the values $u(x_j, t_n)$ at the mesh points

$$x_j = j\Delta x, \quad j = 0, \pm 1, \pm 2, \ldots;$$

for this set of values we denote $\mathcal{U}^n := \{U_j^n\}$. We also denote $p_j(x;\mathcal{U}^n)$ the interpolating polynomial, associated with the points x_j, through U_j^n and the values at a certain number of neighbouring points. Then finite difference schemes can be generated from evolution of this interpolating approximation by

$$U_j^{n+1} = \int_{-\infty}^{+\infty} p_j(\eta;\mathcal{U}^n) G(x_j - \eta; \Delta t) d\eta. \tag{2.6}$$

If the approximation scheme obtained comes from approximating $\mathcal{U}^n$ near x_j by a polynomial $p_j(x;\mathcal{U}^n)$, of degree R,

$$p_j(x;\mathcal{U}^n) = \sum_{r=0}^{R} b_{jr}(x - x_j)^r,$$

then

$$U_j^{n+1} = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{+\infty} p_j(x - V\Delta t + 2\sqrt{D\Delta t}\xi;\mathcal{U}^n) \mathrm{e}^{-\xi^2} d\xi.$$

When evaluating the previous integral we come across integrals of the form

$$a_r = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{+\infty} \xi^r \mathrm{e}^{-\xi^2} d\xi.$$

For $r = 1, 2, \ldots$ then

$$a_r = \begin{cases} 1 & r = 0 \\ \dfrac{r-1}{r} & r \quad \text{even} \\ 0 & r \quad \text{odd} \end{cases}$$

The approximate solution can be written

$$\begin{aligned} U_j^{n+1} = b_{j0} - b_{j1}V\Delta t + b_{j2}[V^2(\Delta t)^2 + 2D\Delta t] - b_{j3}[V^3(\Delta t)^3 + 6VD(\Delta t)^2] \\ + b_{j4}[V^4(\Delta t)^4 + 12V^2D(\Delta t)^3 + 12D^2(\Delta t)^2] + \ldots. \end{aligned}$$

Within this general framework one can obtain Lax-Wendroff and Quickest schemes by different interpolation schemes on a uniform mesh. We use the usual central, backward and second difference operators,

$$\Delta_0 U_j := \frac{1}{2}(U_{j+1} - U_{j-1}), \quad \Delta_- U_j := U_j - U_{j-1}, \quad \text{and} \quad \delta^2 U_j := U_{j+1} - 2U_j + U_{j-1}$$

to evaluate the coefficients b_{jr} in terms of the nodal values $\mathcal{U}^n$.

Quadratic interpolation - Lax-Wendroff

If a quadratic interpolant of U_{j-1}, U_j and U_{j+1} is used then

$$b_{j0} = U_j^n, \quad b_{j1} = \frac{\Delta_0 U_j^n}{\Delta x}, \quad b_{j2} = \frac{\delta^2 U_j^n}{2\Delta x^2},$$

and the approximation formula for U_j^{n+1} becomes the Lax-Wendroff scheme

$$U_j^{n+1} = [1 - \nu\Delta_0 + (\frac{1}{2}\nu^2 + \mu)\delta^2]U_j^n. \tag{2.7}$$

Cubic approximation - Quickest

If $p_j(x, \mathcal{U}^n)$ is extended to include a cubic term, then there would be a choice of points which can be interpolated. If the cubic expansion is obtained by interpolating U_{j-2}^n as well as U_{j-1}^n, U_j^n and U_{j+1}^n, then

$$b_{j0} = U_j^n, \quad b_{j1} = \frac{\Delta_0 U_j^n}{\Delta x} - \frac{\delta^2 \Delta_- U_j^n}{6\Delta x^3}, \quad b_{j2} = \frac{\delta^2 U_j^n}{2\Delta x^2}, \quad b_{j3} = \frac{\delta^2 \Delta_- U_j^n}{6\Delta x^3},$$

and the approximation formula becomes the Quickest scheme

$$U_j^{n+1} = [1 - \nu\Delta_0 + (\frac{1}{2}\nu^2 + \mu)\delta^2 + \nu(\frac{1}{6} - \frac{\nu^2}{6} - \mu)\delta^2\Delta_-]U_j^n. \tag{2.8}$$

We have an understanding of how schemes which previously might be classified as finite element or finite difference methods can be reconciled in a relatively general evolutionary operator framework as presented in Morton and Sobey [49]. We have described how Morton and Sobey deduced the Lax-Wendroff and Quickest finite difference schemes using an evolutionary operator. In the next section we show how this framework can also be used to derive the Allen-Southwell operator.

2.3 The Allen and Southwell operator

A different scheme for approximating the convection-diffusion operator was used by Allen and Southwell [1] and later independently by Il'in [33]. This different approach led to what have come to be called exponentially fitted schemes, and the discrete operator that was used by Allen and Southwell to approximate the

convection-diffusion operator was henceforth called the Allen Southwell operator. A paper written by Roos [66] is also worth mentioning, since it discusses different ways of generating schemes that are related to the Allen and Southwell operator.

In this section we show how to deduce the Allen and Southwell operator in a different manner from the above authors, by using an evolutionary operator.

The Allen and Southwell operator was obtained for the steady case. Here we consider the unsteady convection-diffusion equation

$$\epsilon\frac{\partial u}{\partial t} + V\frac{\partial u}{\partial x} = D\frac{\partial u}{\partial x^2},$$

and then from this equation we can obtain the steady equation by letting $\epsilon \to 0$. We obtain a slightly changed expression for the solution of the convection-diffusion equation given by (2.5). Denoting $h = \Delta x$, we have

$$u(0,t) = \frac{1}{\sqrt{\pi}}\int_{-\infty}^{+\infty} u(xh,0)\mathrm{e}^{-(x+c)^2/\gamma}\frac{dx}{\sqrt{\gamma}} \tag{2.9}$$

where $c = Vt/\epsilon h$ and $\gamma = 4Dt/\epsilon h^2$. Now we use a Taylor expansion around $u_0 = u(0,0)$,

$$u(xh,0) \approx u_0 + xh[\frac{(1+\alpha)}{h}\Delta_0 - \frac{\alpha}{h}\Delta_-]u_0 + \frac{(xh)^2}{2h^2}\delta^2 u_0, \quad -1 < x < 1, \tag{2.10}$$

where α is the weighting factor, giving a weighted average of upwind and central differencing to model the first derivative. Substituting (2.10) into (2.9), if we integrate in $\mathbb{R}$, for $\alpha = 0$ we have the Lax-Wendroff. If we integrate in $[-1,1]$ we need to calculate

$$\int_{-1}^{1}\frac{x}{\sqrt{\gamma}}\mathrm{e}^{\frac{-(x+c)^2}{\gamma}}dx \qquad \text{and} \qquad \int_{-1}^{1}\frac{x^2}{\sqrt{\gamma}}\mathrm{e}^{\frac{-(x+c)^2}{\gamma}}dx.$$

Let

$$\beta = \frac{Vh}{D}, \quad \theta = \frac{1+c^2}{\gamma} \quad \text{and} \quad S_0 = \frac{1}{\sqrt{\gamma}}\int_{-1}^{1}\mathrm{e}^{-(x+c)^2/\gamma}dx.$$

Then,

$$\begin{aligned}
\int_{-1}^{1}\frac{x}{\sqrt{\gamma}}\mathrm{e}^{\frac{-(x+c)^2}{\gamma}}dx &= \sqrt{\gamma}\mathrm{e}^{-\theta}\sinh(\beta/2) - cS_0 \\
\int_{-1}^{1}\frac{x^2}{\sqrt{\gamma}}\mathrm{e}^{\frac{-(x+c)^2}{\gamma}}dx &= -\sqrt{\gamma}\mathrm{e}^{-\theta}\cosh(\beta/2) \\
&\quad + \frac{S_0\gamma}{2} - c\sqrt{\gamma}\mathrm{e}^{-\theta}\sinh(\beta/2) + c^2S_0.
\end{aligned}$$

Since we want to study the transition between the unsteady case and the steady case, suppose that $u(0,t) = u_0$, for all $t \geq 0$. We obtain

$$\begin{aligned}\sqrt{\pi}u_0 &= S_0u_0 + (\sqrt{\gamma}e^{-\theta}\sinh(\beta/2) - cS_0)[(1+\alpha)\Delta_0 - \alpha\Delta_-]u_0 \\ &\quad +\frac{1}{2}[-\sqrt{\gamma}e^{-\theta}\cosh(\beta/2) + \frac{S_0\gamma}{2} - c\sqrt{\gamma}e^{-\theta}\sinh(\beta/2) + c^2S_0]\delta^2u_0.\end{aligned}$$

Since $\Delta_-u_0 = \Delta_0u_0 - \delta^2u_0/2$ then

$$\begin{aligned}(\sqrt{\pi} - S_0)u_0 &= (\sqrt{\gamma}e^{-\theta}\sinh(\beta/2) - cS_0)\Delta_0u_0 \\ &\quad +[\frac{1}{2}\alpha\sqrt{\gamma}e^{-\theta}\sinh(\beta/2) - c\frac{1}{2}\alpha S_0 \\ &\quad -\frac{1}{2}\sqrt{\gamma}e^{-\theta}\cosh(\beta/2) + \frac{S_0\gamma}{4} - c\frac{1}{2}\sqrt{\gamma}e^{-\theta}\sinh(\beta/2) \\ &\quad +\frac{1}{2}c^2S_0]\delta^2u_0.\end{aligned}$$

If $\alpha = c$ we can write

$$\begin{aligned}(\sqrt{\pi} - S_0)u_0 &= (\sqrt{\gamma}e^{-\theta}\sinh(\beta/2) - cS_0)\Delta_0u_0 \\ &\quad +[-\frac{1}{2}\sqrt{\gamma}e^{-\theta}\cosh(\beta/2) + \frac{S_0\gamma}{4}]\delta^2u_0,\end{aligned}$$

and for $\epsilon \to 0$ we have

$$\sqrt{\pi}u_0 = \Delta_0u_0 - \frac{1}{2}\coth(\beta/2)\delta^2u_0.$$

Consequently

$$\begin{aligned}\frac{V}{h}\sqrt{\pi}u_0 &= \frac{V}{h}\Delta_0u_0 - \frac{1}{2}\frac{V}{h}\coth(\beta/2)\delta^2u_0 \\ &\equiv \text{Allen Southwell operator}\end{aligned}$$

This shows how the known Allen and Southwell operator emerges in a different manner using an evolution operator for the unsteady convection-diffusion equation.

Stability analysis is one of the most important aspects of the study of finite difference schemes. The next section is devoted to the stability analysis of the Lax-Wendroff scheme and Quickest scheme. First we provide a brief introduction into the stability analysis of finite difference schemes in unbounded domains, and then we present the stability results for the aforementioned schemes.

2.4 Stability analysis

The problems that the finite differences deal with are mostly assumed to be properly posed from the outset, in such a way that existence and uniqueness of solutions are ensured under physically reasonable assumptions. The questions about stability and accuracy of a finite difference method are directly related to the convergence property of the numerical method as stated in the important and well known Lax Equivalence Theorem. Before we state the theorem we give first the definitions of stability and consistency.

Assume that the initial value problem (2.1), (2.3) is approximated by a finite difference scheme of the form

$$U_j^{n+1} = QU_j^n, \tag{2.11}$$

where $Q = \sum_{j=-r}^{p} a_j E^j, \quad EU_j^n = U_j^{n+1}.$

Definition 2.1: The finite difference method (2.11) is called stable in the norm $||\cdot||$ if there exist constants K and c such that

$$||U^n|| \leq K\mathrm{e}^{cn\Delta t}||U^0|| = K\mathrm{e}^{ct_n}||U^0||$$

where $t_n = n\Delta t$, and $K > 0$ and c are independent of the space steps and time step.

Definition 2.2: A finite difference scheme (2.11) is consistent up to time T_0 in the norm $||\cdot||$ with equation (2.1) if the actual solution u to the initial value problem (2.1), (2.3) satisfies

$$u_j^{n+1} = Qu_j^n + \Delta t T^n,$$

where $u_j^n = u(j\Delta x, n\Delta t)$, $||T^n|| \leq \tau(\Delta x)$, $n\Delta t < T_0$ and $\tau(\Delta x) \to 0$ as $\Delta t \to 0$. Here is assumed that Δx is defined in terms of Δt and goes to zero with Δt.

Theorem 2.1 *Lax Equivalence Theorem (see Richtmyer and Morton [62]): Given a properly posed initial-value problem for a linear partial differential equation and a linear finite difference approximation to it that satisfies the consistency condition, stability is the necessary and sufficient condition for convergence.*

The von Neumann (Fourier) method is the most well known classical method to determine necessary and sufficient stability conditions. Stability has been a

concern in the literature and is sometimes a controversial matter. To illustrate this, we turn to give an historical description of the stability analysis for the central scheme for the convection-diffusion equation.

2.4.1 The controversial stability analysis

Consider the space-centred, explicit Euler discretisation in one dimension:

$$U_j^{n+1} = U_j^n - \nu \Delta_0 U_j^n + \mu \delta^2 U_j^n. \tag{2.12}$$

The history of attempts to state stability conditions for the scheme (2.12) illustrates the difficulties in stability analysis as soon as a scheme becomes more complex. This historical review is illuminating with regard to the definition of stability.

The first controversy is associated with the von Neumann analysis. Historically, a first von Neumann stability condition was incorrectly derived in 1964 in a paper by Fromm [16]. Fromm used a von Neumann stability analysis technique to determine necessary stability limits on the two-dimensional vorticity transport equation. He mistakenly arrived at a conclusion, that in one dimension would be:

$$\text{(1964)} \qquad 0 \leq \nu \leq 2\mu \leq 1.$$

The incorrect concept of a mesh size limitation for stability, introduced by Fromm, has generated considerable confusion as has been shown by Thompson *et al* [81]. Roache [65] quoted the wrong result of Fromm [16] and helped to spread the misconception. The correct results were obtained initially in 1968 by Hirt [31], applying a different approach:

$$\text{(1968)} \qquad \nu^2 \leq 2\mu \leq 1.$$

In an investigation of the stability of the explicit central differenced convection-diffusion equation, (2.12), in 1978 Siemieniuch and Gladwell [70] applied a matrix method with a criterion on the spectral radius for stability. The conditions derived by Siemieniuch and Gladwell [70] are clearly distinct from the von Neumann conditions obtained by Hirt [31]. Siemieniuch and Gladwell [70] did not explain the discrepancies they observed between their derived stability limits

$$\text{(1978)} \qquad 0 \leq \mu \leq \frac{1}{\nu^2},$$

and the inaccuracy or instability of their computed results.

An increase of interest in this subject has generated a variety of publications for one and multi-dimensional stability analysis of the discretised convection-diffusion equation e.g. Chan [11]; Griffiths *et al* [26]; Hindamarsh *et al* [29]; Leonard [42]; Rigal [63, 64]; Thompson [81]. The theoretical explanation underlying these differences was provided by Morton [48] in 1980. The differences were mainly caused by the different definitions used for the stability of the numerical scheme and by the fact that some are only sufficient conditions and others only necessary conditions. We refer to the different ways of analysing stability in the following chapter.

2.4.2 Von Neumann stability analysis

Clearly the von Neumann condition is very important both practically and theoretically. Even for variable coefficient problems it can be applied locally (with local values of the coefficients) and because instability is a local phenomenon, due to the high frequency modes being the most unstable, it gives necessary stability conditions which can often be shown to be sufficient.

The following important points should be noted concerning the von Neumann method of examining stability:

The method which is based on Fourier analysis applies only if the coefficients of the linear difference equation are constant. If the difference equation has variable coefficients, the method can still be applied locally and it might be expected that a method will be stable if the von Neumann condition, derived as though the coefficients were constant, is satisfied at every point of the field.

Boundary conditions are neglected by the von Neumann method which applies in theory only to pure initial value problems with periodic initial data. It does however provide necessary conditions for stability of constant coefficient problems regardless of type of boundary conditions.

If we assume periodic boundary conditions the von Neumann analysis is based on the decomposition of the numerical solution into a Fourier series as

$$U_j^n = \sum_{p=-N}^{N} \kappa_p^n \mathrm{e}^{\mathrm{i}\xi_p(j\Delta x)}$$

where $\mathrm{i} = \sqrt{-1}$, κ_p^n is the amplification factor of the p-th harmonic and $\xi_p = \dfrac{p\pi}{N\Delta x}$. The product $\xi_p \Delta x$ is often called the phase angle:

$$\theta = \xi_p \Delta x$$

and covers the domain $(-\pi, \pi)$ in steps of π/N. The region around $\theta = 0$ corresponds to the low frequencies while the region $\theta = \pi$ is associated with the high-frequencies. In particular, the value $\theta = \pi$ corresponds to the highest frequency resolvable on the mesh, namely the frequency of wavelength $2\Delta x$.

Considering a single mode, $\kappa^n e^{ij\theta}$, its time evolution is determined by the same numerical scheme as the complete numerical solution U_j^n. Hence inserting a representation of this form into a numerical scheme we obtain a stability condition by majorising the amplification factor, κ.

The amplification factor is said to satisfy the *von Neumann condition* if there is a constant K such that

$$|\kappa(\xi)| \leq 1 + K\Delta t, \quad \forall \xi \in \mathbb{R}. \tag{2.13}$$

Consider a function U defined in a discrete set of points $x_j = j\Delta x$, $U_j = U(j\Delta x)$. The Euclidean or l_2-norm is defined to be

$$||U||_2 = \left(\sum_j \Delta x U_j^2 \right)^{1/2}.$$

We have the following theorem, the proof of which can be found for instance in Sod [73]:

Theorem 2.2 *A two level linear finite difference method is stable in the l_2-norm if and only if the von Neumann condition is satisfied.*

However, for some problems the presence of the arbitrary constant in (2.13) is too generous for practical purposes, although being adequate for eventual convergence in the limit $\Delta t \to 0$. In practice, the inequality (2.13) is substituted by the following stronger condition.

$$|\kappa(\xi)| \leq 1, \quad \forall \xi \in \mathbb{R}. \tag{2.14}$$

This has been called practical stability by Richtmyer and Morton [62] or strict stability by other authors. In some cases condition (2.13) allows numerical modes to grow exponentially in time for finite values of Δt. Therefore, the practical, or

strict, stability condition (2.14) is recommended in order to prevent numerical modes from growing faster than the physical modes of the differential equation.

The next result can be found in various works such as Warming and Hyett [91].

Theorem 2.3 *A necessary and sufficient condition for stability of the Lax-Wendroff scheme is*

$$\nu^2 + 2\mu \leq 1.$$

Proof: The proof of this result is well known; we provide a proof here merely to illustrate one general idea in stability analysis.

Suppose we substitute $\kappa^n e^{i\xi(j\Delta x)}$ into the numerical scheme. Then the amplification factor is given by:

$$\kappa(\xi) = 1 - \frac{\nu}{2}(e^{i\xi\Delta x} - e^{-i\xi\Delta x}) + (\frac{\nu^2}{2} + \mu)(e^{i\xi\Delta x} - 2 + e^{-i\xi\Delta x}).$$

Let $\theta = \xi\Delta x$

$$\kappa(\theta) = 1 - i\nu\sin\theta + (\frac{\nu^2}{2} + \mu)(-2 + 2\cos\theta)$$

and

$$|\kappa(\theta)|^2 = [1 - (\nu^2 + 2\mu)(-1 + \cos\theta)]^2 + \nu^2\sin^2\theta.$$

So

$$|\kappa(\theta)|^2 = [1 - 2(\nu^2 + 2\mu)\sin^2(\theta/2)]^2 + \nu^2\sin^2\theta. \tag{2.15}$$

Note that if we consider $\nu^2 = O(\Delta t)$ (presuming that the mesh is refined with the value μ fixed) we will have for $\nu^2 + 2\mu \leq 1$,

$$|\kappa(\theta)|^2 \leq 1 + O(\Delta t).$$

However, this is an inadequate stability criterion to apply to any computation carried out for a fixed value of Δx and Δt. If we continue the computation of (2.15) we obtain

$$\begin{aligned} |\kappa(\theta)|^2 &= 1 - 4(\nu^2 + 2\mu)\sin^2(\theta/2) + 4(\nu^2 + 2\mu)^2\sin^4(\theta/2) \\ &\quad +2\nu^2\sin^2(\theta/2) - 2\nu^2\sin^4(\theta/2). \end{aligned}$$

Let $C = \nu^2 + 2\mu$ and $s = \sin(\theta/2)$. We can write,

$$|\kappa(\theta)|^2 = 1 - [4C - 2\nu^2]s^2 + [4C^2 - 2\nu^2]s^4.$$

Since $s^4 \leq s^2$ then

$$|\kappa(\theta)|^2 \leq 1 - [4C - 2\nu^2]s^4 + [4C^2 - 2\nu^2]s^4$$

and

$$|\kappa(\theta)|^2 \leq 1 - [4C - 4C^2]s^4.$$

For $C \leq 1$ we will have $|\kappa(\theta)|^2 \leq 1, \ \forall\theta \in [-\pi, \pi]$.

The fact that the condition is necessary, is obtained straightforwardly by imposing

$$|\kappa(\pi)| \leq 1.$$

□

The Quickest scheme is more complex than the Lax-Wendroff scheme and consequently so is its von Neumann stability analysis. A necessary stability condition for the Quickest scheme was given by Leonard [41]. In the next lemma we combine this necessary condition with an additional one.

Lemma 2.4 *If the Quickest scheme is stable then*

$$\nu^2 + 6\mu\frac{1-2\nu}{3-2\nu} \leq 1, \tag{2.16}$$

$$\nu^2 + 6\mu(1-2\nu) \geq -2\nu. \tag{2.17}$$

Proof: The amplification factor for the Quickest scheme is given by:

$$\begin{aligned}\kappa(\theta) &= 1 - \mathrm{i}\nu\sin\theta - (\nu^2 + 2\mu)(1-\cos\theta) \\ &\quad -\frac{\nu}{3}(1-\nu^2-6\mu)(1-\mathrm{e}^{-\mathrm{i}\theta})(1-\cos\theta).\end{aligned}$$

The necessary conditions (2.16), (2.17) are obtained imposing

$$|\kappa(\pi)| \leq 1$$

since for $\theta = \pi$ we have the fundamental frequency that corresponds to the maximum wavelength. The necessary condition given by Leonard [41], the condition (2.16), was obtained from $\kappa(\pi) \geq -1$. We have

$$\kappa(\pi) = 1 - 2(\nu^2 + 2\mu) - \frac{4\nu}{3}(1-\nu^2-6\mu)$$

and if $|\kappa(\pi)| \leq 1$ then

$$\begin{aligned}\nu^2 + 2\mu + \frac{2\nu}{3}(1-\nu^2-6\mu) &\leq 1 \\ \nu^2 + 2\mu + \frac{2\nu}{3}(1-\nu^2-6\mu) &\geq 0.\end{aligned}$$

The conditions of the lemma follow from these inequalities. □

Although analytical von Neumann necessary and sufficient stability conditions have not been so far stated in the literature for the Quickest scheme, they have been computed numerically and plotted in papers by Leonard [41] and Morton and Sobey [49]. In the following theorem however we provide the analytical necessary and sufficient conditions for the Quickest scheme.

Theorem 2.5 *Let* $\alpha = 2\nu\mu - (\nu/3)(1-\nu^2)$, $\mathrm{n} = (2\mu+\nu^2)^2 - \nu^2 + 2\alpha(1-2\nu)$ *and* $\mathrm{d} = 4\alpha(2\mu+\nu^2-\nu-\alpha)$. *The Quickest scheme is stable if and only if*

a) The condition $-2\mu + \mathrm{n} - \mathrm{d} \leq 0$ *is satisfied;*

b) Let $S = \{(\mu,\nu) : 0 \leq \mathrm{n}/2\mathrm{d} \leq 1\}$. *For* $(\mu,\nu) \in S$, $\mathrm{n}^2/4\mathrm{d} \leq 2\mu$.

Proof: Considering the fact that

$$1 - \cos\theta = 2\sin^2(\theta/2) \quad \text{and} \quad \sin^2\theta = 4\sin^2(\theta/2)(1-\sin^2(\theta/2)),$$

the modulus of the amplification factor of the Quickest scheme is given by

$$\begin{aligned} |\kappa(\theta)|^2 &= 1 - 8\mu\sin^2(\theta/2) + 4[(2\mu+\nu^2)^2 - \nu^2 + 2\alpha(1-2\nu)]\sin^4(\theta/2) \\ &\quad -16\alpha(2\mu+\nu^2-\nu-\alpha)\sin^6(\theta/2). \end{aligned}$$

Let $s = \sin^2(\theta/2)$ then

$$\begin{aligned} |\kappa(\theta)|^2 &= 1 - 8\mu s + 4[(2\mu+\nu^2)^2 - \nu^2 + 2\alpha(1-2\nu)]s^2 \\ &\quad -16\alpha(2\mu+\nu^2-\nu-\alpha)s^3. \end{aligned}$$

It follows

$$|\kappa(\theta)|^2 = 1 + 4s\phi(s),$$

where

$$\begin{aligned} \phi(s) &= -2\mu + [(2\mu+\nu^2)^2 - \nu^2 + 2\alpha(1-2\nu)]s \\ &\quad -4\alpha(2\mu+\nu^2-\nu-\alpha)s^2, \ s \in [0,1]. \end{aligned}$$

The stability condition $|\kappa(\theta)| \leq 1, \forall\theta \in \mathbb{R}$, is satisfied if and only if the condition

$$\phi(s) \leq 0, \quad s \in [0,1]$$

is satisfied. To prove this condition it is necessary and sufficient to prove that $\phi(0) \leq 0$, $\phi(1) \leq 0$ and that for $s_* \in [0,1]$ such that $\phi'(s_*) = 0$ then $\phi(s_*) \leq 0$.

We have that $\phi(0) = -2\mu$ and it is negative for all μ. The inequality $\phi(1) \leq 0$ is true if and only if the condition a) of the theorem is satisfied. The zero of the function $\phi'(s)$ is

$$s_* = \frac{(2\mu + \nu^2)^2 - \nu^2 + 2\alpha(1 - 2\nu)}{8\alpha(2\mu + \nu^2 - \nu - \alpha)}. \tag{2.18}$$

We want to find μ and ν such that $0 \leq s_* \leq 1$ and $\phi(s_*) \leq 0$. We have

$$\phi(s_*) = -2\mu + \frac{1}{4}\frac{[(2\mu + \nu^2)^2 - \nu^2 + 2\alpha(1 - 2\nu)]^2}{4\alpha(2\mu + \nu^2 - \nu - \alpha)}. \tag{2.19}$$

The condition $\phi(s_*) \leq 0$ is verified if and only if

$$\frac{[(2\mu + \nu^2)^2 - \nu^2 + 2\alpha(1 - 2\nu)]^2}{16\alpha(2\mu + \nu^2 - \nu - \alpha)} \leq 2\mu.$$

and this proves the theorem. □

We illustrate the stability conditions for the Lax-Wendroff scheme and Quickest scheme in figure 2.1.

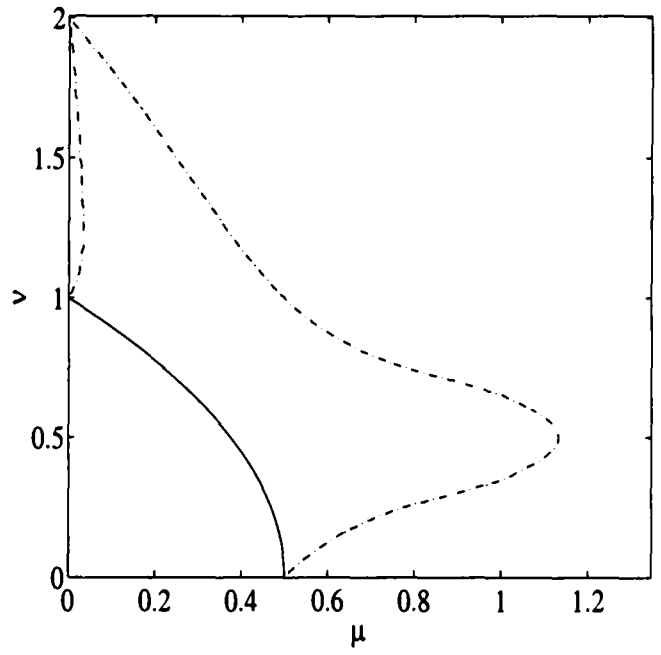


Figure 2.1: von Neumann stability regions for the Quickest scheme $(- \cdot -)$ and the Lax-Wendroff scheme $(-)$.

Although the analytical necessary and sufficient conditions presented above, for the Quickest scheme have a cumbersome form, we can check for a fixed value of ν or μ for which values the method is stable.

Assume that $\nu = 1/2$ and we want to find for which values of μ the method is practically stable. We have

$$\alpha = \mu - 1/8, \quad \mathrm{n} = 4(\mu - 1/8)(\mu - 3/8), \quad \mathrm{d} = 4(\mu - 1/8)^2$$

The first condition of the theorem is satisfied for all μ, but the second condition gives us that the method is practically stable for $0 \leq \mu \leq 9/8$.

Consider $\mu = 0$. We have

$$\alpha = -(\nu/3)(1-\nu^2), \quad \mathrm{n} = (1-\nu^2)(\nu/3)(\nu-2), \quad \mathrm{d} = (4/3^2)\nu^2(1-\nu^2)(1-\nu)(2-\nu)$$

The conditions of the theorem give us $0 \leq \nu \leq 1$.

For $\nu = 0, 1$ we have

$$\alpha = 0, 2\mu, \quad \mathrm{n} = 4\mu^2, \quad \mathrm{d} = 0.$$

The method is stable for $\mu \leq 1/2$.

All the values obtained by these examples are in agreement with figure 2.1, which was calculated numerically. Because of the consistent third-order estimation of the convective term, the stability range of the amplitude factor for Quickest is considerably improved over other explicit methods. Of particular interest is the significant region above $\nu = 1$ for finite μ.

In subsequent chapters we study in detail the effect of boundary conditions on the stability of a numerical scheme and observe that the von Neumann stability analysis is still an important tool for the determination of the stability region of the scheme.

2.5 Truncation error

For the Lax-Wendroff and Quickest schemes in the previous section, the error $E^n = u^n - U^n$ for the set of nodal errors, is given by

$$E^{n+1} = AE^n + \Delta t T^n$$

where the matrix A contains the coefficients of the difference formulas and T^n is the truncation error. For any chosen norm for the error, the practical stability requirement is that $||A|| \leq 1$. Then a global error bound is given by

$$||E^n|| \leq ||E^0|| + \Delta t \sum_{j=0}^{n-1} ||T^j|| \leq ||E^0|| + (n\Delta t) \max_{0 \leq j \leq n-1} ||T^j||.$$

Practical stability conditions, in the l_2 norm, were discussed in the previous section since in this case the von Neumann condition is equivalent to the condition $||A|| \leq 1$.

The following local truncation error of the Lax-Wendroff scheme and Quickest scheme can be derived using the modified equation method as in Warming and Hyett [91] or the Peano kernel theorem as in Morton and Sobey [49].

Lax-Wendroff:

$$\begin{aligned} \Delta t T_j^n &= \frac{1}{6}\Delta x^3 \nu(1-\nu^2-6\mu)U_{x^3}^n(x_j) \\ &\quad + \frac{1}{24}\Delta x^4(12\mu^2 - 2\mu + 3\nu^2(1-\nu^2-4\mu))U_{x^4}^n(x_j) + \ldots \end{aligned} \tag{2.20}$$

Quickest:

$$\begin{aligned} \Delta t T_j^n &= \frac{1}{24}\Delta x^4(12\mu^2 - 2\mu - 12\mu\nu(1-\nu) + \nu(1-\nu^2)(2-\nu))U_{x^4}^n(x_j) \\ &\quad + \ldots \end{aligned} \tag{2.21}$$

On theoretical grounds, over a finite interval of time and assuming $\Delta t = O(\Delta x)$, we expect the Lax-Wendroff method to be close to $O(\Delta x^2)$ accuracy while the Quickest method should be $O(\Delta x^3)$ accurate. These estimates are not rigorous since μ and ν need not be constant and may vary depending on how Δx and Δt are related when we refine the mesh.

The one dimensional Quickest scheme resembles a modified Lax-Wendroff scheme where the additional term is introduced to give an overall truncation error of $O(\Delta t^2, \Delta x^2)$. The convective terms are discretised to $O(\Delta x^3)$ accuracy and the time derivative is discretised to $O(\Delta t^3)$ accuracy in the limit $D \to 0$.

As we have already mentioned, Quick and Quickest where derived by Leonard using control volume arguments. Interestingly enough the Quick method — the method equivalent to Quickest but developed for quasi-steady flows — caused some confusion in literature over the actual order of accuracy of the convective term. This confusion was generated by a different approach to how the actual truncation error of a finite-volume formulation should be computed. The different points of view can be found in Johnson and Mackinnon [34] and Leonard [46]. In summary, Leonard [46] argues that a truncation error of a discretisation obtained by finite-volume formulation is obtained in a different way from the one we use for finite-difference discretisations. On the other hand Johnson and Mackinnon [34] contend that one should not distinguish whether a discrete scheme is derived using the finite-difference or finite-volume method in applying

Taylor series analysis to determine its order of accuracy. In fact these two approaches give two different orders of accuracy. This debate does not affect our discussion of the Quickest scheme since the difference shows up only in steady-state calculations.

In the following chapter we consider the one-dimensional convection-diffusion problem but with a left boundary condition. The main focus will be on the Quickest scheme and the choice of an adequate numerical boundary condition for this scheme.

Chapter 3

On the influence of boundary conditions

Our understanding of the behaviour of numerical solutions for evolutionary convection-diffusion equations is mainly based on the analysis of infinite domain situations, with stability given by von Neumann analysis. Almost all practical problems involve physical domains with boundaries. For evolution problems with Dirichlet boundary conditions, some algorithms can be used without alteration near a boundary. However, the application of higher order methods such as Quickest or second order upwinding (including schemes with flux limiters) introduces difficulty near an inflow boundary, since for interior points adjacent to the boundary there are insufficient upstream points for the high order scheme to be applied without alteration. For that reason such methods require a careful treatment on the inflow boundary, where additional numerical boundary conditions have to be introduced. The choice of numerical boundary conditions turns out to be crucial for stability.

Different approaches have been used in the literature to investigate the stability of a two level finite difference scheme:

a) One is to perform a von Neumann analysis, which is only applicable to pure initial value problems and to problems with periodic boundary conditions, disregarding the boundary data;

b) To apply the difference scheme for some values and look at the solutions to see if those values give a stable approximation. This technique can only be used to demonstrate instability.

c) To write the difference equation in one time-step form $U^{n+1} = AU^n$ and then study the eigenvalues of the matrix A which contains the coefficients of

the difference formulas and take the region of stability as the region where the spectral radius of A is less than one;

d) To use the matrix analysis that consists in majorising the norm of the iterative matrix A by a reasonable constant;

e) To use the energy method that usually leads to sufficient conditions;

f) To use normal mode theory.

In this chapter we consider a model problem with positive advection velocity and a left physical boundary condition and discuss the different choices of numerical boundary conditions and its effects on the stability and accuracy of the resulting numerical schemes. A test problem is described, showing the practical advantages of some numerical boundary conditions versus the others by comparison with an exact solution. The results we present are from Sousa and Sobey [76].

To analyse the stability of the schemes we examine the spectral radius and the matrix norm of the iteration matrix A, taking into account the stability condition given by the von Neumann method. Furthermore, this framework is used in subsequent chapters to investigate the stability of finite difference schemes for model problems in two dimensions with constant flow velocity and non-uniform flow velocity. In the next chapter, however, we will show how it is possible to apply a more powerful method, the Godunov-Ryabenkii theory, to investigate the stability of the same schemes considered in this chapter for the one-dimensional problem.

3.1 The model problem

In this chapter we focus on a one dimensional problem on a half real line. We consider the convection-diffusion equation (2.1) with the initial condition

$$u(x,0) = f(x), \quad x \geq 0, \tag{3.1}$$

given and subject to the boundary conditions

$$u(x,t) \to 0, \ x \to \infty \qquad \text{and} \qquad u(0,t) = g(t), \ t \geq 0. \tag{3.2}$$

The exact solution of the system (2.1), (3.1) and (3.2) can be found using Laplace transforms in t:

$$u(x,t) \ = \ \frac{1}{\sqrt{\pi}} \int_0^t g(t-\tau) G^*(x,\tau) d\tau$$

$$+\frac{1}{\sqrt{\pi}}\int_{\frac{Vt-x}{2\sqrt{Dt}}}^{+\infty} f(x-Vt+2\sqrt{Dt}\xi)\mathrm{e}^{-\xi^2}d\xi$$
$$-\frac{1}{\sqrt{\pi}}\int_{\frac{Vt+x}{2\sqrt{Dt}}}^{+\infty} f(-x-Vt+2\sqrt{Dt}\xi)\mathrm{e}^{Vx/D}\mathrm{e}^{-\xi^2}d\xi \qquad (3.3)$$

where the function $G^*(x,\tau)$ is given by

$$G^*(x,\tau) = \frac{x}{2\sqrt{D}\tau^{2/3}}\mathrm{e}^{-(x-V\tau)^2/4D\tau}.$$

Applying the result to evolution over one time step, we write,

$$u(x,t_n+\Delta t) = \frac{1}{\sqrt{\pi}}\int_{t_n}^{t_n+\Delta t} g(t_n+\Delta t-\tau)G^*(x,\tau)d\tau$$

$$+\frac{1}{\sqrt{\pi}}\int_{\frac{V\Delta t-x}{2\sqrt{D\Delta t}}}^{+\infty} u(x-V\Delta t+2\sqrt{D\Delta t}\xi,t_n)\mathrm{e}^{-\xi^2}d\xi \qquad (3.4)$$

$$-\frac{1}{\sqrt{\pi}}\int_{\frac{V\Delta t+x}{2\sqrt{D\Delta t}}}^{+\infty} u(-x-V\Delta t+2\sqrt{D\Delta t}\xi,t_n)\mathrm{e}^{Vx/D}\mathrm{e}^{-\xi^2}d\xi.$$

The exact solution for this model problem differs from the solution (2.5) obtained for a convection-diffusion problem on the whole real line. This is also the fundamental solution we shall use, in a following chapter, to deduce new numerical schemes.

3.2 The numerical boundary condition

The model problem we consider here is a simplified form of (3.2) where, for the solution defined on the half-line, the inflow boundary condition is given by

$$u(0,t) = 0. \qquad (3.5)$$

As the Lax-Wendroff scheme is a three point scheme it can be used at all interior points. On the other hand the Quickest scheme uses two points upstream and can not be applied on the first interior point of the mesh. At that point we need to apply a numerical boundary condition. In the next sections we discuss a number of different numerical boundary conditions which can be used at the first interior point of the scheme. The new results in this work concern the consequences for stability and accuracy of the resulting schemes.

3.2.1 A numerical boundary condition suggested by Leonard

Leonard [41] suggested the following boundary condition based on control-volume arguments for a cell $[\Delta x/2, 3\Delta x/2]$: a hypothetical node is specified at $\Delta x/2$ downstream of the physical boundary at $x = 0$. This node is denoted by B. It is assumed that the Dirichlet condition can be applied at this node, rather than at $x = 0$, so that in this case: $U_B = 0$. A linear interpolation, between the next boundary value and the first interior point, $U_B^{n+1/2} = (U_0^{n+1} + U_1^n)/2$ gives for this case

$$U_0^{n+1} = -U_1^n. \tag{3.6}$$

Then a control volume argument is applied to the first interior region $\Delta x/2 < x < 3\Delta x/2$ to obtain

$$U_1^{n+1} = U_1^n - \nu(U_r^n - U_l^n) + \mu(U_0^n + U_2^n - 2U_1^n), \tag{3.7}$$

where fictitious values U_r^n,U_l^n are evaluated at $3\Delta x/2$ and $\Delta x/2$ respectively. Applying the numerical boundary condition (3.5) at $\Delta x/2$ and an interpolation at $3\Delta x/2$ gives:

$$\begin{aligned} U_l^n &= U_B^n, \\ U_r^n &= \tfrac{1}{2}(U_1^n + U_2^n) - \tfrac{1}{8}(U_0^n + U_2^n - 2U_1^n). \end{aligned}$$

Since in this case $U_B^n = 0$, (3.7) can be rewritten

$$U_1^{n+1} = U_1^n - \frac{\nu}{8}(6U_1^n + 3U_2^n - U_0^n) + \mu(U_0^n + U_2^n - 2U_1^n). \tag{3.8}$$

This provides an algorithm for dealing with the first interior point which incorporates the numerical boundary condition.

3.2.2 Downwind third difference

The derivation of the numerical scheme Quickest using (2.6) was based on a local cubic approximation. If at the first internal point of the scheme we choose the points used for interpolation as U_0^n,U_1^n, U_2^n and U_3^n we bring in a forward third difference instead of a backward third order difference, giving a scheme

$$U_1^{n+1} = [1 - \nu\Delta_0 + (\frac{1}{2}\nu^2 + \mu)\delta^2 + \nu(\frac{1}{6} - \frac{\nu^2}{6} - \mu)\delta^2\Delta_+]U_1^n, \tag{3.9}$$

where Δ_+ is the forward operator defined by $\Delta_+ U_j := U_{j+1} - U_j$. The use of this downwind third difference does not affect accuracy since it is still based on a local cubic approximation. However as we shall show, it does have penalties in terms of stability.

3.2.3 Lax-Wendroff

An alternative numerical boundary condition at the first interior point is obtained by applying a quadratic local approximation using the points U_0^n,U_1^n and U_2^n for interpolation. In that way we have the Lax-Wendroff method only at that point:

$$U_1^{n+1} = [1 - \nu\Delta_0 + (\frac{1}{2}\nu^2 + \mu)\delta^2]U_1^n. \tag{3.10}$$

3.2.4 A fictitious point U_{-1}

Let us suppose we apply a Quickest scheme to the first internal point without modification by assuming a fictitious point U_{-1}. In this section we describe one way to calculate this fictitious point using the boundary data.

We know that on the whole real line the exact solution of the convection-diffusion equation (2.1) subject to an initial condition is given by

$$u(x,t) = \int_{-\infty}^{+\infty} u(\eta,0)G(x-\eta;t)d\eta \tag{3.11}$$

with $G(z;\tau) = \dfrac{1}{2\sqrt{D\pi\tau}}\mathrm{e}^{-(z-V\tau)^2/4D\tau}$.

In our case we only have initial data for $x \geq 0$ but the boundary data at $x = 0$ for $t > 0$ will correspond to (unknown) initial data for $x < 0$. In particular at $x = 0$, $g(t) = u(0,t)$ is given by

$$g(t) = \int_{-\infty}^{+\infty} u(\eta,0)G(-\eta;t)d\eta. \tag{3.12}$$

If we define

$$\begin{aligned} u_+(\eta) &= u(\eta,0) \qquad \eta \geq 0 \\ u_-(\eta) &= u(\eta,0), \qquad \eta < 0, \end{aligned}$$

then we can write

$$\int_{-\infty}^{0} u_-(\eta)G(-\eta;t)d\eta = g(t) - \int_{0}^{+\infty} u_+(\eta)G(-\eta;t)d\eta.$$

Given u_+ and g, this defines an inverse problem for u_-. This gives one way to deal with a fictitious point on the left of $x = 0$. Rather than try to determine $u_-(\eta)$ analytically, we consider an application of (3.12) over one time step:

$$g(t_{n+1}) = \int_{-\infty}^{+\infty} u(\eta,t_n)G(-\eta;\Delta t)d\eta. \tag{3.13}$$

Then approximating the solution $u(\eta, t_n)$ by a quadratic polynomial around $x = 0$ using U^n_{-1}, U^n_0 and U^n_1 gives a Lax-Wendroff approximation on the boundary

$$g^{n+1} = g^n - \frac{\nu}{2}(U^n_1 - U^n_{-1}) + (\mu + \frac{\nu^2}{2})(U^n_1 - 2g^n + U^n_{-1}), \tag{3.14}$$

where $g^n := g(t_n)$. This is of course the same as (2.7) with $j = 0$ and $U^n_0 = g^n$.

Since g^n, g^{n+1} and U^n_1 are known, from (3.14) we can calculate the fictitious value U^n_{-1} . After we have obtained the value, we can apply the correct third difference at the first interior point. As we shall show below this numerical boundary condition has a substantial advantage in terms of stability.

3.3 Stability analysis

Von Neumann stability analysis is usually only possible for fairly idealised situations such as linear constant coefficient equations on infinite domains. Thus in the case of a bounded domain it is no longer possible to use simple von Neumann analysis. However, for a scheme subject to boundary conditions to be stable, first of all we need to assure that the Cauchy problem is stable, that is, that the scheme is von Neumann stable in the infinite domain.

When we refer in the next sections to a scheme as von Neumann stable, we mean that the practical von Neumann condition (2.14) is satisfied, as we did in the second chapter.

All the explicit methods we discuss can be written in the form of a matrix iteration. Assume that the nodal points are $U^n_j, j = 0, \ldots, N$ and that the outflow boundary is such that

$$U^n_N = 0, \quad \forall n. \tag{3.15}$$

The choice of this outflow boundary is motivated by the fact that we assume that the exact solution goes to zero when x goes to infinity.

Introducing the vector $U^n = \{U^n_0, U^n_1, \ldots, U^n_{N-1}\}^T$, all the schemes may be written as matrix equations

$$U^{n+1} = AU^n, \quad n = 0, 1, 2, \ldots. \tag{3.16}$$

where A is an $N \times N$ matrix and depends on the scheme used.

Any errors E^n in a calculation based on (3.16) will grow according to

$$E^{n+1} = AE^n, \quad n = 0, 1, 2, \ldots. \tag{3.17}$$

where $E^n = u^n - U^n$ with u^n, U^n the exact and numerical solutions of (3.16), respectively, at $t = n\Delta t$.

Given $A \in \mathbb{R}^{N\times N}$ denote the spectral radius of A by $\rho(A)$ and the L_2-norm of the matrix A by $||A||$. We recall that

$$||A|| = \rho(A) \quad \text{if} \quad A \in \mathbb{R}^{N\times N} \quad \text{is normal.}$$

It is well known that for any $A \in \mathbb{R}^{N\times N}$

$$A^m \to 0 \quad \text{as} \quad m \to \infty \quad \text{if and only if} \quad \rho(A) < 1,$$

and that

$$\rho(A) \leq ||A||.$$

A simple criterion for regulating the error growth governed by (3.17) is given by

$$\rho(A) \leq 1. \tag{3.18}$$

When the matrix A is not normal the spectral radius gives no indication of the magnitude of E^n for finite n. In this case a condition of the form $\rho(A) < 1$ guarantees eventual decay of the solution, but does not control the intermediate growth of the solution.

A more severe condition for regulating error growth follows from (3.17). If the matrix norm, $||A||$, is consistent with the vector norm, $||E||$, then

$$||E^{n+1}|| \leq ||A||||E^n||, \quad n = 0, 1, 2, \ldots,$$

and the condition

$$||A|| \leq 1, \tag{3.19}$$

is sufficient to ensure that the error cannot grow with n.

From (3.17) we have

$$E^n = A^n E^0, \quad n = 1, 2, \ldots. \tag{3.20}$$

The expression (3.20) shows that in order for all E^n to remain bounded and the scheme (3.16) to remain stable the infinite set of operators A^n has to be uniformly bounded for all n, Δt and Δx. Consequently by replacing the severe condition (3.19) by

$$||A^m|| \leq K, \quad \forall m \tag{3.21}$$

with a suitable choice of m and K, gives a more relaxed condition which allows a limited growth of the error vector after m time steps. The error is controlled

by a reasonable constant for all $m \geq 0$, although in practice the concept of a reasonable constant is not straightforward. Recently several authors, including van Dorsselaer [89], Lenferink and Spijker [40] and Reddy and Trefethen [59, 60, 61] have carried out work related to non-normality effects and have found some sufficient conditions to bound $||A^m||$ for all $m \geq 0$.

By examining both the spectral radius and the matrix norm, we are able to find very accurate regions of stability for our methods.

The Godunov-Ryabenkii theory can also be applied to these problems as we will see in the next chapter. Godunov and Ryabenkii [21] derived necessary conditions occasioned by the boundary conditions. This work was further developed by Kreiss [35] and Gustafsson *et al* [28]. This method is quite powerful, but often leads to very complex and intractable calculations.

3.4 Practical stability regions

To ensure stability of a scheme subject to numerical boundary conditions a necessary condition is that the scheme is von Neumann stable in the infinite domain. The Lax-Wendroff and Quickest schemes are von Neumann stable, provided μ, ν are such as to lie within the region bounded by the respective curves in figure 2.1. The regions plotted in figure 2.1 are necessary and sufficient for von Neumann stability of these schemes. This means that when the Quickest scheme is subject to numerical boundary conditions, any stability region should lie inside the stability region displayed in figure 2.1.

Our plan is to show curves which define $\rho(A) = 1$, curves which define $||A|| = 1$, and curves which define $||A^n||$ for some fixed n. These curves have been computed using MATLAB for finite size matrices. The shaded area between two curves is where eigenvalue analysis would indicate stability but where matrix analysis tells us the error might grow by many orders of magnitude before eventually decaying. A simple guide for practical stability is to stay within the region where $||A|| \leq 1$, although that can be very restrictive in some cases. A less restrictive condition is to consider the region where $||A^n|| \leq 1$ for some $n \geq 1$ not very large. Typically the size of the matrix A considered is $N = 30$ unless another size is mentioned. The outflow boundary condition considered is always the Dirichlet boundary condition (3.15).

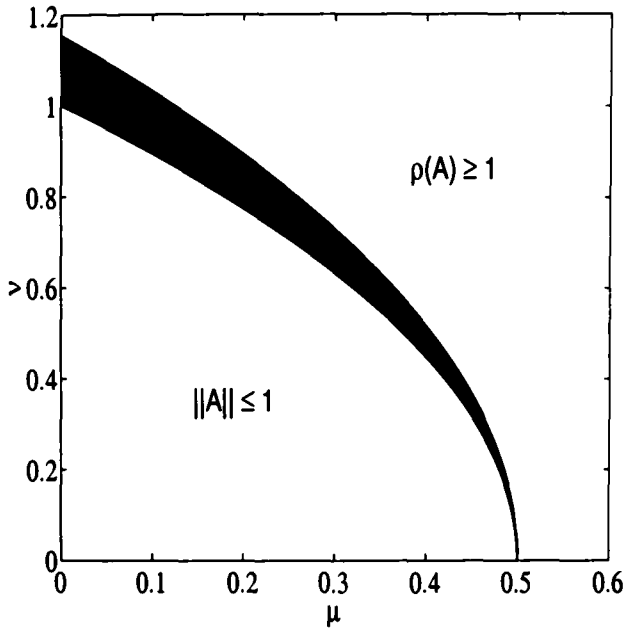


Figure 3.1: Stability region for the Lax-Wendroff scheme

3.4.1 Lax-Wendroff

For the Lax-Wendroff scheme we consider homogeneous Dirichlet boundary conditions at the inflow and outflow. By periodisation of the real line this problem may be studied by means of Fourier analysis. Consequently the stability region is given by the von Neumann condition. In figure 3.1 we see that the region $||A|| \leq 1$ coincides with the well known von Neumann condition: $\nu^2 + 2\mu \leq 1$. The eigenvalues of the matrix A are given by $\lambda_s = \nu^2 + 2\mu + \mathrm{i}\nu\cos(s\pi/(N+1))$, $s = 1, \ldots, N-1$. We can also observe that for finite matrices the spectral radius is greater than that indicated by von Neumann analysis.

3.4.2 Quickest

As we have indicated, one difficulty in applying the Quickest scheme is choosing the appropriate numerical boundary condition. Since the iteration matrix A is slightly different for different boundary conditions, stability results also differ. We apply inlet (3.5), and outlet (3.15), Dirichlet boundary conditions.

Numerical boundary condition suggested by Leonard

On applying the boundary method suggested by Leonard (section 3.2.1), the

region where the eigenvalues are less than one (figure 3.2a) contains almost all the von Neumann stability region (figure 2.1), except for a small portion on the left corner of figure 3.2a, although the norm of the matrix A is never less than one. The fact that the norm is never less than one does not imply that the method is not stable, since $||A|| \leq 1$ is only a sufficient condition for stability but not a necessary condition. It is interesting to see what happens when the norm of powers of A, $||A^n||$ is computed. We plot in figure 3.2b the regions $||A^n|| \leq 1$ for $n = 3, 6, 12, 48$. The region defined by $||A^{48}|| = 1$ is approximately the same as the von Neumann region. Of course the condition $\rho(A) \leq 1$ implies that $||A^n||$ tends to zero when $n \to \infty$, but the main point for a practical stability is that $||A^n||$ does not grow very strongly and starts to decay after few steps in time. The practical stability region for this case is approximately the von Neumann region given by the intersection of the condition $\rho(A) \leq 1$ (figure 3.2a) with the von Neumann condition (figure 2.1).

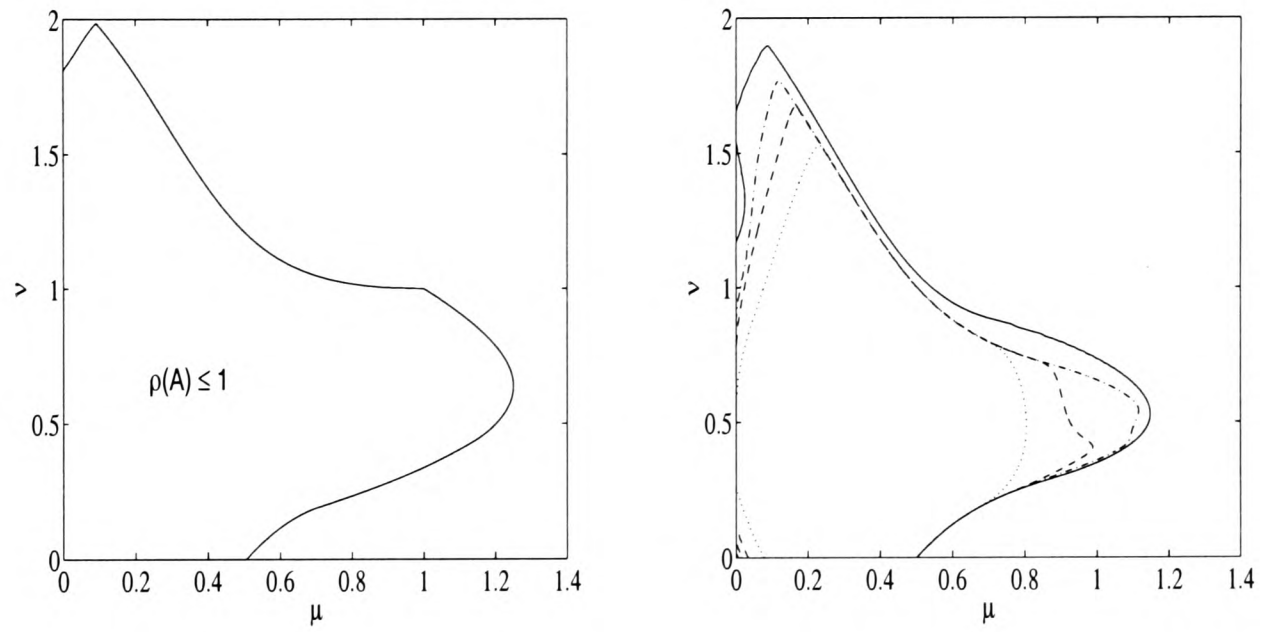


Figure 3.2: Stability region for the Quickest scheme combined with the numerical boundary condition suggested by Leonard (section 3.2.1): (a) Region where the spectrum is less than one; (b) $||A^3|| = 1$ $(\cdots)$, $||A^6|| = 1$ (- -), $||A^{12}|| = 1$ $(-\cdot)$, $||A^{48}|| = 1$ (–)

Downwind third difference numerical boundary condition

When the Quickest scheme is used with a downwind third order difference

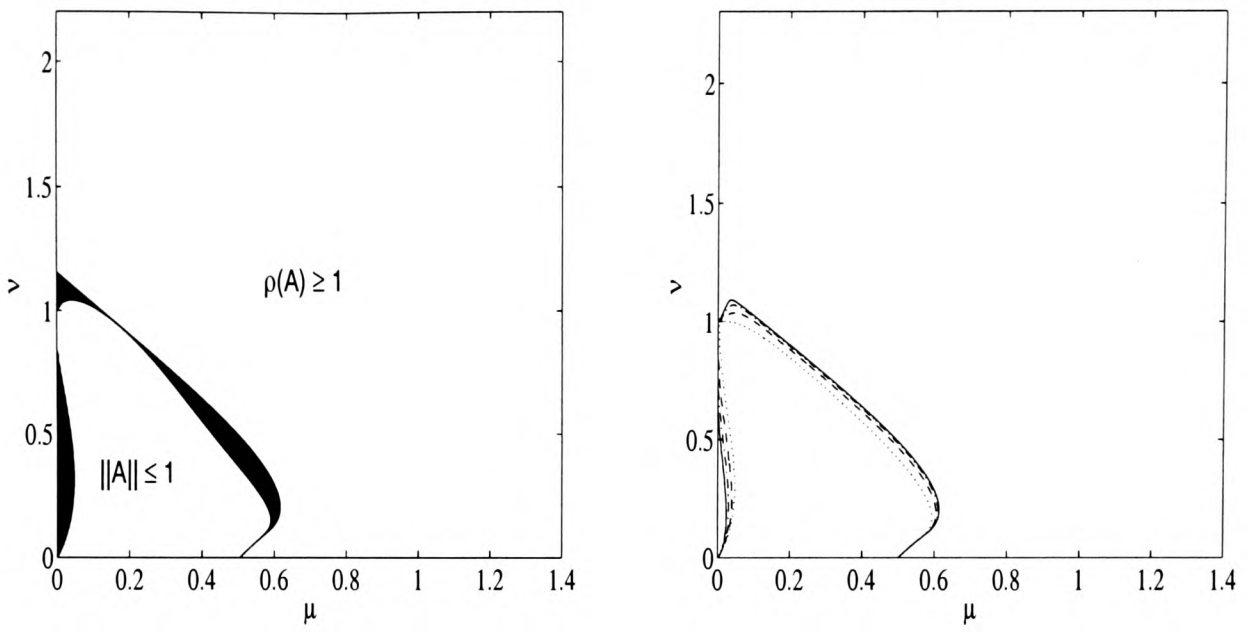


Figure 3.3: Stability region for the Quickest scheme with a downwind numerical boundary condition (section 3.2.2): (a) Norm and spectral radius of the iterative matrix A; (b) $||A^3|| = 1$ $(\cdots)$, $||A^6|| = 1$ (- -), $||A^{12}|| = 1$ $(-\cdot)$, $||A^{24}|| = 1$ (–)

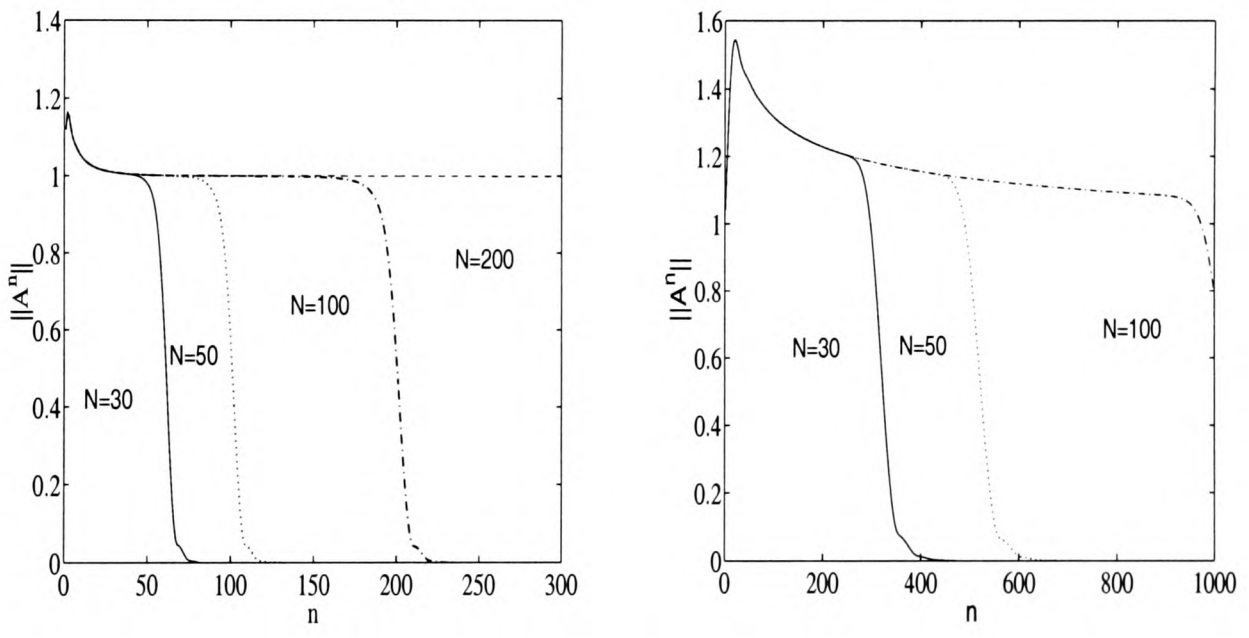


Figure 3.4: Evolution of matrix norm for the Quickest scheme with a downwind numerical boundary condition (section 3.2.2): (a) Behaviour of the function $||A^n||$ at $\mu = 0.001$, $\nu = 0.5$ for different matrices sizes (N) (b) Behaviour of the function $||A^n||$ at $\mu = 0.001$, $\nu = 0.1$ for different matrices sizes (N)

applied to the first mesh point, a substantial part of the stability region is lost (figure 3.3a). When plotting the regions $||A^n|| \leq 1, n = 3, 6, 12, 24$ (figure 3.3b) we can observe that as we increase n the region $||A^n|| \leq 1$ approximates the region defined by $\rho(A) \leq 1$. There is a small portion for μ small (figure 3.3b) where $||A^n||$ does not become less than one for a relatively small n, although it does not grow significantly either, as shown in figure 3.4.

If we consider the area where $||A|| > 1$ and $\rho(A) \leq 1$ for small μ (the shaded region on figure 3.3a) then for $\mu = 0.001$ and $\nu = 0.5$, if N (the size of the matrix A) is increased, the maximum value of $||A^n||$ does not increase, i.e., $||A^n|| \leq 1.2$ for all n and N considered (figure 3.4a). In figure 3.4b for $\mu = 0.001$ and $\nu = 0.1$ we can observe that for the matrix size $N = 30$ the norm starts to be less than one around $n = 300$. We also have that $||A^n|| \leq 1.6$ for all n and N considered. In this region more steps in time are needed before $||A^n||$ becomes less than one.

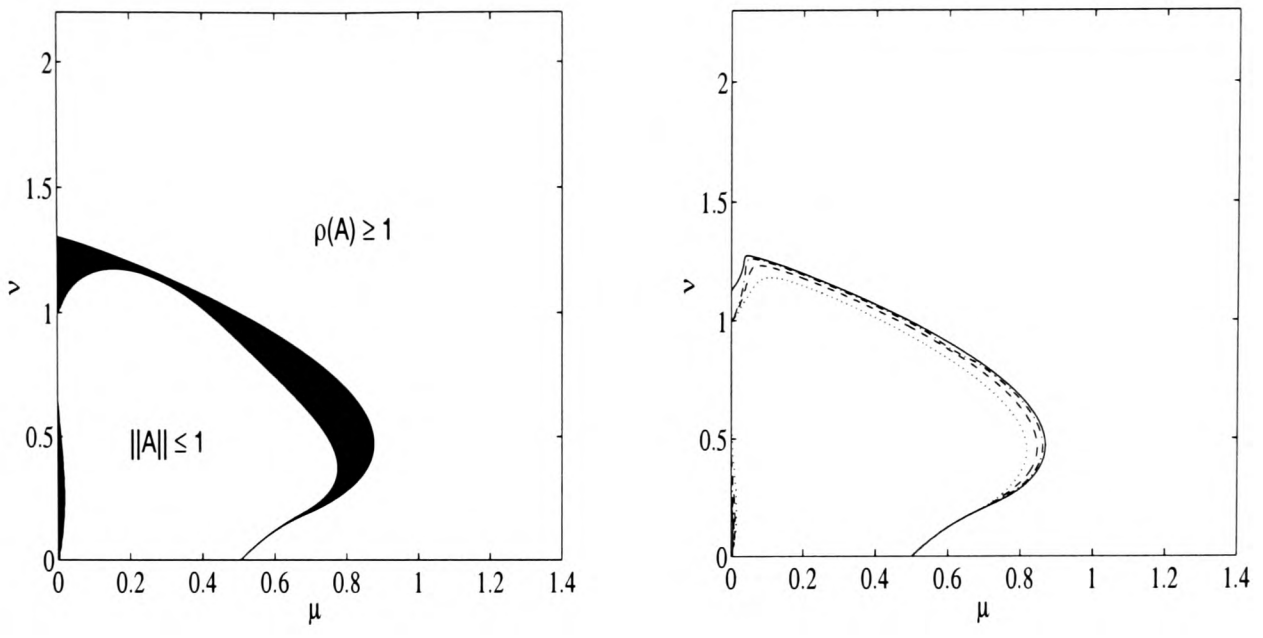
In cases where μ is small , figure 3.3 indicates a region of potential instability but it is in fact a stable region where the condition (3.21) is satisfied with values of K not much larger than one, see figure 3.4. Consequently the practical stability region is given by the condition $\rho(A) \leq 1$ that lies inside the von Neumann region. In the next chapter we shall prove analytically that the stability region for the Quickest scheme with the downwind third difference numerical boundary condition is given by this region by applying the Godunov-Ryabenkii theory.

Lax-Wendroff numerical boundary condition

The effect on stability of applying the Lax-Wendroff scheme to the first interior point of the scheme is shown in figure 3.5. The region of stability is larger than with a downwinded third difference. The shaded area in figure 3.5 that lies inside the von Neumann stability region (figure 2.1) is still a region where we have practical stability although the norm exceeds one, as we can conclude from the behaviour of $||A^n||$ as n increases in figure 3.5b, where the regions $||A^n|| \leq 1, \quad n = 6, 12, 24, 48$ are plotted.

A fictitious point numerical boundary condition

The numerical boundary condition which used a fictitious point value to apply an upwinded third difference, associated with inlet and outlet Dirichlet boundary conditions gives essentially the same stability region as the von Neumann condition. The region where $||A|| \leq 1$ (see figure 3.6) is coincident with the region where the interior scheme Quickest is von Neumann stable. We can



(a) (b)

Figure 3.5: Stability region for the Quickest scheme with Lax Wendroff as the numerical boundary condition (section 3.2.3): (a) Norm and spectral radius of the iterative matrix A; (b) $||A^6|| = 1$ $(\cdots)$, $||A^{12}|| = 1$ (- -), $||A^{24}|| = 1$ $(-\cdot)$, $||A^{48}|| = 1$ (–)

conclude that we have practical stability for that scheme in the region given by the condition $||A|| \leq 1$.

An important question is whether the results presented are sensitive to changes in the size of the iteration matrix. Our numerical experience is that changing the matrix size does not change the general conclusions. This is illustrated by showing the effect of changing the size of the matrices on the eigenvalues and norm in figure 3.7. Although these results are for a Quickest scheme with Lax-Wendroff as a numerical boundary condition, this is also a general profile of the results for the other methods. Figure 3.7 shows what happens to the spectrum and the norm for two cases of interest, one when the norm and eigenvalues are simultaneously less than one (figure 3.7a) and the other when the spectrum is still less than one but the norm is not (figure 3.7b). We observe slight changes in the spectral radius with the dimension of the matrix but it does not become larger than one. The matrix norm seems more stable to changes of the matrix size.

Although when we are dealing with non-normal matrices the eigenvalues are not reliable indicators of stability, in our examples the region where the eigenvalues are less than one intersected the von Neumann stability region for

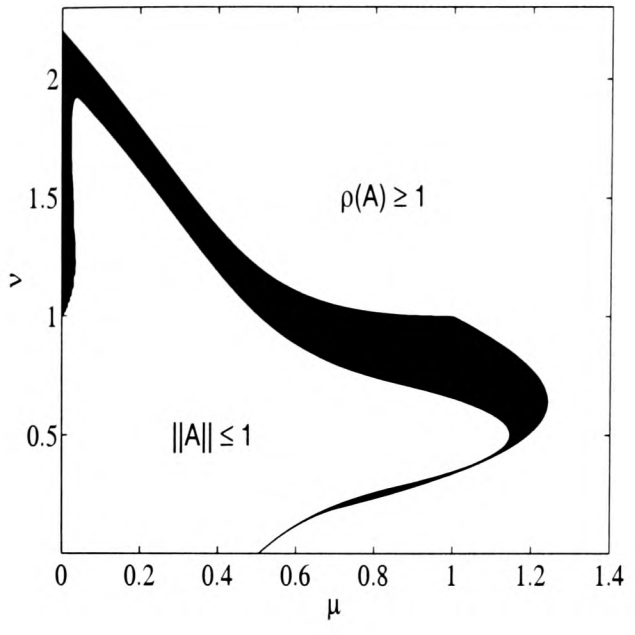


Figure 3.6: Stability region for the Quickest scheme using a fictitious point (section 3.2.4)

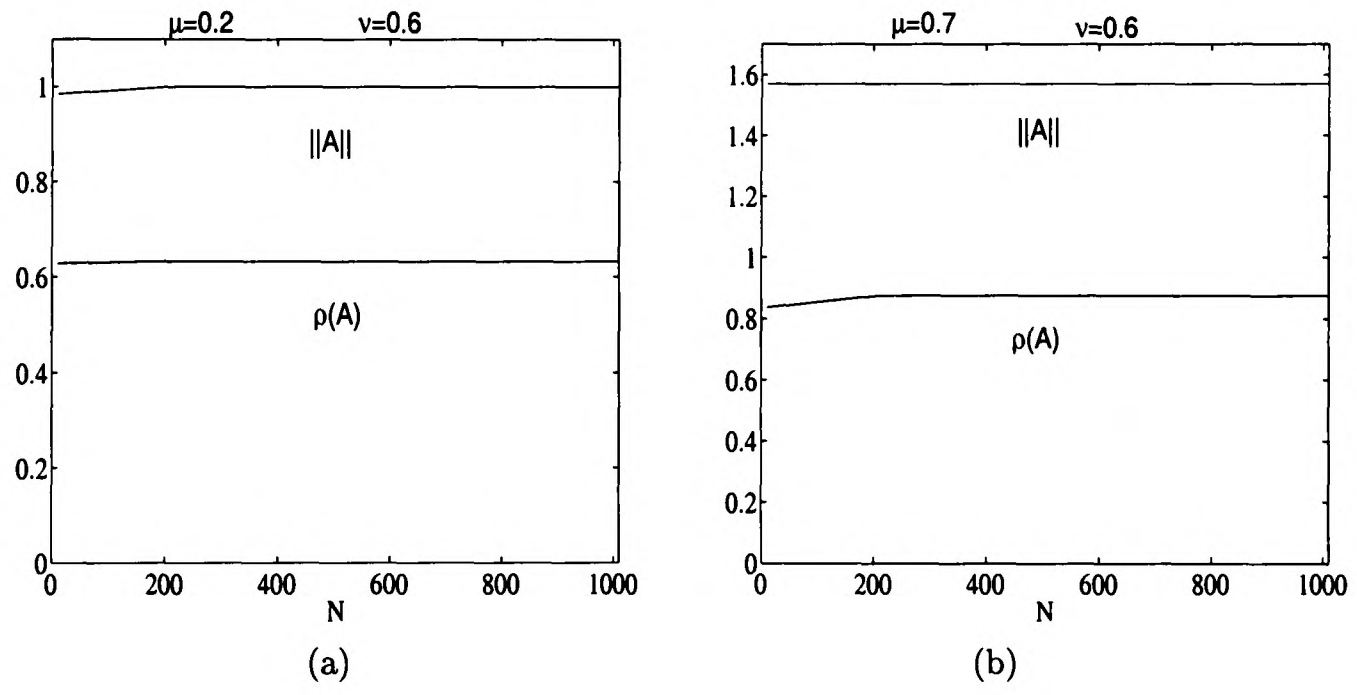


Figure 3.7: Effect of the matrix size on the spectral values and matrix norm for the Quickest scheme with the Lax-Wendroff scheme as a numerical boundary condition: (a) $\mu = 0.2, \quad \nu = 0.6$; (b) $\mu = 0.7, \quad \nu = 0.6$

the interior scheme so as to give quite accurate practical stability regions.

3.5 Accuracy and test problem

We have analysed the local truncation error of the Lax-Wendroff scheme and Quickest scheme in the first chapter. The estimates of the order of accuracy for these methods are not rigorous since there will be variation of the error with μ and ν depending on how Δx and Δt are related when the mesh is refined and also depending on the different choices of numerical boundary conditions.

We compare the effect of different numerical boundary conditions using the following test problem. If we consider the convection-diffusion problem (2.1), (3.1) and (3.2), then an exact solution of this system on the half line $x \geq 0$ is given by (3.3). Consider the initial data

$$u(x,0) = \mathrm{e}^{-x^2/L^2}, \quad x \geq 0, \quad u(0,t) = 0$$

where L is an arbitrary length scale. We will eventually take $L = 1$ but we retain it for the present to keep track of dimensions in the solution. Our reason for considering this test case is that it is straightforward to calculate an exact solution for this initial profile:

$$\begin{aligned} u(x,t) &= \frac{L}{2\sqrt{4Dt+L^2}} \left[\mathrm{e}^{-\frac{(x-Vt)^2}{4Dt+L^2}} \mathrm{Erfc}\left(-\frac{(x-Vt)L}{2\sqrt{Dt(4Dt+L^2)}} \right) \right. \\ &\quad \left. -\mathrm{e}^{-\frac{(x+Vt)^2}{4Dt+L^2}+\frac{Vx}{D}} \mathrm{Erfc}\left(\frac{(x+Vt)L}{2\sqrt{Dt(4Dt+L^2)}} \right) \right], \end{aligned} \tag{3.22}$$

where $\mathrm{Erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty \mathrm{e}^{-s^2} \mathrm{d}s$. The time evolution of the solution is shown in figure 3.8 for $L = 1$.

The following test problem results are for the section $0 \leq x \leq 20$ and for $0 \leq t \leq 20$. There seems nothing particular about these ranges which change the general nature of our conclusions.

For the initial solution $u(x,0) = \mathrm{e}^{-x^2}, V = 0.5, D = 0.001$ we compute the approximate solutions given respectively by the Lax-Wendroff scheme and by the Quickest scheme associated with the different numerical boundary conditions for a finite domain $0 \leq x \leq 20$. We plot the results in figure 3.9 at $t = 20$, for a discrete mesh $x_j = j\Delta x$, $j = 1, \ldots, 300$; $\Delta x = 20/300$ and $\Delta t = \Delta x^2$.

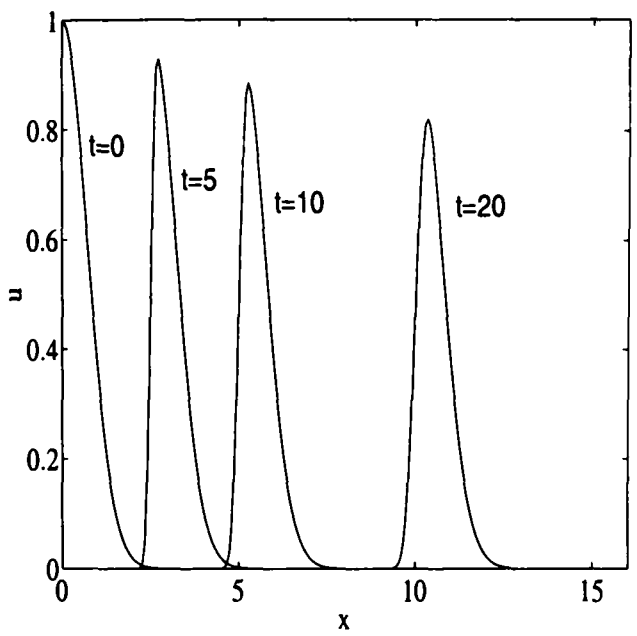


Figure 3.8: Exact solution defined by (3.22) at the times t=0, 5, 10, 20.

We can observe in figure 3.9b the typical oscillations associated with a central differencing of the convective term. As can also be seen in figure 3.9 that the Quickest scheme associated with different numerical boundary conditions gives slightly different approximate solutions and in this particular example the figure 3.9c and figure 3.9e seem to be the most accurate when compared with figure 3.9a.

Consider the vector $u_{ex} = (u(x_0,t), u(x_1,t), \ldots, u(x_N,t))$, where u is the exact solution (3.22) and the vector $U_{app} = (U(x_0,t), U(x_1,t), \ldots, U(x_N,t))$, where U is the approximated solution given by the respective numerical scheme. The error is then given by

$$\text{Error}(\Delta x) = ||u_{ex}(\Delta x) - U_{app}(\Delta x)||,$$

where $||\cdot||$ is the l_2 norm.

In figure 3.10 we plot the error versus the mesh size on a logarithmic scale for the Lax-Wendroff scheme and Quickest scheme associated with the different numerical boundary conditions.

If the Courant number ν is kept fixed and $\Delta x \to 0$, the other parameter $\mu \to \infty$ and at some point the stability boundary is reached, so we need to have a controlled refinement path with ν fixed (figure 3.10b). It is possible to have a refinement path with $\Delta x \to 0$ with μ fixed, since $\nu \to 0$ (figure 3.10a).

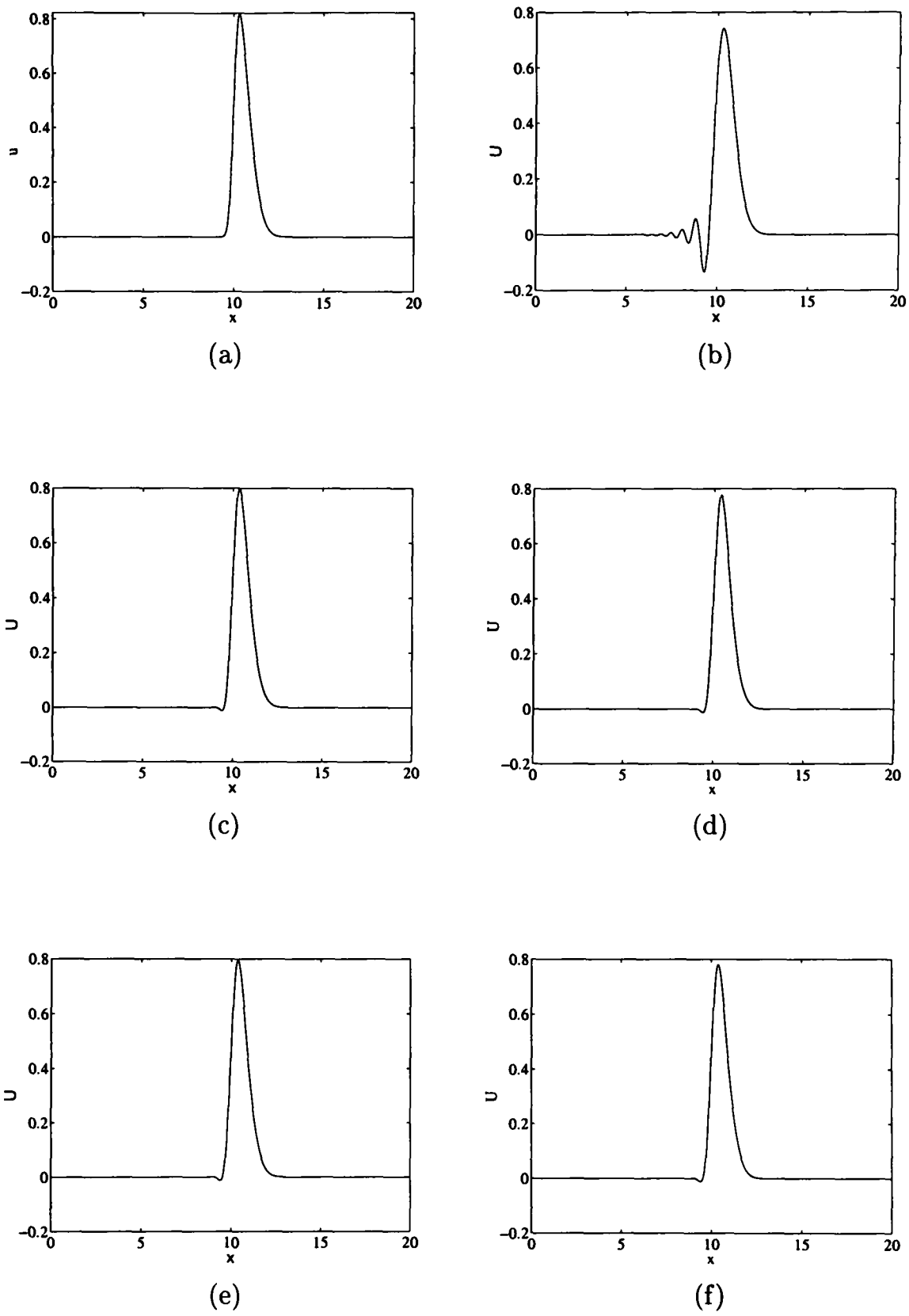
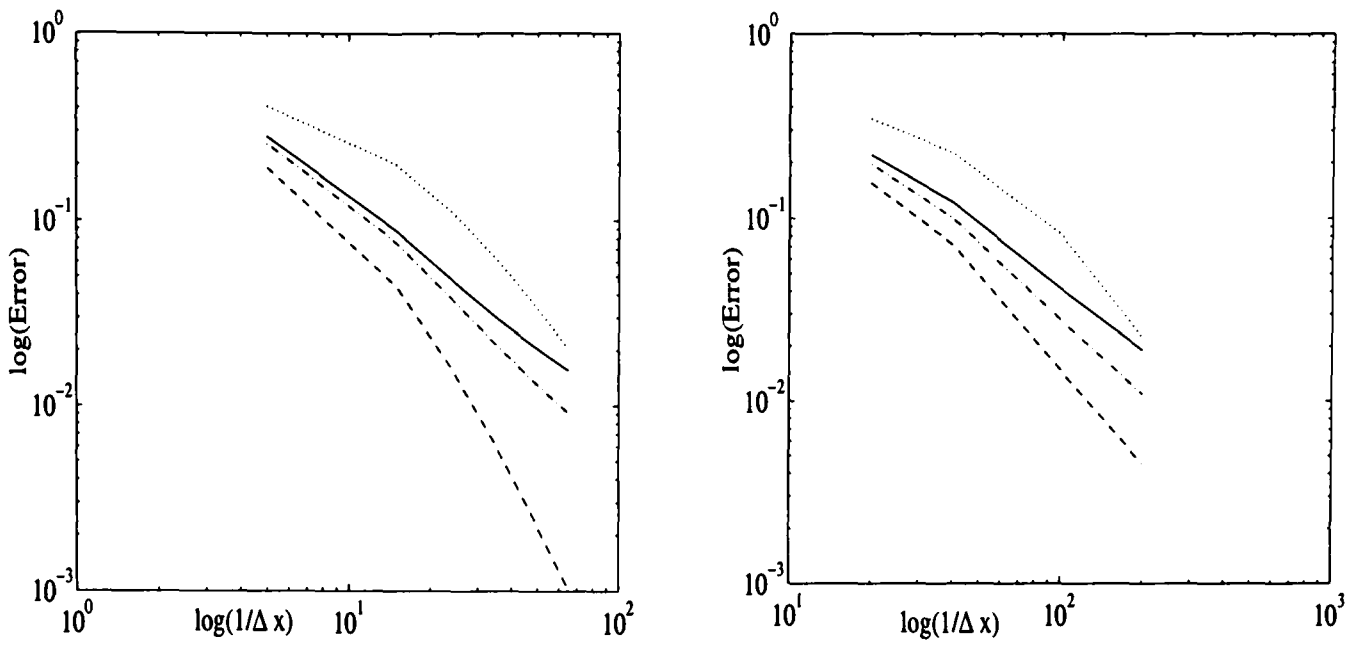


Figure 3.9: Approximated solutions and exact solution at $t = 20$. (a) Exact solution. (b) The Lax-Wendroff scheme. The Quickest scheme with the numerical boundary condition: (c) downwind third difference; (d) Leonard suggestion; (e) Lax-Wendroff; (f) fictitious point.



(a) (b)

Figure 3.10: Error function as mesh is refined for the Lax-Wendroff scheme $(\cdots)$, Quickest scheme with the respective numerical boundary conditions: Boundary suggested by Leonard (section 3.2.1) (—); Fictitious point boundary (section 3.2.4) $(-\cdot-)$; Downwind third difference boundary (section 3.2.2) and Lax-Wendroff boundary (section 3.2.3) $(- - -)$. (a) $\mu = 0.001$ (b) $\nu = 0.25$

We show results for $\mu = 0.001$ in figure 3.10a. We consider $V = 0.5$, $D = 0.001$ and $t = 20$ fixed. The Courant number ν is going to zero as we refine the mesh. The refinement is such that $\Delta t = O(\Delta x^2)$. In figure 3.10b we consider $\nu = 0.25$ fixed, $V = 0.5$, $D = 0.0001$ and $t = 20$ fixed. As we have already mentioned, when we refine the mesh with ν fixed, μ becomes larger, so we can only refine the mesh until a certain value, since μ will eventually move outside the stable region. For some methods we can refine further since the stability region is larger. The refinement is such that $\Delta t = O(\Delta x)$.

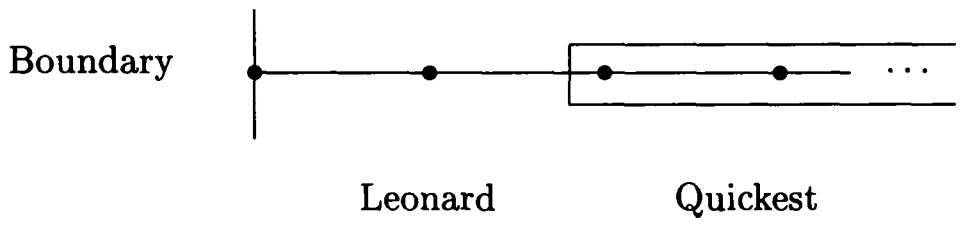
It is evident that there are some gains in accuracy by using the numerical boundary conditions described in section 3.2.2 and 3.2.3. We notice too in figure 3.10a that there is an advantage to Quickest schemes compared with Lax-Wendroff when convection is dominant, that is, μ is small.

3.6 Conclusion

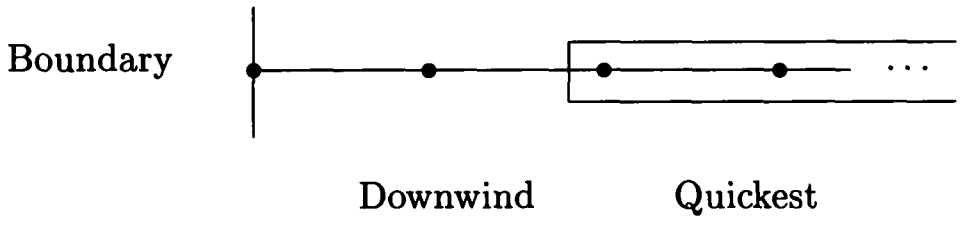
We have studied constant velocity convection-diffusion on the positive half line in order to examine how a higher order finite difference scheme can be implemented and proper account taken of a numerical boundary condition. The Lax-Wendroff scheme is a good scheme but if accuracy is a concern then a higher order scheme like Quickest is very important, but then so is the treatment of points adjacent to a boundary. The stability regions are substantially affected by the numerical boundary conditions and in the cases we have examined they can be determined quite accurately by using a von Neumann analysis associated with the spectrum and matrix analysis. When we choose the downwind third difference numerical boundary condition (section 3.2.2) or a Lax-Wendroff boundary condition (section 3.2.3) we maintain good accuracy but we lose some stability. When we require a large region of stability, the numerical boundary condition involving a fictitious point (section 3.2.4) seems to be a very good choice.

We finish this chapter summarising the numerical boundary conditions used with the Quickest scheme as a guide to the following chapters, since most of them are going to be used again in one dimension and also generalised to two dimensions.

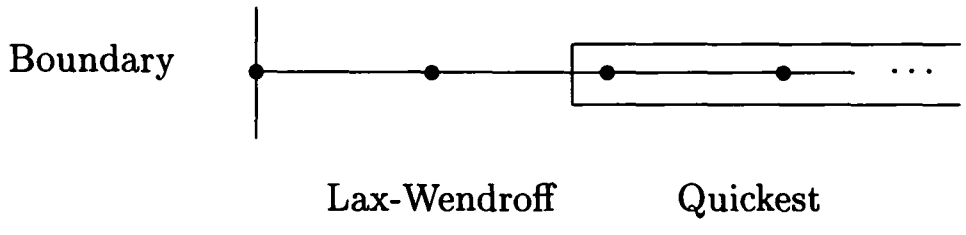
Numerical boundary condition suggested by Leonard



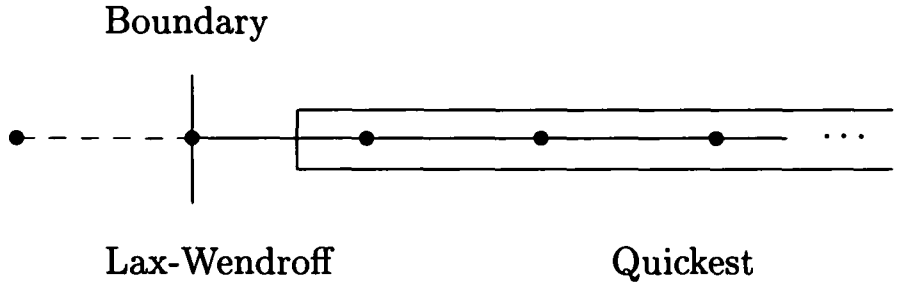
Downwind numerical boundary condition



Lax-Wendroff numerical boundary condition



Numerical boundary condition with a fictitious point



Chapter 4

Normal mode analysis

In the context of an investigation into the stability of a finite difference scheme, the von Neumann analysis is clearly very important both practically and theoretically. Von Neumann analysis is a standard method for analysing the stability of discretisations of an initial value problem on a regular structured grid. However, von Neumann analysis is only applicable to finite domain problems when these have periodic boundary conditions.

The normal mode analysis constitutes perhaps the most powerful method for local analysis of the influence of boundary conditions. This method was initially presented by Godunov and Ryabenkii [21, 22] and developed by Kreiss [35] and Osher [55]. The original Godunov-Ryabenkii theory provided a necessary condition for stability. A necessary and sufficient condition for stability was later developed by Gustafsson, Kreiss and Sundstrom [28], henceforth called GKS theory. This theory covers linear, first order hyperbolic systems in one space dimension. Since 1971, when GKS theory was first presented, related work has been done by Varah [90] for parabolic problems, by Strikwerda [78] for semi-discretised equations, by Michelson [47] for multidimensional problems and by Trefethen [84, 85] where a relation between the GKS theory and group velocity is established.

Additional work on applying the normal mode analysis can be found in Carpenter *et al* [9], Goldberg and Tadmor [24], Oliger [53, 54], Otto and Thuné [56] and Sloan [71] showing us that this method often leads to very complex calculations. In this chapter the complexity of the theoretical approach is illustrated by giving an example taken from Sousa [75].

To overcome the difficulty of the theoretical approach, recently Thuné proposed a numerical algorithm to calculate GKS stability for linear hyperbolic

equations [82] and linear hyperbolic systems [83].

We are interested in parabolic problems with convective and diffusive coefficients. The GKS theory that leads to necessary and sufficient conditions for stability was proven for hyperbolic problems. For parabolic problems we can apply the Godunov-Ryabenkii method which theoretically will give us only necessary conditions for stability although in a vast number of cases appear to be also sufficient conditions.

In this chapter we give a brief description of the Godunov-Ryabenkii theory and we apply it to the Quickest scheme with a particular numerical boundary condition. We also prove some properties related to this theory and develop an algorithm that can be applied successfully to our problem.

4.1 Godunov-Ryabenkii stability analysis

The following is an adaptation of the von Neumann method for problems subject to non periodic boundary conditions and numerical boundary conditions. One essential aspect of normal mode analysis for the investigation of the influence of boundary conditions on the stability of a scheme is that the initial value problem needs to be stable for the Cauchy problem, which is best analysed with the von Neumann method. In this section we give a general overview of the Godunov-Ryabenkii theory. For a complete description of the theory see Gustafsson *et al* [27], Richtmyer and Morton [62] and Strikwerda [78].

We present the stability theory for a quarter-plane $x, t \geq 0$. If there are two physical boundaries, then the theory shows that each boundary can be analysed separately. Thus, it is sufficient to study quarter-plane problems.

The model problem we consider is a linear and parabolic initial boundary problem. We rewrite the following equations to make this section self-contained. We have a convection-diffusion problem defined on a half-real line:

$$\frac{\partial u}{\partial t} + V\frac{\partial u}{\partial x} = D\frac{\partial^2 u}{\partial x^2} \quad 0 \leq x < \infty, \quad t \geq 0, \tag{4.1}$$

$$u(x,0) = f(x), \tag{4.2}$$

$$u(0,t) = 0, \tag{4.3}$$

$$u(\cdot,t) \in \mathrm{L}_2(0,\infty), \text{ for every fixed } t. \tag{4.4}$$

Assume we approximate the problem (4.1)–(4.4) by the difference scheme

$$U_j^{n+1} = QU_j^n, \qquad j = r, r+1, \ldots \tag{4.5}$$

$$Q = \sum_{j=-r}^{p} a_j E^j, \quad EU_j^n = U_{j+1}^n, \tag{4.6}$$

where a_{-r} and a_p are non-zero.

An important assumption is made: The finite difference scheme (4.5) is von Neumann stable and dissipative.

Definition: The scheme is dissipative if the amplification factor, z, is of the form

$$|z(\theta)| \leq 1 - \delta\theta^{2r}, \quad \text{when} \quad |\theta| < \pi,$$

for $\delta \in \mathbb{R}^+$ and positive integer r.

The requirement of the scheme to be dissipative, arises quite naturally for difference schemes for parabolic equations.

Considering the finite difference scheme (4.5) we observe that as Q uses r points upstream, the basic approximation can not be used at x_0, x_1, x_2, ..., x_{r-1}, so there we need to apply boundary conditions. In our particular case the boundary given by the physical problem is associated only with the point x_0. At the other points the boundary conditions, called numerical boundary conditions, affect the difference scheme. Let us assume that the boundary conditions can be written as

$$U_\beta^{n+1} = \sum_{j=0}^{q} l_{\beta j} U_j^n \quad \beta = 0, 1, \ldots, r-1. \tag{4.7}$$

The general technique is based on Laplace transforms of nodal values, which vary continuously with time. Therefore we define,

$$U_j(t) = U_j^n, \quad \text{for} \quad t_n \leq t \leq t_{n+1}.$$

Applying the Laplace transform

$$\tilde{U}_j(s) = \int_0^\infty e^{-st} U_j(t) dt$$

to (4.5)–(4.7) gives (see Gustafsson [27] for details),

$$z\tilde{U}_j = Q\tilde{U}_j \quad j = r, r+1, \ldots$$

$$z\tilde{U}_\beta = \sum_{j=0}^{q} l_{\beta j} \tilde{U}_j \quad \beta = 0, 1, \ldots, r-1$$

$$\tilde{U} \in l_2(0, \infty).$$

where $z = e^{s\Delta t}$ and the coefficients $l_{\beta j}$ depend on ν and μ, and we assume that μ is constant when Δx varies.

Therefore, the eigenvalue problem associated with our approximation is:

$$z\phi_j = Q\phi_j \quad j = r, r+1, \ldots \tag{4.8}$$

$$z\phi_\beta = \sum_{j=0}^{q} l_{\beta j}\phi_j \quad \beta = 0, 1, \ldots, r-1 \tag{4.9}$$

$$\phi \in l_2(0, \infty). \tag{4.10}$$

Lemma 4.1 Godunov-Ryabenkii Condition *The approximation is unstable if the eigenvalue problem (4.8)-(4.10) has an eigenvalue $z \in \mathbf{C}$ with $|z| > 1$.*

Proof: This follows from observing that if z is an eigenvalue of (4.8)-(4.10) with eigenfunction ϕ_j that it is also true that $U_j^n = z^n\phi_j$ is a solution of (4.5)-(4.7). At a fixed time t, we have

$$U_j^{t/\Delta t} = z^{t/\Delta t}\phi_j,$$

and for decreasing Δt the solution grows without bound. See Theorem 10.1.1 in Gustafsson [27]. □

Now we discuss how to solve the eigenvalue problem (4.8)-(4.10). Note that we are concerned with the behaviour of the solutions for $|z| > 1$ (Lemma 4.1). Consider the characteristic equation of the interior scheme. The characteristic equation is generated by substituting U_j^n by $z^n\kappa^j$ in the interior numerical scheme, obtaining

$$z - \sum_{j=-r}^{p} a_j\kappa^j = 0. \tag{4.11}$$

It can be proved that certain solutions κ's, of the characteristic equation (4.11) are associated with the eigenfunctions ϕ_j of the eigenvalue problem (4.8)-(4.10). We do not prove it here, but the proof can be found in Gustafsson *et al* [27]. However, the description of how the eigenfunctions ϕ_j's can be written in terms of the κ's is given below. We will frequently refer to the solutions of (4.11) as the modes κ's.

We can now explain in the context of normal mode analysis what is the meaning of the interior difference scheme being dissipative. The dissipativity condition that we assume for the interior difference scheme (4.5), tells us that $|z| < 1$ for $\kappa = e^{i\xi}, \xi \in \mathbb{R}, \xi \neq 0$. This condition is imposed to strengthen the

Godunov-Ryabenkii condition. This is because the Godunov-Ryabenkii condition is a necessary condition that does not take into account the instability mechanism associated with $|z| = |\kappa| = 1$.

The Lemma 4.1 tells us that when z is such that $|z| > 1$, then we have an instability. What follows are results that help us to find an instability by looking for an eigenvalue z such that $|z| > 1$. Therefore the following lemma describes the behaviour of the modes κ that are the solutions of the equation (4.11), when $|z| > 1$.

Lemma 4.2 *(compare Lemma 12.1.6 [27]): For $z \in \mathbf{C}$ such that $|z| > 1$, there is no solution of equation (4.11) with $|\kappa| = 1$ and there are exactly r solutions, counted according to multiplicity, with $|\kappa| < 1$.*

Proof: Assume that there is a root $\kappa = e^{i\xi}, \xi \in \mathbb{R}$. Then equation (4.11) implies $z = \sum_{j=-r}^{p} a_j e^{ij\xi}$. Because we have assumed that the approximation is von Neumann stable, we necessarily have $|z| = |\sum_{j=-r}^{p} a_j e^{ij\xi}| \leq 1$. This is a contradiction to the hypothesis $|z| > 1$; that is, there are no solutions κ with $|\kappa| = 1$. The solutions κ are continuous functions of z and cannot cross the unit circle. Therefore, the number of solutions with $|\kappa| < 1$ is constant for $|z| > 1$, and we can determine their number from the limit $z \to \infty$. In this limit, the solutions with $|\kappa| < 1$ converge to zero and are, because $a_{-r} \neq 0$, to first approximation, determined by

$$z - a_{-r}\kappa^{-r} = 0.$$

This equation has exactly r solutions $\kappa = O(z^{-1/r})$. This proves the lemma. □

We now describe how the eigenfunction ϕ_j can be written in terms of the κ's, although as already mentioned, we do not prove it here. A general solution of (4.8)–(4.10) is of the form (see Gustafsson *et al* [27]),

$$\phi_j = \sum_{|\kappa_a|<1} P_a(j)\kappa_a^j, \quad \kappa_a = \kappa_a(z), \ |z| > 1, \tag{4.12}$$

where κ_a are solutions of the characteristic equation (4.11). This solution depends on r free parameters $\sigma = (\sigma_1, \ldots, \sigma_r)$. $P_a(j)$ is a polynomial in j. Its order is at most $m_a - 1$ where m_a is the multiplicity of κ_a.

Note that if the roots κ_a are simple, this implies that the solution has the form

$$\phi_j = \sum_{|\kappa_a|<1} \sigma_a \kappa_a^j, \tag{4.13}$$

for some constants σ_a. Substituting (4.12) into the boundary conditions (4.9) yields a system of equations

$$C(z)\ \sigma = 0,$$

$\sigma = (\sigma_1, \ldots, \sigma_r)^T$, and we can rephrase Lemma 4.1 in the following form:

Lemma 4.3 *The approximation is unstable for some $z \in \mathbf{C}$ with $|z| > 1$, if*

$$\mathbf{det}\ C(z) = 0. \tag{4.14}$$

Proof: Direct application of Lemma 4.1. □

In other words, the theory states that the interior scheme needs to be von Neumann stable and when considered in the half-plane $x \geq 0$, a mode κ^j with $|\kappa| > 1$ will lead to an unbounded solution in space, that is, κ^j will increase without bound when j goes to infinity. Therefore $|\kappa|$ should be less than one, and the Godunov-Ryabenkii (necessary) stability condition states that all the modes with $|\kappa| \leq 1$, generated by the boundary conditions, should correspond to $|z| < 1$. The form of the solution is very similar to the assumed Fourier modes, except that the amplitude of the spatial oscillation decays exponentially with j away from the boundary.

In the next section we apply the Godunov-Ryabenkii theory to the Quickest scheme subject to a numerical boundary condition.

4.2 Instability of a Quickest scheme

In this section we apply the Godunov-Ryabenkii theory to find the region where the Quickest scheme is unstable when considering a numerical boundary condition, although the choice of this numerical boundary condition has already been discussed in the previous chapter. These results are taken from Sousa [75].

We approximate the problem (4.1)–(4.4) by the Quickest difference scheme (2.8) which can be written in the form:

$$U_j^{n+1} = [1 - 2c_1\Delta_0 + c_2\delta^2 + c_3\delta^2\Delta_-]U_j^n, \quad j \geq 2, \tag{4.15}$$

where $c_1 = \nu/2$, $c_2 = \nu^2/2 + \mu$ and $c_3 = (\nu/6)(1 - \nu^2 - 6\mu)$. Additionally we consider the two boundary conditions:

$$U_0^{n+1} = 0, \tag{4.16}$$

$$U_1^{n+1} = [1 - 2c_1\Delta_0 + c_2\delta^2 + c_3\delta^2\Delta_+]U_1^n. \tag{4.17}$$

We explained how to deduce the numerical boundary condition (4.17) in section 3.2.2.

In figure 4.1 and figure 4.2 we show the approximate solution given by (4.15)–(4.17) for values of μ and ν where the interior scheme (4.15) is von Neumann stable. Although the interior scheme (4.15) is von Neumann stable we can see that in figure 4.2 we have an unstable solution for $\mu = 0.6$, $\nu = 0.4$.

Consider the eigenvalue problem associated with our approximation:

$$\begin{aligned} z\phi_j &= [1 - 2c_1\Delta_0 + c_2\delta^2 + c_3\delta^2\Delta_-]\phi_j, \quad j \geq 2, && (4.18)\\ \phi_0 &= 0, && (4.19)\\ z\phi_1 &= [1 - 2c_1\Delta_0 + c_2\delta^2 + c_3\delta^2\Delta_+]\phi_1. && (4.20) \end{aligned}$$

The Godunov-Ryabenkii condition tell us that if the system (4.18)–(4.20) has an eigenvalue $z \in \mathbf{C}$ with $|z| > 1$, then the approximation (4.15) – (4.17) is unstable. The characteristic equation for the interior scheme (4.15) is given by

$$f(\kappa, z, \mu, \nu) = 0, \tag{4.21}$$

where

$$\begin{aligned} f(\kappa, z, \mu, \nu) &= \kappa^3(-c_1 + c_2 + c_3) + \kappa^2(-z + 1 - 2c_2 - 3c_3)\\ &\quad +\kappa(c_1 + c_2 + 3c_3) - c_3. \end{aligned}$$

By Lemma 4.2 we know that the equation (4.21) has exactly two solutions κ_1 and κ_2 with $|\kappa_1| < 1$ and $|\kappa_2| < 1$, for $|z| > 1$. Assuming that the two solutions κ_1 and κ_2 are distinct, the solution of (4.18)–(4.20) has the form

$$\phi_j = \sigma_1\kappa_1^j(z) + \sigma_2\kappa_2^j(z), \quad \text{for } |z| > 1. \tag{4.22}$$

Substituting (4.22) into the boundary conditions (4.16) and (4.17) yields the linear and homogeneous system

$$\begin{aligned} \sigma_1 + \sigma_2 &= 0\\ \sigma_1 g(\kappa_1, z, \mu, \nu) + \sigma_2 g(\kappa_2, z, \mu, \nu) &= 0 \end{aligned} \tag{4.23}$$

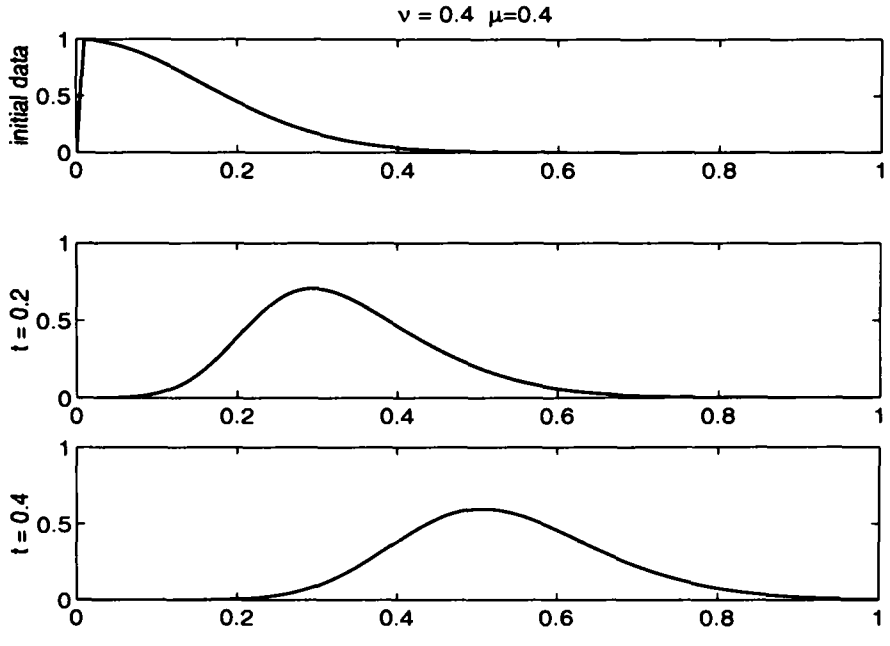


Figure 4.1: Solution for $t = 0, 0.2, 0.4$ (Stable)

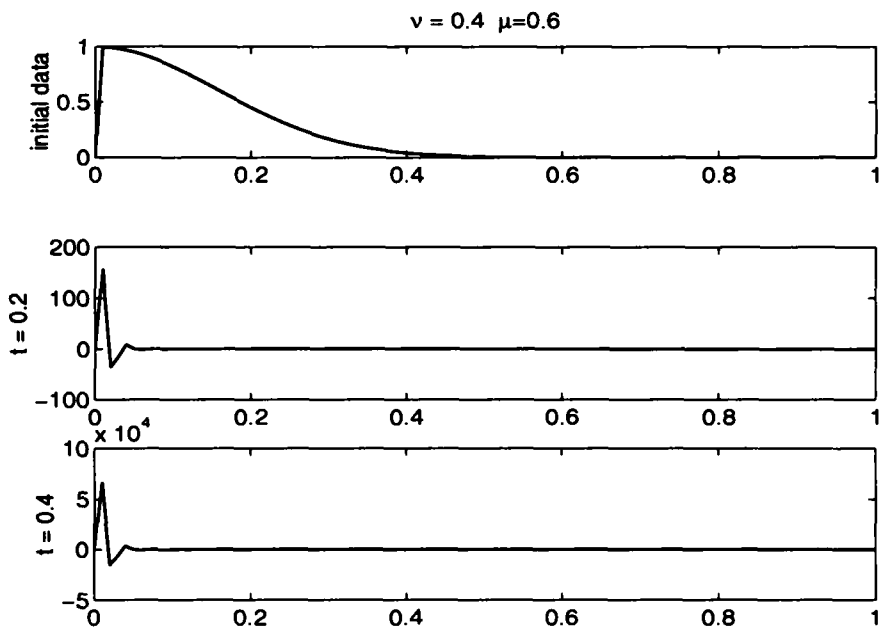


Figure 4.2: Solution for $t = 0, 0.2, 0.4$ (Unstable)

where

$$g(\kappa, z, \mu, \nu) = \kappa^3 c_3 + \kappa^2(-c_1 + c_2 - 3c_3) + \kappa(1 - 2c_2 + 3c_3 - z).$$

Since the first equation gives $\sigma_1 = -\sigma_2$, the linear homogeneous system (4.23) has a non-trivial solution if

$$g(\kappa_1, z, \mu, \nu) - g(\kappa_2, z, \mu, \nu) = 0. \tag{4.24}$$

If (4.24) is verified for some value $z = z_0$ with $|z_0| > 1$, the analysis is complete and we know that the approximation is unstable.

Consider κ_1 and κ_2 defined as:

$$\kappa_1(z, \mu, \nu) = \frac{r_1}{2} + \frac{\sqrt{-3r_1^2 + 4r_2}}{2}, \quad \kappa_2(z, \mu, \nu) = \frac{r_1}{2} - \frac{\sqrt{-3r_1^2 + 4r_2}}{2}, \tag{4.25}$$

where r_1 and r_2 are:

$$r_1(z, \mu, \nu) = \frac{(1 - z)(-c_1 + c_2 + c_3) - 4c_1c_3 + 2c_2(c_1 - c_2)}{(1 - z)c_3 - 2c_1c_3 - (c_1 - c_2)^2}, \tag{4.26}$$

$$r_2(z, \mu, \nu) = \frac{(1 - z)(z - 1 + 4c_2) - (c_1^2 + 6c_1c_3 + 3c_2^2)}{(1 - z)c_3 - 2c_1c_3 - (c_1 - c_2)^2}. \tag{4.27}$$

After some algebraic manipulations we can prove that for $c_3 \neq 0$, κ_1 and κ_2 are solutions of (4.24) and also solutions of

$$f(\kappa_1, z, \mu, \nu) - f(\kappa_2, z, \mu, \nu) = 0. \tag{4.28}$$

If additionally to (4.24) and (4.28) κ_1 and κ_2 satisfy

$$f(\kappa_1, z, \mu, \nu) + f(\kappa_2, z, \mu, \nu) = 0, \tag{4.29}$$

then we have two solutions κ_1 and κ_2 of f, that verify (4.24).

Considering the fact that κ_1 and κ_2 are given by (4.25) let

$$\begin{aligned} F(z, \mu, \nu) &= f(\kappa_1, z, \mu, \nu) + f(\kappa_2, z, \mu, \nu) \\ &= r_1(z, \mu, \nu)(3r_2(z, \mu, \nu) - 2r_1^2(z, \mu, \nu))(-c_1 + c_2 + c_3) \\ &\quad +(2r_2(z, \mu, \nu) - r_1^2(z, \mu, \nu))(-z + 1 - 2c_2 - 3c_3) \\ &\quad +r_1(z, \mu, \nu)(c_1 + c_2 + 3c_3) - 2c_3. \end{aligned}$$

For each (μ, ν) we want to find $z_{\mu\nu}$ such that

$$F(z_{\mu\nu}, \mu, \nu) = 0. \tag{4.30}$$

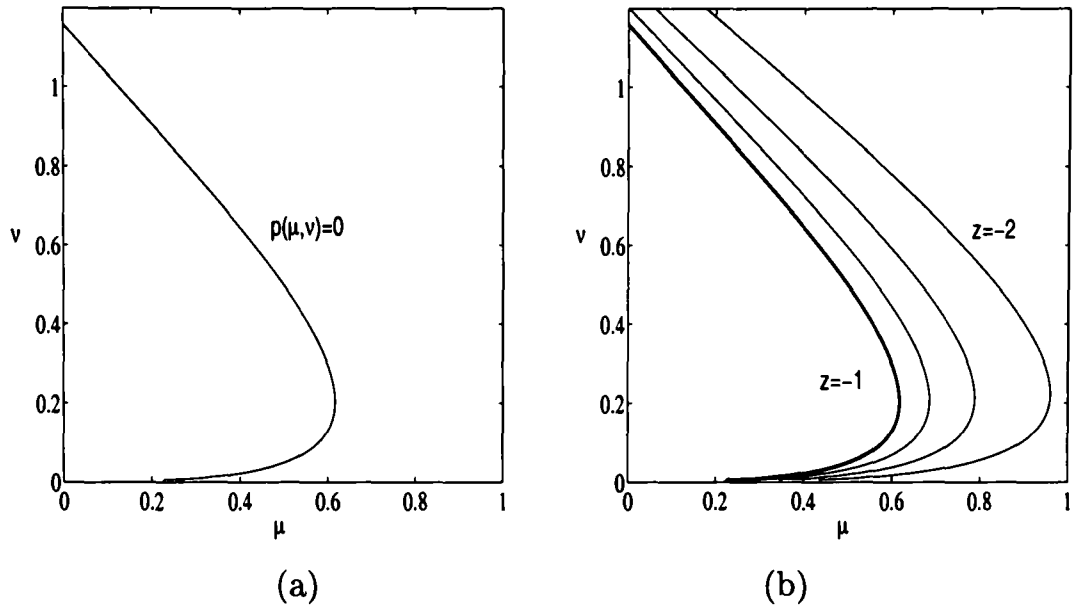


Figure 4.3: (a) $p(\mu, \nu) = 0$, where the value of $p(\mu, \nu)$ inside the curve is positive; (b) $F(z, \mu, \nu) = 0$ for $z = -1, -1.2, -1.5, -2$

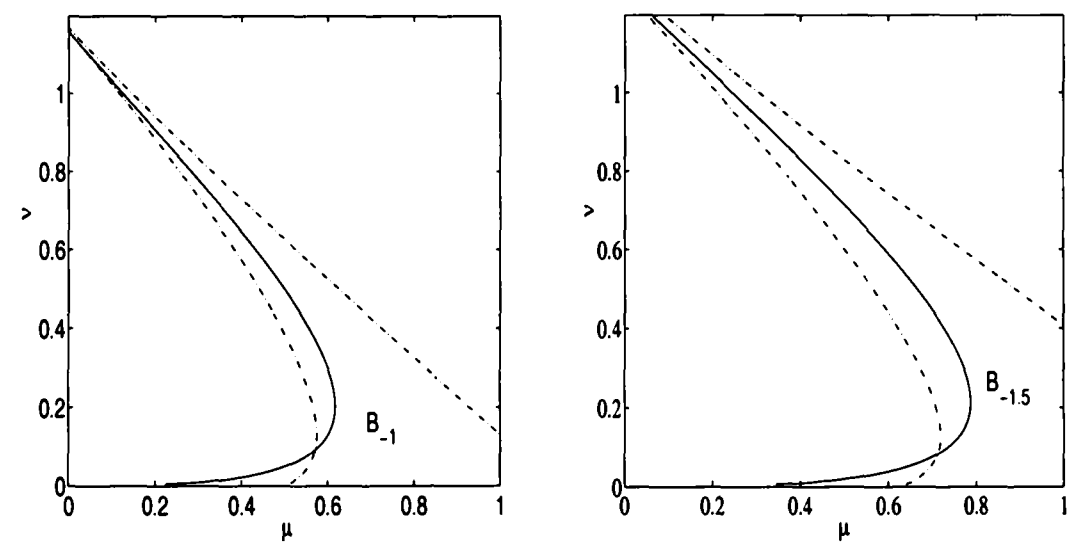


Figure 4.4: (a) $F(-1, \mu, \nu) = 0$ is the line (–) and B_{-1} is the region between the lines (---); (b) $F(-1.5, \mu, \nu) = 0$ is the line (–) and $B_{-1.5}$ is the region between the lines (---).

The requirement for instability is $|z_{\mu\nu}| > 1$. Consider a solution $z_0(\mu,\nu)$ of (4.30) that lies inside the circle $|z| = 1$ and then crosses the circle for certain variations in the values of μ and ν. Experimentally we can observe that this crossing happens at $z = -1$. We can say that $z = -1$ is the value of transition from stable to unstable.

Denote

$$p(\mu,\nu) = F(-1,\mu,\nu).$$

We plot $p(\mu,\nu) = 0$ in figure 4.3a. For (μ,ν) such that $p(\mu,\nu) < 0$ there exists a real eigenmode $z_{\mu\nu} < -1$ such that $F(z_{\mu\nu},\mu,\nu) = 0$ (see figure 4.3b). This means that for (μ,ν) such that $p(\mu,\nu) < 0$ there exists $z_{\mu\nu}$ real and less than -1 such that $\kappa_1(z_{\mu\nu},\mu,\nu)$ and $\kappa_2(z_{\mu\nu},\mu,\nu)$ are solutions of f and verify (4.24). To show that this eigenmode $z_{\mu\nu}$ has absolute value bigger than one and hence determines an unstable region we still need to verify that we do have $|\kappa_1(z_{\mu\nu},\mu,\nu)| < 1$ and $|\kappa_2(z_{\mu\nu},\mu,\nu)| < 1$.

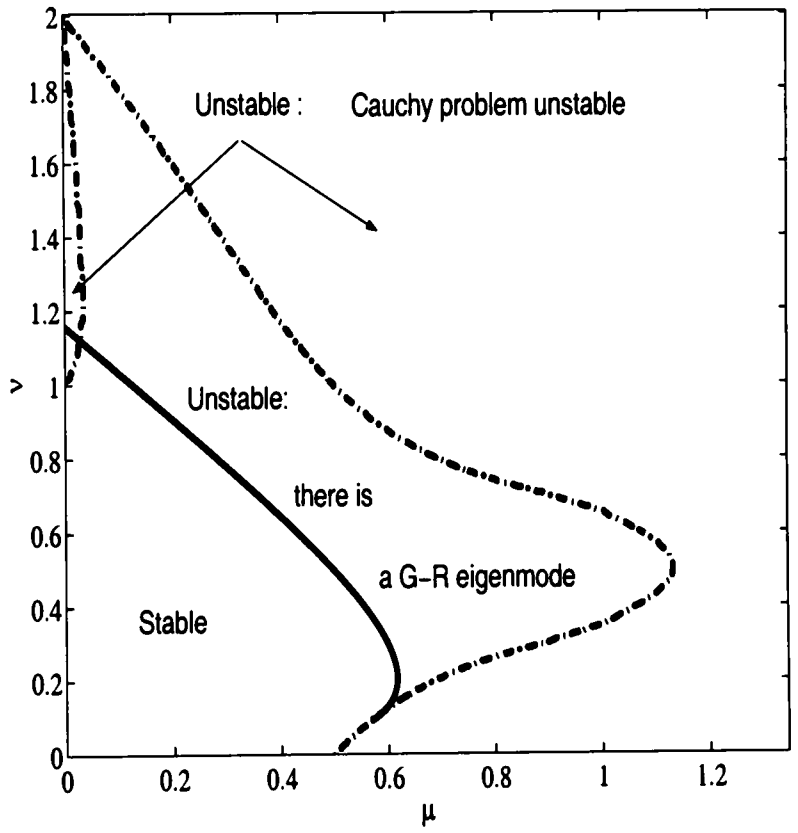


Figure 4.5: Stability region: von Neumann condition (-·-) and Godunov-Ryabenkii condition (−)

For z fixed let us define the following sets:

$$A_z = \{(\mu,\nu) : |\kappa_1(z,\mu,\nu)| < 1\} \quad \text{and} \quad B_z = \{(\mu,\nu) : |\kappa_2(z,\mu,\nu)| < 1\}.$$

For $z < -1$, $B_z \subset A_z$ that is, if $|\kappa_2(z,\mu,\nu)| < 1$ then $|\kappa_1(z,\mu,\nu)| < 1$. We plot $F(z,\mu,\nu) = 0$ and B_z for $z = -1, -1.5$ in figure 4.4. From figure 4.4 we observe that in the region B_{-1} the root $\kappa_2(-1,\mu,\nu)$, for (μ,ν) such that $p(\mu,\nu) = 0$, becomes bigger than one approximately for $\nu < 0.09$. For $z = -1.5$ the same happens but for ν even smaller. We can not conclude that the method is unstable for $\nu < 0.09$, since one of the roots we found became larger than one. On the other hand the von Neumann condition gave us a stability limit for this region as shown in figure 4.5. We plot in figure 4.5 the curve $p(\mu,\nu) = 0$ for $\nu > 0.09$ and the von Neumann stability condition, where the different causes of instability are captioned. Performing experiments numerically the region called stable in figure 4.5 is the exact region of practical stability, as already discussed in the previous chapter.

4.3 Some results on normal mode analysis

In this section we present some new results on normal mode analysis. These results are connected to the determinant condition (4.14), and they are used in the next section to develop an algorithm. The results concern properties of the modes κ and the fact that the function defined in the left part of the determinant condition (4.14) is a discontinuous function in z, since the analytical form of the function changes according to whether the κ modes present multiple roots or not. Therefore we suggest the use of another function that is continuous and has the same z roots as the previously mentioned function.

We consider a finite difference approximation of (4.1)–(4.4). The crucial parameters for the stability of the approximations are μ and ν. The equations (4.11) and (4.14) form an implicit condition on μ and ν: the approximation of (4.1)–(4.4) is practically unstable for those values of μ and ν for which (4.11) and (4.14) has solutions with $|z| > 1$, $|\kappa_a| < 1$, $a = 1, \ldots, r$.

Let g be the function

$$g(z) = \mathbf{det}\ C(z).$$

The essential task when performing the stability analysis is to find the eigenvalues $z \in \mathbf{C}$ with $|z| > 1$ that are solutions of the determinant condition

$$g(z) = 0. \tag{4.31}$$

As already explained, (4.31) is obtained by substituting (4.12) into (4.9) and the form of the general solution (4.12) changes according to the multiplicity of the roots $\kappa_a, a = 1, \ldots, r$, for each $|z| > 1$. Therefore the function g is not a

continuous function. To see it more clearly, we write the matrix $C(z)$ explicitly in both cases, when the κ's are all simple and when there are some κ's that are confluent. We denote $C_r(z) := C(z)$, for z such that the κ's are simple roots, and $C_{\text{conf}}(z) := C(z)$, for z such that the κ's are confluent, that is, some of the κ's are multiple roots.

The matrix $C_r(z)$ has the following form:

$$\begin{pmatrix} z - \sum_{j=1}^{q} l_{0j}\kappa_1^j & \cdots & z - \sum_{j=1}^{q} l_{0j}\kappa_a^j & \cdots & z - \sum_{j=1}^{q} l_{0j}\kappa_r^j \\ \vdots & & \vdots & & \vdots \\ z\kappa_1^\beta - \sum_{j=1}^{q} l_{\beta j}\kappa_1^j & \cdots & z\kappa_a^\beta - \sum_{j=1}^{q} l_{\beta j}\kappa_a^j & \cdots & z\kappa_r^\beta - \sum_{j=1}^{q} l_{\beta j}\kappa_r^j \\ \vdots & & \vdots & & \vdots \\ z\kappa_1^{r-1} - \sum_{j=1}^{q} l_{r-1j}\kappa_1^j & \cdots & z\kappa_a^{r-1} - \sum_{j=1}^{q} l_{r-1j}\kappa_a^j & \cdots & z\kappa_r^{r-1} - \sum_{j=1}^{q} l_{r-1j}\kappa_r^j \end{pmatrix}$$

Each column of the matrix $C_r(z)$ is associated with one of the κ's. We denote the columns of the matrix C_r by $\ell(\kappa_a)$, $a = 1, \ldots, r$, and the columns of the confluent matrix $C_{\text{conf}}(z)$ by $\ell_{\text{conf}}(\kappa_a)$, $a = 1, \ldots, r$. Suppose that one of the κ's, which we take to be the first one, κ_1, has multiplicity m, and the others are all simple. Then $\ell_{\text{conf}}(\kappa_a) = \ell(\kappa_a)$ for $a > m$, while for $1 < a \leq m$ we have

$$\ell_{\text{conf}}(\kappa_a) = \begin{pmatrix} -\sum_{j=1}^{q} l_{0j} j^{a-1} \kappa_a^j \\ z\kappa_a - \sum_{j=1}^{q} l_{1j} j^{a-1} \kappa_a^j \\ \vdots \\ z\beta^{a-1}\kappa_a^\beta - \sum_{j=1}^{q} l_{\beta j} j^{a-1} \kappa_a^j \\ \vdots \\ z(r-1)^{a-1}\kappa_a^{r-1} - \sum_{j=1}^{q} l_{r-1j} j^{a-1} \kappa_a^j \end{pmatrix}$$

We also give the definition of a Vandermonde matrix and a confluent Vandermonde matrix since we use it below.

A Vandermonde matrix of order r is a matrix of the form

$$V_r = V_r(\kappa_1, \kappa_2, \ldots, \kappa_r) = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \kappa_1 & \kappa_2 & \cdots & \kappa_r \\ \vdots & \vdots & & \vdots \\ \kappa_1^{r-1} & \kappa_2^{r-1} & \cdots & \kappa_r^{r-1} \end{pmatrix} \tag{4.32}$$

By a confluence of the l-th column into the a-th column we mean the following limit operation: Replace in the l-th column κ_l by $\kappa_a + \epsilon$ and subtract from it the a-th column; divide this new column by ϵ and then let $\epsilon \to 0$. The resulting matrix is denoted by V_{conf}:

$$V_{\text{conf}} = \begin{pmatrix} 1 & \cdots & 1 & 0 & 1 & \cdots & 1 \\ \kappa_1 & \cdots & \kappa_{l-1} & 1 & \kappa_{l+1} & \cdots & \kappa_r \\ \kappa_1^2 & \cdots & \kappa_{l-1}^2 & 2\kappa_a & \kappa_{l+1}^2 & \cdots & \kappa_r^2 \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ \kappa_1^{r-1} & \cdots & \kappa_{l-1}^{r-1} & (r-1)\kappa_a^{r-1} & \kappa_{l+1}^{r-1} & \cdots & \kappa_r^{r-1} \end{pmatrix}$$

In other words, V_{conf} is the same matrix as V_r except for the l-th column, which is the derivative of the a-th column. A matrix that it is obtained from (4.32) by one or more confluences of columns is called a confluent Vandermonde matrix, see Gautschi [17, 18, 19] for more details about Vandermonde matrices.

We denote by U_{bc} the vector of the approximate solution in the first r points of the mesh where we need to have numerical boundary conditions. We have that

$$U_{bc} = V(z)\ \sigma, \tag{4.33}$$

where $\sigma = (\sigma_1, \cdots, \sigma_r)^T$ are the coefficients in the solution (4.12) and $V(z)$ is the Vandermonde matrix or confluent Vandermonde matrix depending on the multiplicity of the κ's for each z.

Consider the vector $b = (b_1, \cdots, b_r)^T$ defined as

$$b = C(z)\ \sigma. \tag{4.34}$$

From (4.34) and (4.33) we obtain

$$b = C(z)V^{-1}(z)U_{bc}, \quad \text{for each } z. \tag{4.35}$$

For different eigenvalues z the multiplicity of the κ's can change and consequently the matrices $C(z)$ and $V(z)$ change accordingly. Before we prove that the matrix $C(z)V^{-1}(z)$ is a continuous matrix, we prove some useful lemmas.

Lemma 4.4 *Assume that κ_a has multiplicity of order two and that κ_l and κ_a are the two confluent roots at the eigenvalue $z = z_{\text{conf}}$. Then there is a family of matrices $E(z)$ such that:*

(a) $\lim_{z \to z_{conf}} V_r(z)E(z) = V_{\text{conf}}(z_{\text{conf}})$,

(b) $\lim_{z \to z_{\text{conf}}} C_r(z)E(z) = C_{\text{conf}}(z_{\text{conf}})$.

Proof: We have $V_r(z) = V_r(\kappa_1(z), \ldots, \kappa_l(z), \ldots, \kappa_r(z))$, for $z \neq z_{\text{conf}}$. The roots $\kappa_l(z)$ and $\kappa_a(z)$ are confluent at $z = z_{\text{conf}}$. Then

$$\kappa_l(z) \to \kappa_a(z) \quad \text{when} \quad z \to z_{\text{conf}}.$$

We can write

$$\kappa_l(z) = \kappa_a(z_{\text{conf}}) + \epsilon(z), \quad \text{for} \quad z \neq z_{\text{conf}}. \tag{4.36}$$

Consequently, to prove the condition (a) of the lemma, we can prove that there exists a family of matrices $E(\epsilon(z))$ such that:

$$\lim_{\epsilon(z) \to 0} V_r(\epsilon(z))E(\epsilon(z)) = V_{\text{conf}}(z_{\text{conf}}), \tag{4.37}$$

where $V_r(\epsilon(z)) = V_r(\kappa_1(z_{\text{conf}}), \ldots, \kappa_a(z_{\text{conf}}) + \epsilon(z), \ldots, \kappa_r(z_{\text{conf}}))$, $z \neq z_{\text{conf}}$. The $V_{\text{conf}}(z_{\text{conf}})$ is the same as the $V_r(z_{\text{conf}})$ except the l-column that is the derivative of the a-column, in order to κ_a. If we define the family of matrices $E(\epsilon(z))$ as

$$E(\epsilon(z)) = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & 1 & \cdots & -\epsilon^{-1}(z) & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & 0 & \cdots & \epsilon^{-1}(z) & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix}, \tag{4.38}$$

where $-\epsilon^{-1}(z)$ is in the a-th row and $\epsilon^{-1}(z)$ is in the l-th row, it is straightforward that the condition (4.37) is verified.

Now consider the matrix $C_r(z) = C_r(\kappa_1(z), \ldots, \kappa_l(z), \ldots, \kappa_r(z))$, for $z \neq z_{\text{conf}}$. For the same reason as above, we want to find a family of matrices $E(\epsilon(z))$, such that

$$\lim_{\epsilon(z) \to 0} C_r(\epsilon(z))E(\epsilon(z)) = C_{\text{conf}}(z_{\text{conf}}), \tag{4.39}$$

where $C_r(\epsilon(z)) = C_r(\kappa_1(z_{\rm conf}), \ldots, \kappa_a(z_{\rm conf}) + \epsilon(z), \ldots, \kappa_r(z_{\rm conf}))$, for $z \neq z_{\rm conf}$. The difference between $C_{\rm conf}(z_{\rm conf})$ and $C_r(z_{\rm conf})$ is the same difference we pointed out in the case of the Vandermonde matrices, the l-column of the confluent matrix is the derivative of the a-column, in order to κ_a. This means

$$\ell'(\kappa_a(z_{\rm conf})) \approx \frac{\ell(\kappa_l(z)) - \ell(\kappa_a(z_{\rm conf}))}{\epsilon(z)},$$

where $\ell(\kappa_i(z))$, $i = 1, \ldots, r$, denote the columns of the matrix $C_r(\epsilon(z))$. Then we have that for the family of matrices (4.38), (4.39) is verified. □

The following lemma is a generalisation of the previous one.

Lemma 4.5 *Assume that κ_1 has multiplicity of order m and that $\kappa_1, \ldots, \kappa_m$ are the confluent roots at the eigenvalue $z = z_{\rm conf}$. Then there is a family of matrices $E(z)$ such that:*

(a) $\lim_{z \to z_{conf}} V_r(z)E(z) = V_{\rm conf}(z_{\rm conf})$,

(b) $\lim_{z \to z_{\rm conf}} C_r(z)E(z) = C_{\rm conf}(z_{\rm conf})$.

Proof: This proof follows the same lines as the previous one. The roots $\kappa_1, \ldots, \kappa_m$ are confluent at $z = z_{\rm conf}$. Then for $\alpha = 2, \ldots, m$

$$\kappa_\alpha(z) \to \kappa_1(z_{\rm conf}), \quad \text{when} \quad z \to z_{\rm conf}.$$

We write,

$$\begin{aligned} \kappa_2(z) &= \kappa_1(z_{\rm conf}) + \epsilon_1(z), \\ \kappa_3(z) &= \kappa_1(z_{\rm conf}) + \epsilon_2(z), \\ &\vdots \\ \kappa_{m(z)} &= \kappa_1(z_{\rm conf}) + \epsilon_{m-1}(z). \end{aligned}$$

We want to prove that there exists a family of matrices $E(\epsilon_1(z), \ldots, \epsilon_{m-1}(z))$ such that

$$\lim_{\substack{\epsilon_1(z)\to 0 \\ \vdots \\ \epsilon_{m-1}(z)\to 0}} V_r(\epsilon_1(z), \ldots, \epsilon_{m-1}(z))E(\epsilon_1(z), \ldots, \epsilon_{m-1}(z)) = V_{\rm conf}(z_{\rm conf}) \qquad (4.40)$$

and

$$\lim_{\substack{\epsilon_1(z)\to 0 \\ \vdots \\ \epsilon_{m-1}(z)\to 0}} C_r(\epsilon_1(z),\ldots,\epsilon_{m-1}(z))E(\epsilon_1(z),\ldots,\epsilon_{m-1}(z)) = C_{\text{conf}}(z_{\text{conf}}), \quad (4.41)$$

where

$$\begin{aligned} & V_r(\epsilon_1(z),\ldots,\epsilon_{m-1}(z)) \\ = \; & V_r(\kappa_1(z_{\text{conf}}),\kappa_1(z_{\text{conf}})+\epsilon_1(z),\ldots,\kappa_1(z_{\text{conf}})+\epsilon_{m-1}(z),\ldots,\kappa_r(z_{\text{conf}})). \end{aligned}$$

The matrices $C_r(\epsilon_1(z),\ldots,\epsilon_{m-1}(z))$ are defined similarly.

The differences between $V_{\text{conf}}(z_{\text{conf}})$, $C_{\text{conf}}(z_{\text{conf}})$ and $V_r(z_{\text{conf}})$, $C_r(z_{\text{conf}})$ respectively are in column 2 to column m. The $(i+1)$-th column of the confluent matrices is the derivative of the i-th column, in order to κ_1, for $1 \le i \le m-1$.

We have the following system with $m-1$ equations and $m-1$ asymptotic approximations of the derivatives in order to κ_1:

$$\begin{aligned} \ell(\kappa_2(z)) &\approx \ell(\kappa_1(z_{\text{conf}})) + \sum_{i=1}^{m-1} \frac{\epsilon_1^i(z)}{i!}\ell^{(i)}(\kappa_1(z_{\text{conf}})) \\ \ell(\kappa_3(z)) &\approx \ell(\kappa_1(z_{\text{conf}})) + \sum_{i=1}^{m-1} \frac{\epsilon_2^i(z)}{i!}\ell^{(i)}(\kappa_1(z_{\text{conf}})) \\ \vdots & \quad \vdots \\ \ell(\kappa_m(z)) &\approx \ell(\kappa_1(z_{\text{conf}})) + \sum_{i=1}^{m-1} \frac{\epsilon_{m-1}^i(z)}{i!}\ell^{(i)}(\kappa_1(z_{\text{conf}})). \end{aligned}$$

From this fact we deduce a family of matrices $E(\epsilon_1(z),\ldots,\epsilon_{m-1}(z))$ that satisfies the conditions (4.40) and (4.41). The family of matrices $E(\epsilon_1(z),\ldots,\epsilon_{m-1}(z))$ is defined such that

$$E(\epsilon_1(z),\ldots,\epsilon_{m-1}(z)) = E_1(\epsilon_1(z))\; E_2(\epsilon_2(z))\; \ldots\; E_{m-1}(\epsilon_{m-1}(z)),$$

where

$$E_i(\epsilon_i(z)) = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & 1 & \cdots & -\epsilon_i^{-1}(z) & \cdots & 0 \\ 0 & \cdots & 0 & \cdots & \epsilon_i^{-1}(z) & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix}, \qquad 1 \le i \le m-1$$

and $-\epsilon_i^{-1}(z)$ is in the i-th row and $\epsilon_i^{-1}(z)$ is in the $(i+1)$-th row. □

Theorem 4.6 *The matrix $C(z)V^{-1}(z)$ is continuous.*

Proof: The problem of continuity of the matrix $C(z)V^{-1}(z)$ arises when for certain eigenvalues z the matrices $C(z)$ and $V(z)$ change because of the multiplicity of the κ's. Consequently the matrix is continuous if

$$\lim_{z \to z_{\text{conf}}} C_r(z)V_r^{-1}(z) = C_{\text{conf}}(z_{\text{conf}})V_{\text{conf}}^{-1}(z_{\text{conf}}), \tag{4.42}$$

where $z = z_{\text{conf}}$ is the eigenvalue where the roots κ are confluent.

The equality (4.42) is a consequence of the existence of a family of matrices $E(z)$ such that

$$\lim_{z \to z_{\text{conf}}} C_r(z)E(z) = C_{\text{conf}}(z_{\text{conf}}), \qquad \lim_{z \to z_{\text{conf}}} V_r(z)E(z) = V_{\text{conf}}(z_{\text{conf}}) \tag{4.43}$$

The result of the theorem follows directly using the results of the previous lemmas. □

Corollary 4.7 *The function*

$$d(z) = \textbf{det } C(z)/\textbf{det } V(z),$$

is continuous.

Proof: This result follows from the continuity of $C(z)V^{-1}(z)$ and the definition of the determinant. □

Furthermore

$$g(z) = 0$$

if and only if

$$d(z) = 0.$$

The function $d(z)$ has the same roots as $g(z)$, it is a continuous function and also independent of ordering of κ's. Therefore on trying to solve $g(z) = 0$ for $|z| > 1$, we can have advantages in solving $d(z) = 0$ as we will see in the next section.

4.4 The numerical algorithm

In this section we present an algorithm for the stability of linear parabolic equations defined by Godunov-Ryabenkii theory. The central point of our algorithm is that it contains a new way to find the eigenvalues $z \in \mathbf{C}$, with $|z| > 1$, that are solutions of the determinant condition (4.31) and responsible for the phenomenon of instability.

The most recent approach to the creation of an algorithm for stability investigation according to normal mode theory is given by Thuné [82, 83]. To investigate stability he used GKS theory for hyperbolic equations [82] and hyperbolic systems [83], taking advantage of the special structure of the system of algebraic equations whose solutions govern stability.

One of the difficulties associated with the determinant condition (4.31) is that there are discontinuities, due to the changes of the multiplicities of the κ's. Thuné [83] writes about the problem of multiplicities in section 5.1. There, he notes (Lemma 6) that if $\kappa(z_0)$ has multiplicity greater than one, then $g_1(z_0) = 0$, where $g_1(z)$ is the determinant for the case when all eigenvalues have multiplicity one. Thus, he always used the form g_1 of g. If, during the iterative process, a solution z_0 was found such that $g_1(z_0) = 0$, then he subsequently checked whether the corresponding κ's had multiplicity one. If they had not, he went on to formulate $g_*(z)$, the correct $g(z)$ with respect to these multiplicities. If $g_*(z_0) = 0$, then z_0 was truly a solution, otherwise z_0 was considered a false alarm.

In our case, one of the important tools that is applied to determine the roots of $g(z)$ is the choice of the function $d_r(z)$ defined by

$$d_r(z) = \mathbf{det}\ C_r(z)/\mathbf{det}\ V_r(z).$$

The function $d_r(z)$ is a meromorphic function, meaning that is analytic except for a finite number of poles. We assume these poles do not lie on the circle $|z| = 1$. Therefore we can approximate $d_r(z)$ by a Laurent series on the unit circle. Our concern is to have a good approximation of the function $d_r(z)$ around the unit circle since the onset of instability is when a root z crosses the unit circle as μ and ν change. We denote $S_{n_1}^{n_2}(z)$ the truncated Laurent series that approximates $d_r(z)$. Considering $d_r(z)$ analytic in the annulus $A = \{z : 1-\alpha_1 < |z| < 1+\alpha_2\}$, α_1, α_2 real and positive, then for $z \in A$

$$d_r(z) = \sum_{k=-\infty}^{+\infty} a_k z^k \quad \text{with} \quad a_k = \frac{1}{2\pi \mathrm{i}} \oint_C \frac{d_r(w)}{w^{k+1}} dw,$$

for all integer k, where C is the unit circle. Our approximation $S_{n_1}^{n_2}(z)$ is given by

$$S_{n_1}^{n_2}(z) = \sum_{k=-n_1}^{n_2} a_k z^k. \tag{4.44}$$

The functions $d_r(z)$ and $S_{n_1}^{n_2}(z)$ also depend on the parameters μ and ν, although we have omitted this in the notation, in the interest of clarity. In the same way the z roots of $d_r(z)$ and $S_{n_1}^{n_2}(z)$ depend on μ and ν, namely $z(\mu, \nu)$.

Our numerical algorithm is implemented using MATLAB:

for $\mu = \mu_1$ **step** μ_{step} **until** μ_s
  for $\nu = \nu_1$ **step** ν_{step} **until** ν_s
    for $\theta = 0$ **step** θ_{step} **until** 2π
      a. Check von Neumann condition, by doing $U_j^n = z^n e^{ij\theta}$ and impose $|z| \leq 1$.
    endfor
    if von Neumann unstable **then**
      b. instability.
    else
      for $\theta = 0$ **step** θ_{step} **until** 2π
        c. $z = e^{i\theta}$.
        d. Compute roots $\kappa_a(z)$ of the characteristic equation.
        e. Order roots by magnitude to find the κ's inside the unit circle.
        f. Compute $d_r(z) =$ **det** $C_r(z)/$**det** $V_r(z)$.
      endfor
      g. Compute Laurent series $S_{n_1}^{n_2}(z)$ that approximates $d_r(z)$, by using the Fast Fourier Transform algorithm incorporated in MATLAB.
      h. Compute roots of $S_{n_1}^{n_2}(z)$.
      i. Count number of roots of $S_{n_1}^{n_2}(z)$ that are less than one in modulus.
    endif
  endfor
endfor

For instance, suppose we fix ν. The roots of $d_r(z)$ are continuous functions of μ, $z(\mu)$. The instability is found when for some μ_0 there is an eigenvalue z_0 such that $|z_0(\mu_0)| > 1$. When we approximate $d_r(z)$ by $S_{n_1}^{n_2}(z)$ we determine the

instability point by counting the number of roots of $S_{n_1}^{n_2}(z)$ that are less than one in modulus, as μ changes. If the number of z's that are less than one in modulus decreases at a certain μ_0, then we have found the instability point, since one of the roots that was less than one has become larger than one.

The basic idea of our method is: for each (μ, ν) we first check the practical von Neumann stability for the difference scheme. Inside the von Neumann stability region we know the exact number of κ's, the roots of the characteristic equation, that are less than one in modulus for each z with $|z| > 1$, namely r. Once we find these roots κ, we order them by magnitude so we can select the smallest first r roots and compute the function $d_r(z)$. The next step consists not in finding the z roots of the function $d_r(z)$ but in counting the number of z roots, of the approximated function $S_{n_1}^{n_2}(z)$, that lie inside the unit circle. With this counting process we expect to detect when one of the z's crosses the unit circle.

4.5 Test problems

In this section we apply the numerical algorithm described in the previous section to some problems. By doing this we also hope to give a better understanding of how the algorithm works.

First we consider the Quickest scheme (4.15) with the Dirichlet boundary condition (4.16) and the downwind numerical boundary condition (4.17). This was the example discussed in section 4.2. The matrices $C_r(z)$ and $V_r(z)$, for this case, are given by

$$C_r(z) = \begin{pmatrix} 1 & 1 \\ g_1(\kappa_1, z, \mu, \nu) & g_1(\kappa_2, z, \mu, \nu) \end{pmatrix}, \qquad V_r(z) = \begin{pmatrix} 1 & 1 \\ \kappa_1 & \kappa_2 \end{pmatrix}$$

where κ_1 and κ_2 are the solutions of the characteristic equation (4.21) and the function $g_1(\kappa, z, \mu, \nu)$ is defined as

$$g_1(\kappa, z, \mu, \nu) = \kappa^3 c_3 + \kappa^2(-c_1 + c_2 - 3c_3) + \kappa(1 - 2c_2 + 3c_3 - z).$$

We approximate the function $d_r(z)$ by the truncated Laurent series $S_{n_1}^{n_2}(z)$ defined on (4.44). Recall that the functions $d_r(z)$ and $S_{n_1}^{n_2}(z)$ depend also on the parameters μ and ν. The coefficients a_k on (4.44) are computed using the Fast Fourier Transform algorithm in MATLAB.

Next we calculate the roots of the polynomial function $z^{n_1} S_{n_1}^{n_2}(z)$. For each μ and ν we count the roots, of this polynomial function, that are inside the unit circle and we detect that for a certain value of μ and ν one of the z roots that was inside the unit circle travels to the outside.

The fact that the polynomial function $S_{n_1}^{n_2}(z)$ approximates $d_r(z)$ on the unit circle assures us that when a root of $z^{n_1}S_{n_1}^{n_2}(z)$ crosses the unit circle, this root approximates one of the roots of $d_r(z)$.

In figure 4.6 we plot the z roots of the polynomial function $z^{15}S_{15}^{16}(z)$ for $\nu = 0.4$ and μ changing, and we can observe that at some point one of the z roots crosses the unit circle at $z = -1$. We show only the roots z with $|z| < 2$, although we have 31 roots in total. Figure 4.7 plots the output of the algorithm and we observe it is in agreement with the theoretical approach presented in section 4.2. Compare also figure 4.7 with the figure 3.3.

We also apply the numerical algorithm to the other two different numerical boundary conditions discussed in the third chapter: the Lax-Wendroff numerical boundary condition (section 3.2.3) and the numerical boundary condition using a fictitious point U_{-1} (section 3.2.4). The interior scheme is the same as in the previous example with the Dirichlet boundary condition (4.16), but since we have a different numerical boundary condition the matrix $C_r(z)$ changes. Consequently instead of the function $g_1(\kappa, z, \mu, \nu)$ in the matrix $C_r(z)$ we have the functions $g_2(\kappa, z, \mu, \nu)$ and $g_3(\kappa, z, \mu, \nu)$ associated with the Lax-Wendroff numerical boundary condition, and the numerical boundary condition using a fictitious point, respectively. They are defined as:

$$\begin{aligned} g_2(\kappa, z, \mu, \nu) &= \kappa^2(c_1 - c_2) + \kappa(z - 1 + 2c_2) \\ g_3(\kappa, z, \mu, \nu) &= \kappa^2(c_1 - c_2 - c_3) + \kappa(z - 1 + 2c_2 + 3 + \beta), \end{aligned}$$

where $\beta = (-c_1 + c_2)/(c_1 + c_2)$. We obtain the stability regions plotted in figure 4.8 and figure 4.9. In Figure 4.9 we do not have any Godunov-Ryabenkii eigenmodes since the method is stable in all the von Neumann stability region. Compare figures 4.8 and 4.9 with figures 3.5 and 3.6 respectively.

We have obtained a new algorithm for the implementation of stability analysis according to Godunov-Ryabenkii theory. We take advantage of the construction of a continuous function associated with the determinant condition. This also provides a more elegant way of treating the cases associated with the existence of multiple κ's. Although the numerical algorithm we present is efficient for the cases we are interested in, it may have some limitations in other cases, for instance if some of the z roots of $d_r(z)$ lie on the circle $|z| = 1$. Nevertheless we strongly believe it is possible to adjust the algorithm to more complex problems and that the main idea amounts to a new way of looking at the phenomenon of multiple roots.

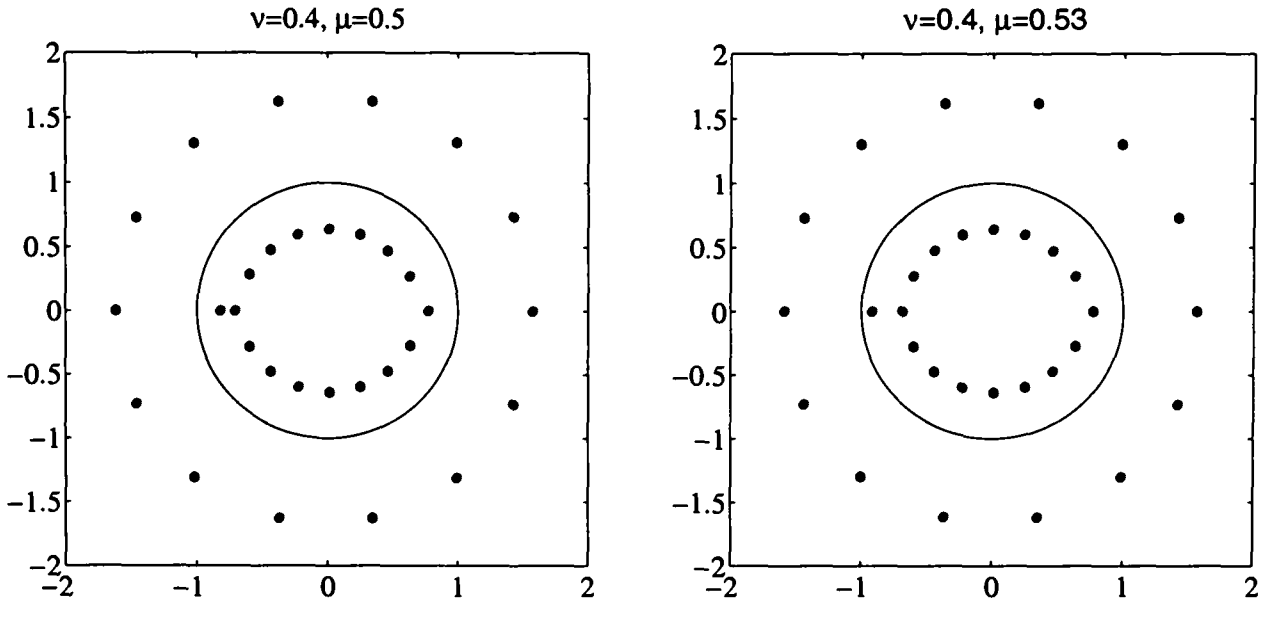


Figure 4.6: Roots of the approximated function $S_{15}^{16}(z)$ and the unit circle $|z| = 1$.

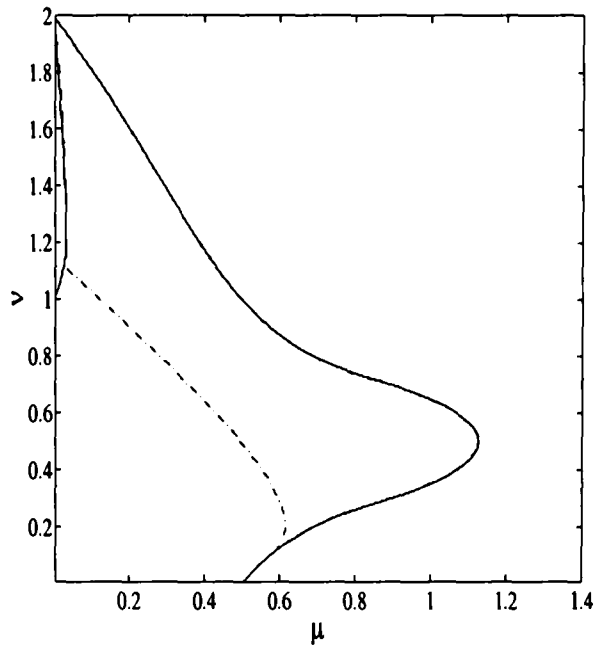


Figure 4.7: Stability region for the downwind third difference numerical boundary condition: von Neumann condition (–) and Godunov-Ryabenkii condition (- · -) using the numerical algorithm with the approximated function $S_{15}^{16}(z)$.

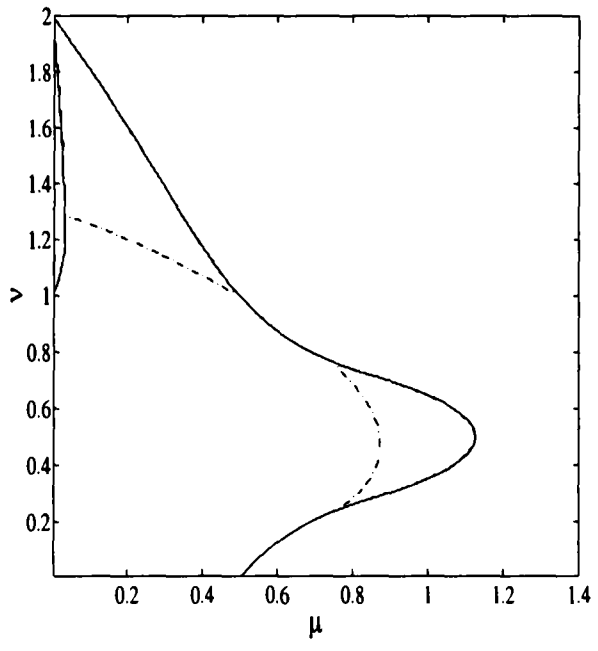


Figure 4.8: Stability region for the Lax-Wendroff numerical boundary condition: von Neumann condition (–) and Godunov-Ryabenkii condition (- · -) using the numerical algorithm with the approximated function $S_{15}^{16}(z)$.

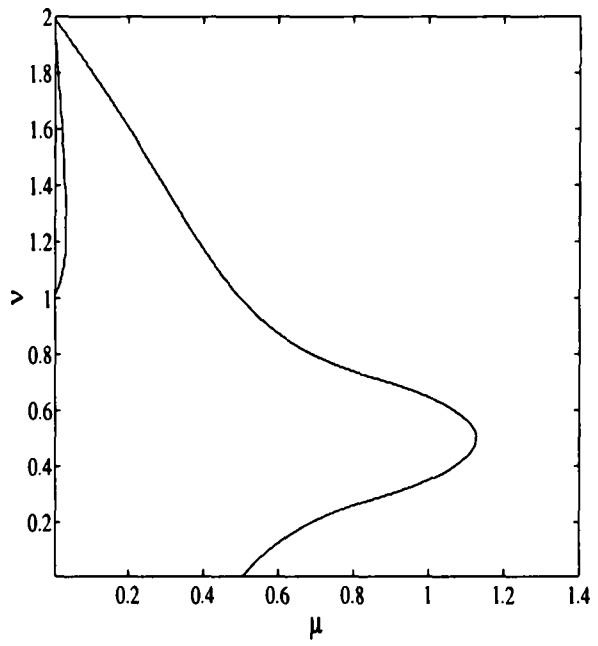


Figure 4.9: Stability region for the numerical boundary condition using a fictitious point: von Neumann condition (−) using the numerical algorithm with the approximated function $S_{15}^{16}(z)$.

Chapter 5

Finite difference methods on the half real line

Morton and Sobey [49] derived numerical schemes by using the exact evolutionary operator for a convection-diffusion problem defined on the whole line. Those numerical schemes were applied to a problem defined on the half-line in the third chapter. In this chapter, we derive new numerical schemes by using the exact evolution operator for a convection-diffusion problem defined on the half-line. We obtain a high order method that requires the use of numerical boundary conditions which are also derived using the same evolution operator. Some of the results are given in Sousa and Sobey [74].

We later try to determine whether there are advantages from the point of view of stability and accuracy in using these new schemes, when compared with the Lax-Wendroff scheme and the Quickest scheme. We conclude that the resulting schemes provide gains in terms of stability, although they do not improve the practical accuracy of existing methods.

5.1 The finite difference schemes

In the third chapter we deduced the exact solution for the convection-diffusion problem on the half line $x \geq 0$, when the convection velocity is positive so that there is an inflow boundary condition at $x = 0$ together with another boundary condition as $x \to \infty$ and an initial condition at $t = 0$. The solution of this system, defined by (2.1), (3.1) and (3.2), is given by (3.3) and this is the fundamental solution we shall use to derive approximation schemes by allowing a local solution to evolve and then restricting the evolved solution to an approximation space.

We rewrite the evolution operator over one time step, given by (3.4), in terms of the Green's function:

$$\begin{aligned} u(x,t_n+\Delta t) &= \frac{1}{\sqrt{\pi}}\int_{t_n}^{t_n+\Delta t} g(t_n+\Delta t-\tau)G^*(x,\tau)d\tau \\ &\quad +\frac{1}{\sqrt{\pi}}\int_0^{+\infty} u(\eta,t_n)G_1(x,\eta,\Delta t)d\eta, \end{aligned} \tag{5.1}$$

where

$$\begin{aligned} G^*(x,\tau) &= \frac{x}{2\sqrt{D}\tau^{2/3}}\mathrm{e}^{-(x-V\tau)^2/4D\tau}, \\ G_1(x,\eta,\beta) &= \frac{\mathrm{e}^{-(\eta-x-V\beta)^2/4D\beta}}{2\sqrt{D\beta}}[1-\mathrm{e}^{\eta x/D\beta}]. \end{aligned}$$

To derive finite difference approximations as in Morton and Sobey [49] we substitute a local polynomial approximation to $u(\eta,t_n)$ in (5.1), and then carry out the integration of a global polynomial. Suppose we have approximations $\mathbf{U}^n := \{U_j^n\}$ to the values $u(x_j,t_n)$ at the mesh points

$$x_j = j\Delta x, \quad j=0,1,2,\ldots.$$

We associate with each point x_j a local interpolating polynomial through U_j^n and the values at a certain number of neighbouring points. Denoting each such polynomial by $p_j(x;\mathbf{U}^n)$, we generate finite difference schemes from

$$\begin{aligned} U_j^{n+1} &= \frac{1}{\sqrt{\pi}}\int_{t_n}^{t_n+\Delta t} g(t_n+\Delta t-\tau)G^*(x,\tau)d\tau \\ &\quad +\frac{1}{\sqrt{\pi}}\int_0^{+\infty} p_j(\eta;\mathbf{U}^n)G_1(x_j,\eta;\Delta t)d\eta. \end{aligned} \tag{5.2}$$

The approximation scheme which we will obtain comes from approximating $\mathbf{U}^n$ near x_j by a polynomial $p_j(x;\mathbf{U}^n)$, of degree R,

$$p_j(x;\mathbf{U}^n) = \sum_{r=0}^{R} b_{jr}(x-x_j)^r.$$

Then

$$\begin{aligned} U_j^{n+1} &= \frac{1}{\sqrt{\pi}}\int_{t_n}^{t_n+\Delta t} g(t_n+\Delta t-\tau)G^*(x,\tau)d\tau \\ &\quad +\frac{1}{\sqrt{\pi}}\int_{\frac{\nu-j}{2\sqrt{\mu}}}^{+\infty} p_j(x_j-V\Delta t+2\sqrt{D\Delta t}\xi;\mathbf{U}^n)\mathrm{e}^{-\xi^2}d\xi \\ &\quad -\frac{1}{\sqrt{\pi}}\int_{\frac{\nu+j}{2\sqrt{\mu}}}^{+\infty} p_j(-x_j-V\Delta t+2\sqrt{D\Delta t}\xi;\mathbf{U}^n)\mathrm{e}^{j\nu/\mu}\mathrm{e}^{-\xi^2}d\xi. \end{aligned} \tag{5.3}$$

To deduce the new numerical schemes we use a simplified form of (5.3) by considering the boundary condition

$$u(0,t) = 0.$$

The first integral in (5.3) is zero and we can write after integration of the polynomial form,

$$\begin{aligned} U_j^{n+1} \;=\; & b_{j0}[\frac{1}{2}\mathrm{Erfc}(\frac{\nu-j}{2\sqrt{\mu}}) - \frac{1}{2}\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu+j}{2\sqrt{\mu}})] \\ & +b_{j1}[-V\Delta t\frac{1}{2}\mathrm{Erfc}(\frac{\nu-j}{2\sqrt{\mu}}) + (2x_j + V\Delta t)\frac{1}{2}\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu+j}{2\sqrt{\mu}})] \\ & +b_{j2}[(V^2(\Delta t)^2 + 2D\Delta t)\frac{1}{2}\mathrm{Erfc}(\frac{\nu-j}{2\sqrt{\mu}}) \\ & -((2x_j + V\Delta t)^2 + 2D\Delta t)\frac{1}{2}\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu+j}{2\sqrt{\mu}}) + 2\frac{\sqrt{D\Delta t}}{\sqrt{\pi}}x_j\mathrm{e}^{-(\nu-j)^2/4\mu}] \\ & +b_{j3}[-(V^3(\Delta t)^3 + 6VD(\Delta t)^2)\frac{1}{2}\mathrm{Erfc}(\frac{\nu-j}{2\sqrt{\mu}}) \\ & +(2x_j + V\Delta t)((2x_j + V\Delta t)^2 + 6D\Delta t)\frac{1}{2}\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu+j}{2\sqrt{\mu}}) \\ & -2(2V\Delta t + 3x_j)\frac{\sqrt{D\Delta t}}{\sqrt{\pi}}x_j\mathrm{e}^{-(\nu-j)^2/4\mu}] + \ldots, \end{aligned}$$

where $\mathrm{Erfc}(x)$ is the complementary error function $\mathrm{Erfc}(x) = \frac{2}{\sqrt{\pi}}\int_x^\infty \mathrm{e}^{-t^2}dt$.

Within this general framework we can now obtain finite difference schemes by interpolation on a uniform mesh. We use the usual central, backward and second difference operators to evaluate the coefficients b_{jr} in terms of the nodal values $\mathbf{U}^n$. We deduce two new numerical schemes using a quadratic interpolant and a cubic interpolant, obtaining in that way the schemes that are equivalent to the Lax-Wendroff scheme and the Quickest scheme, respectively.

Quadratic polynomial interpolation

Using the quadratic interpolant of U_{j-1}^n, U_j^n and U_{j+1}^n we have

$$b_{j0} = U_j^n, \quad b_{j1} = \frac{\Delta_0 U_j^n}{\Delta x}, \quad b_{j2} = \frac{\delta^2 U_j^n}{2\Delta x^2}.$$

The approximation formula for U_j^{n+1}, $j \geq 1$ is:

$$U_j^{n+1} = a(j)[1 - \nu\Delta_0 + (\frac{\nu^2}{2} + \mu)\delta^2]U_j^n + b(j)\Delta_0 U_j^n + c(j)\delta^2 U_j^n, \qquad (5.4)$$

where

$$\begin{aligned} a(j) &= \frac{1}{2}\mathrm{Erfc}(\frac{\nu-j}{2\sqrt{\mu}}) - \frac{1}{2}\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu+j}{2\sqrt{\mu}}) \\ b(j) &= j\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu+j}{2\sqrt{\mu}}) \\ c(j) &= -j(j+\nu)\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu+j}{2\sqrt{\mu}}) + Z(j) \end{aligned}$$

where $Z(j) = \frac{\sqrt{\mu}}{\sqrt{\pi}} j\mathrm{e}^{-(\nu-j)^2/4\mu}$. We call this scheme the Modified Lax-Wendroff scheme. Note that in (5.4), for $a(j) = 1$, $b(j) = 0$ and $c(j) = 0$, we obtain the Lax-Wendroff scheme.

Cubic polynomial interpolation

If $p_j(x, \mathbf{U}^n)$ is extended to include a cubic term, using the interpolation points U^n_{j-2}, U^n_{j-1}, U^n_j and U^n_{j+1} then

$$b_{j0} = U^n_j, \quad b_{j1} = \frac{\Delta_0 U^n_j}{\Delta x} - \frac{\delta^2\Delta_- U^n_j}{6\Delta x}, \quad b_{j2} = \frac{\delta^2 U^n_j}{2\Delta x^2}, \quad b_{j3} = \frac{\delta^2\Delta_- U^n_j}{6\Delta x^3}.$$

The approximation formula for $j \geq 2$ becomes

$$\begin{aligned} U^{n+1}_j &= a(j)[1 - \nu\Delta_0 + (\frac{\nu^2}{2} + \mu)\delta^2 + \frac{\nu}{6}(1 - \nu^2 - 6\mu)\delta^2\Delta_-]U^n_j \\ &\quad + b(j)\Delta_0 U^n_j + c(j)\delta^2 U^n_j + d(j)\delta^2\Delta_- U^n_j, \end{aligned} \tag{5.5}$$

where

$$\begin{aligned} d(j) &= \frac{1}{6}b(j) + \frac{1}{6}e(j) \\ e(j) &= (4j^3 + 6j^2\nu + 3\nu^2 j + 6j\mu)\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu+j}{2\sqrt{\mu}}) - 2(2\nu + 3j)Z(j). \end{aligned}$$

We call this scheme the Modified Quickest scheme. In (5.5), if we put $a(j) = 1$, $b(j) = 0$, $c(j) = 0$ and $d(j) = 0$ we obtain the Quickest scheme.

To obtain the scheme (5.5) we interpolate at two points upwind but we do not have these points for interpolation around the first point of the mesh. Here, therefore, we need to consider a numerical boundary condition at the first mesh point.

We consider two different possibilities for the numerical boundary condition. One is obtained by performing a cubic interpolation of the points U_0^n, U_1^n, U_2^n, U_3^n, namely

$$\begin{aligned} U_1^{n+1} &= a(1)[1-\nu\Delta_0+(\frac{\nu^2}{2}+\mu)\delta^2+\frac{\nu}{6}(1-\nu^2-6\mu)\delta^2\Delta_-]U_1^n \\ &\quad +b(1)\Delta_0 U_1^n + c(1)\delta^2 U_1^n + d(1)\delta^2\Delta_+ U_1^n, \end{aligned} \tag{5.6}$$

where Δ_+ is the forward difference operator defined by $\Delta_+ U_j^n = U_{j+1}^n - U_j^n$. The other possibility is to consider a quadratic interpolation of the points U_0^n, U_1^n, U_2^n, obtaining

$$\begin{aligned} U_1^{n+1} &= a(1)[1-\nu\Delta_0+(\frac{\nu^2}{2}+\mu)\delta^2]U_1^n \\ &\quad +b(1)\Delta_0 U_1^n + c(1)\delta^2 U_1^n. \end{aligned} \tag{5.7}$$

When $j \to \infty$ we have

$$\text{Erfc}\left(\frac{\nu-j}{2\sqrt{\mu}}\right) \to 1, \quad \text{Erfc}\left(\frac{\nu+j}{2\sqrt{\mu}}\right) \to 0, \quad Z(j) \to 0,$$

and

$$a(j) \to 1 \quad b(j) \to 0 \quad c(j) \to 0 \quad d(j) \to 0.$$

These new schemes are considerably different from the Lax-Wendroff scheme and Quickest scheme at the first point of the mesh, but are only slightly different at the other mesh points. In the next section we discuss the stability and accuracy of the new schemes.

5.2 Stability and accuracy

To analyse the stability of the new schemes we cannot use the von Neumann stability analysis since the coefficients are not uniform, although under general conditions (see Richtmyer and Morton [62]) it can be proved that for linear, non-constant coefficient problems a local von Neumann analysis will provide a necessary condition for stability. The more natural option in this case is to use the spectrum and matrix analysis based on the observation of the norm and spectrum behaviour of the iterative matrix.

Concerning accuracy, to calculate the local truncation error we cannot apply the modified equation as described in Warming and Hyett [91], since we have non-uniform terms. On the other hand we can derive formal truncation error estimates in the same way as suggested in Morton and Sobey [49] by applying the Peano kernel theorem (see Powell [58]).

5.2.1 Stability analysis of the new schemes

In section 3.3 we gave a brief overview of the stability analysis using the spectrum and matrix analysis. We apply that framework in this section. Our numerical methods have the form:

$$U^{n+1} = AU^n,$$

where A is an $N \times N$ matrix and $U^n = (U_1^n, \ldots, U_N^n)$. To build the matrix A in addition to the inflow boundary condition

$$U_0^{n+1} = 0,$$

we consider the outflow boundary condition

$$U_{N+1}^{n+1} = 0.$$

Our stability analysis consists essentially in applying the sufficient condition for stability $||A|| \leq 1$ with the necessary stability condition $\rho(A) \leq 1$. We will plot the regions using MATLAB and for the matrix size $N = 30$.

The stability region for the scheme derived using a quadratic polynomial approximation, which we called the Modified Lax-Wendroff scheme, is given by figure 5.1. Comparing figure 5.1 with figure 3.1, where we have plotted the Lax-Wendroff stability region, we observe that the stability region is the same, assuming it to be the region where $||A|| \leq 1$. In this case we do not have an advantage in terms of stability by choosing the Modified Lax-Wendroff scheme instead of the Lax-Wendroff scheme.

Consider the scheme (5.5), derived using a cubic polynomial approximation and called Modified Quickest scheme. First we analyse this scheme when associated with the numerical boundary condition (5.6). This numerical boundary condition for $a(1) = 1$ and $b(1) = c(1) = d(1) = 0$ is the same as the downwind numerical boundary condition given by (3.9). The stability region is given by the region plotted in figure 5.2. Comparing figure 5.2a with figure 3.3a we can see a significant advantage in the stability region of the new scheme. Additionally, we plot in figure 5.2b the practical von Neumann stability region for the Quickest scheme. This allows us to see in what way it relates with the region $||A|| \leq 1$, where A is the iterative matrix of the Modified Quickest scheme with the numerical boundary condition (5.6). The region for small μ where $||A|| \geq 1$ and $\rho(A) \leq 1$, plotted in the left corner of figure 5.2a and figure 5.2b is a region of practical stability. This was checked by running numerical experiments on this region.

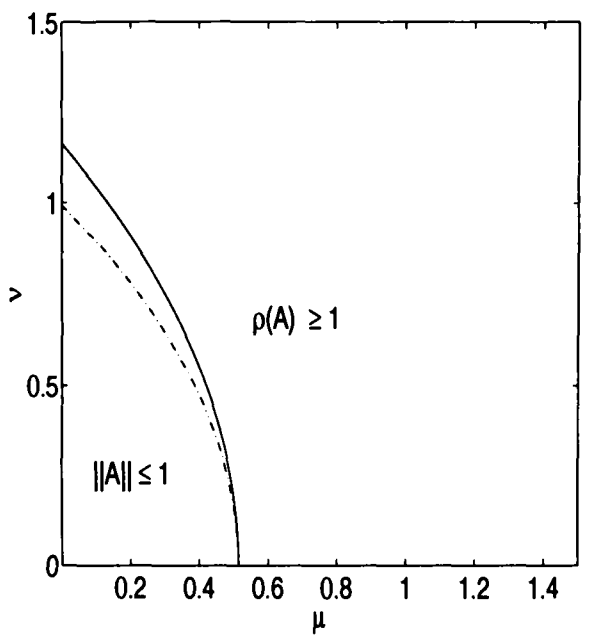
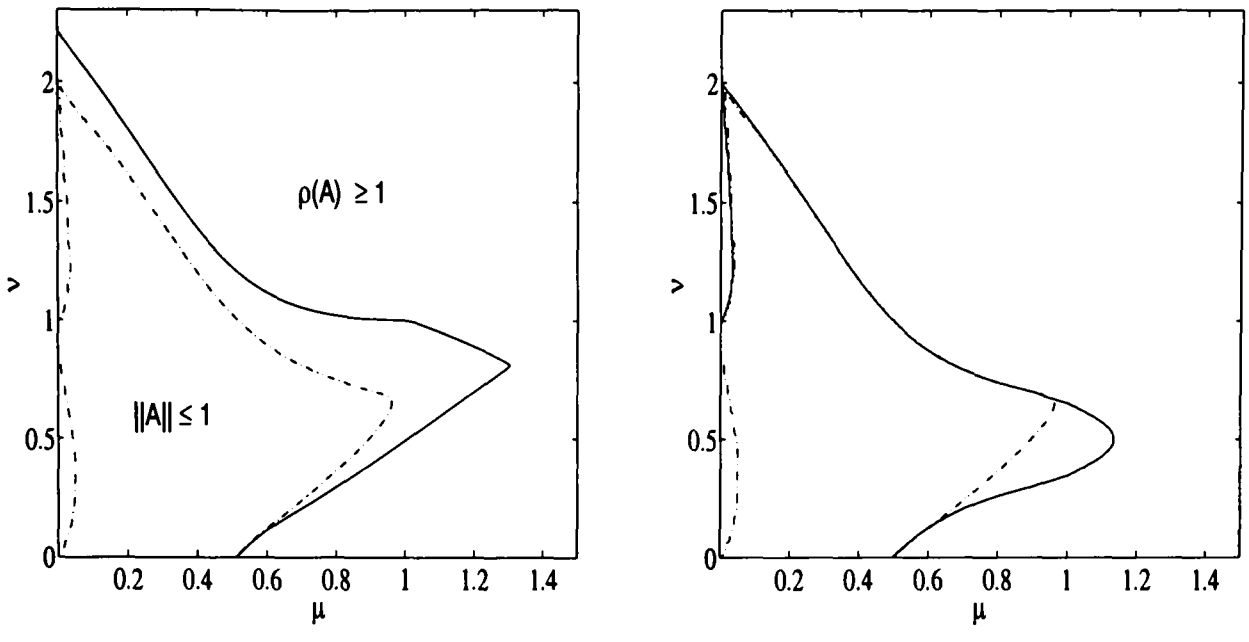


Figure 5.1: Stability region for the Modified Lax-Wendroff scheme: $\rho(A) = 1$ $(-)$ and $||A|| = 1$ $(- \cdot -)$.

In figure 5.3 we plot the Modified Quickest scheme (5.5) associated with the numerical boundary condition (5.7). This numerical boundary condition for $a(1) = 1$ and $b(1) = c(1) = 0$ is the same as the Lax-Wendroff numerical boundary condition given by (3.10). We can see that there is not a large gain in stability, from the choice of the previously considered numerical boundary condition. However, if we compare it with the Quickest scheme associated with the Lax-Wendroff numerical boundary condition as plotted in figure 3.5, there is a significant difference. We also plot the region $||A|| \leq 1$ of the Modified Quickest scheme together with the von Neumann stability region of the Quickest scheme in figure 5.3b. As in the previous case the region for $||A|| \leq 1$ and $\rho(A) \geq 1$ on the left corner of figure 5.3a and figure 5.3b is a stable region.

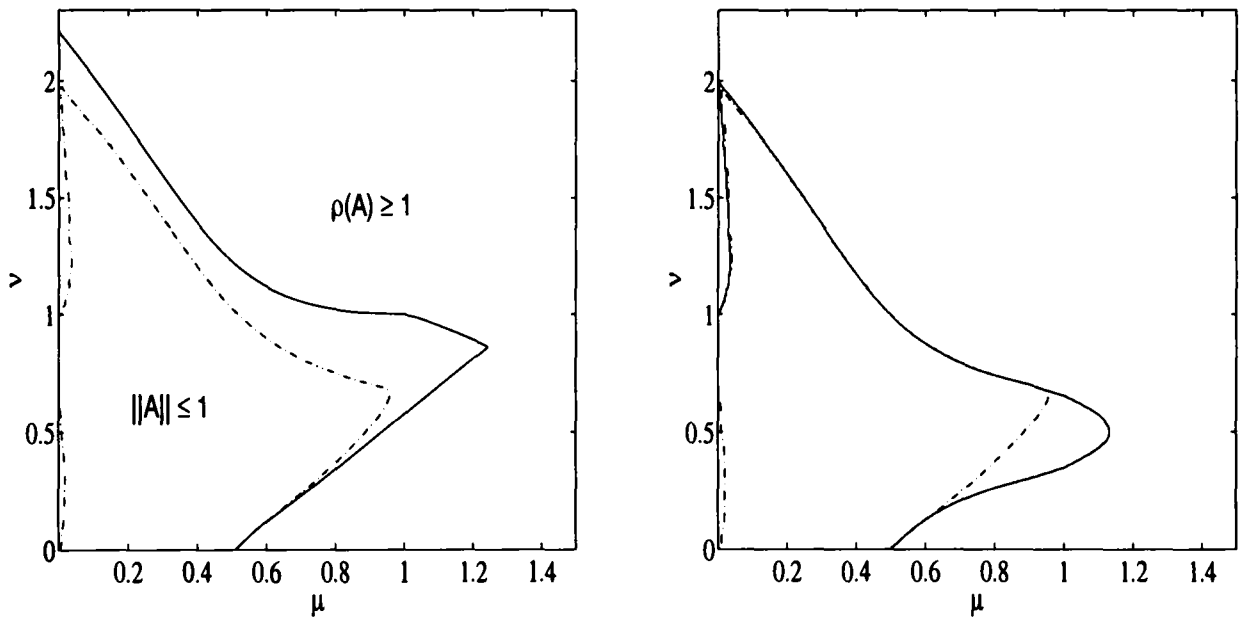
It is important to remember that $||A|| \leq 1$ is a sufficient condition for stability but not a necessary one. In fact, experimentally the new schemes seem to be stable in all the von Neumann region displayed in figure 5.2b and figure 5.3b respectively.

To conclude, we observe there are advantages in terms of stability in using the Modified Quickest scheme associated with the suggested numerical boundary conditions, when compared with the Quickest scheme associated with the downwind numerical boundary condition and Lax-Wendroff numerical boundary condition respectively.



(a) (b)

Figure 5.2: Stability region for the Modified Quickest scheme with the numerical boundary condition (5.6): (a)$\rho(A) = 1$ $(-)$ and $||A|| = 1$ $(- \cdot -)$; (b) $||A|| = 1$ $(- \cdot -)$ and practical von Neumann stability for the Quickest scheme $(-)$



(a) (b)

Figure 5.3: Stability region for the Modified Quickest scheme with the numerical boundary condition (5.7): (a) $\rho(A) = 1$ $(-)$ and $||A|| = 1$ $(- \cdot -)$; (b) $||A|| = 1$ $(- \cdot -)$ and practical von Neumann stability for the Quickest scheme $(-)$

5.2.2 Accuracy of the new schemes

We can derive truncation error estimates in the same way as suggested in Morton and Sobey [49] by applying the Peano kernel theorem (see Powell [58]).

We consider the error committed in one time step. In appendix A, the Peano kernel theorem is used to obtain the truncation error for the Modified Lax-Wendroff scheme and the Modified Quickest scheme. The following expressions for the local truncation error are obtained.

For the scheme derived using the quadratic interpolant, the Modified Lax-Wendroff, we have

$$\Delta t T|_{x_j} = \frac{1}{6}\Delta x^3 u_{xxx} g_j^2(\nu,\mu) + O(\Delta x^4 u_{x^4}), \tag{5.8}$$

where

$$g_j^2(\nu,\mu) = \nu(1-\nu^2-6\mu)a(j) - b(j) + e(j).$$

When $j \to \infty$ then $a(j) = 1, b(j) = e(j) = 0$ and the expression (5.8) is the truncation error obtained for the Lax-Wendroff scheme. Consequently, near the boundary we shall have a different truncation error which does not have an inferior order.

For the Modified Quickest scheme, obtained using a cubic interpolant, the exact error for $x_j, j \geq 2$ is given by the cumbersome expression

$$\Delta t T|_{x_j} = \frac{1}{24}\Delta x^4 u_{xxxx} g_j^3(\nu,\mu) + \cdots, \tag{5.9}$$

with

$$\begin{aligned} g_j^3(\nu,\mu) \;=\; & (12\mu^2 - 2\mu - 12\mu\nu(1-\nu) + \nu(1-\nu^2)(2-\nu))a(j) \\ & -2b(j)(12\mu\nu + 2\nu^3 + 2j\nu + 1 + 2j^3) \\ & +2c(j)(1+6j^2) + 2(1-2j)e(j) + 2Z(j)(3\nu^2 + j^2 + 10\mu). \end{aligned}$$

For $j \to \infty$ as well, the expression (5.9) is the truncation error for the Quickest scheme.

To check the accuracy numerically we considered the same test problem as in section 3.5 with the initial condition $u(x,0) = e^{-x^2}, x \geq 0$ and $u(0,t) = 0$. We computed approximate solutions for μ fixed and ν approaching zero as Δt and Δx were refined. When considering the solution for a finite domain in $0 \leq x \leq 20$ at $t = 20$ for $V = 0.5$, $D = 0.001$ and for various values of fixed μ, we did not obtain better results in the error calculation for the new schemes

compared to the corresponding ones in section 3.5, Lax-Wendroff scheme and Quickest scheme with the respective numerical boundary conditions.

Considering the fact that our numerical experiments showed the same error for the new schemes as for the schemes already analysed in the third chapter, we do not plot them here since the plots are essentially the same as the ones displayed there.

These results indicate that the new schemes offer no real advantage in terms of accuracy. This raises the question whether by using the Quickest scheme, instead of the Modified Quickest scheme, with the numerical boundary conditions suggested in this chapter, we would still have gains in stability. This could allow us to check if the advantage obtained in stability is due mainly to the numerical boundary conditions rather than to the use of the Modified Quickest scheme. We check this possibility in the next section.

5.2.3 Stability of mixed schemes

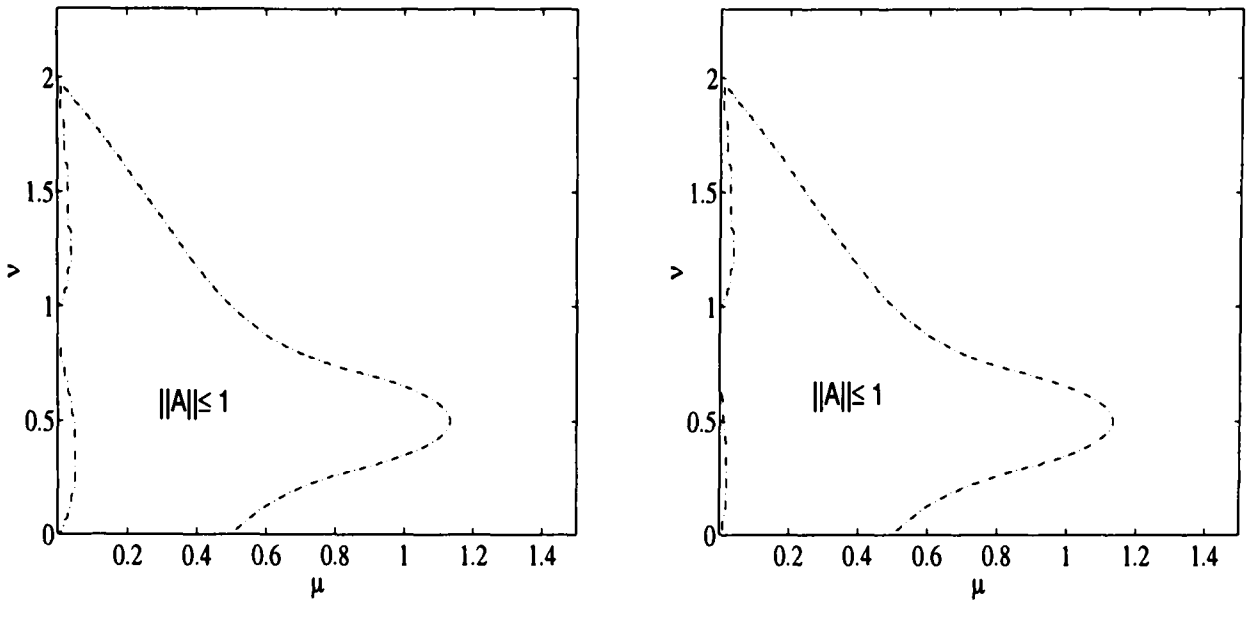
We have already seen, in the two previous chapters, that the choice of numerical boundary conditions may strongly affect the stability of a numerical scheme even if the accuracy is not affected. In this chapter this is confirmed again when we consider the two additional numerical boundary conditions studied in this chapter.

When $j \to \infty$ our new schemes are identical to the existing schemes. It is in the first point of the scheme that a considerable difference may occur and this seems to affect the stability but not the accuracy. This fact motivates us to suggest the use of the Quickest scheme together with the numerical boundary conditions (5.6) and (5.7) described in this chapter.

We plot, in figure 5.4, the region $||A|| \leq 1$, where A is the iteration matrix for the Quickest scheme (2.8) associated with the numerical boundary conditions (5.6) and the numerical boundary condition (5.7) respectively.

We observe that we recover the stability region lost when using the downwind numerical boundary condition and Lax-Wendroff numerical boundary condition mentioned in the third chapter with this same interior Quickest scheme (2.8).

In the next section we compute the analytical solution for a two-dimensional convection-diffusion problem defined in the half-plane, $x \geq 0, y \in \mathbb{R}$. We show how to deduce new schemes in two dimensions in a similar way as we have done for the one-dimensional case, although we will not pursue a further study of those schemes.



(a) (b)

Figure 5.4: Stability region for the Quickest scheme with: (a) the numerical boundary condition (5.6); (b) the numerical boundary condition (5.7);

5.3 Extension to a two dimensional case

Before we deduce the new schemes in two dimensions, we compute the analytical solution for the initial value problem, defined in $x \in \mathbb{R}^+$, $y \in \mathbb{R}$, $t \in \mathbb{R}^+$, by applying Fourier transforms in y and Laplace transforms in t.

Consider the convection-diffusion equation

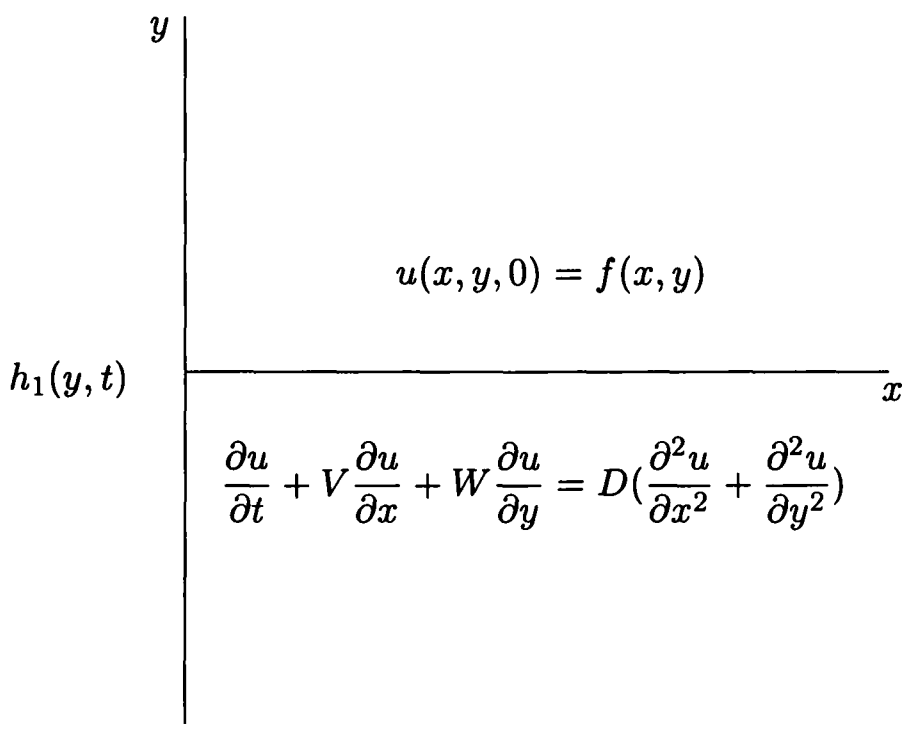
$$\frac{\partial u}{\partial t}(x,y,t) + V\frac{\partial u}{\partial x}(x,y,t) + W\frac{\partial u}{\partial y}(x,y,t) = D\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)(x,y,t) \quad (5.10)$$

subject to conditions

$$\lim_{x\to\infty} u(x,y,t) = 0,\ x \geq 0,\ \ y \in \mathbb{R},\ t \geq 0, \quad (5.11)$$

$$u(x,y,0) = f(x,y),\ x > 0,\ y \in \mathbb{R}, \quad (5.12)$$

$$u(0,y,t) = h(y,t),\ y \in \mathbb{R},\ t \geq 0. \quad (5.13)$$



As in one dimension, if we choose uniform spatial steps Δx and Δy and time step Δt, we have the dimensionless quantities:

$$\nu_x = \frac{V\Delta t}{\Delta x}, \qquad \nu_y = \frac{W\Delta t}{\Delta y},$$
$$\mu_x = \frac{D\Delta t}{\Delta x^2}, \qquad \mu_y = \frac{D\Delta t}{\Delta y^2}.$$

The system (5.10) - (5.13) can be solved exactly using Laplace transform in t, denoted by $\mathcal{L}$, and Fourier transforms in y, denoted by $\mathcal{F}$. We define

$$\begin{aligned}\overline{u}(x,k,s) &= \mathcal{L}\mathcal{F}[u(x,y,t)](x,k,s) \\ &= \frac{1}{2\pi}\int_0^{+\infty}\int_{-\infty}^{+\infty} \mathrm{e}^{-st}\mathrm{e}^{\mathrm{i}ky}u(x,y,t)dydt.\end{aligned}$$

Let $\hat{f}(x,k) := \mathcal{F}[f(x,y)](k)$. Applying $\mathcal{L}$ and $\mathcal{F}$ to the convection-diffusion equation we have the ordinary differential equation,

$$D\frac{d^2}{dx^2}\overline{u}(x,k,s) - V\frac{d}{dx}\overline{u}(x,k,s) - (s - \mathrm{i}Wk + Dk^2)\overline{u}(x,k,s) = -\hat{f}(x,k), \quad (5.14)$$

to solve, subject to the conditions,

$$\lim_{x\to\infty}\overline{u}(x,k,s) = 0,$$

$$\begin{aligned} \hat{u}(x,k,0) &= \hat{f}(x,k), \\ \overline{u}(0,k,s) &= \overline{h}(k,s). \end{aligned} \tag{5.15}$$

After we have obtained $\overline{u}(x,k,s)$ from (5.14), we can apply the inverse of Laplace transform and Fourier transform to it, i.e., $u(x,y,t) = \mathcal{F}^{-1}\mathcal{L}^{-1}[\overline{u}(x,k,s)]$, so we get the exact solution given by

$$u(x,y,t) = \frac{1}{\pi}\int_0^t d\tau \int_{-\infty}^{+\infty} h(y-\beta, t-\tau)G(x,\tau,\beta)d\beta$$

$$+\frac{1}{\pi}\int_{\frac{Vt-x}{2\sqrt{Dt}}}^{+\infty} d\xi \int_{-\infty}^{+\infty} f(x-Vt+2\sqrt{Dt}\xi, y-Wt-2\sqrt{Dt}\tau)\mathrm{e}^{-\xi^2-\tau^2}d\tau$$

$$-\frac{\mathrm{e}^{\frac{Vx}{D}}}{\pi}\int_{\frac{Vt+x}{2\sqrt{Dt}}}^{+\infty} d\xi \int_{-\infty}^{+\infty} f(-x-Vt+2\sqrt{Dt}\xi, y-Wt-2\sqrt{Dt}\tau)\mathrm{e}^{-\xi^2-\tau^2}d\tau$$

where

$$G(x,\tau,\beta) = \frac{x}{4\tau^2 D}\mathrm{e}^{-(x-V\tau)^2/4D\tau}\mathrm{e}^{-(\beta-W\tau)^2/4D\tau}.$$

The evolution over a single time-step $t_{n+1} = t_n + \Delta t$ is

$$u(x,y,t_n+\Delta t) = \frac{1}{\pi}\int_{t_n}^{t_{n+1}} d\tau \int_{-\infty}^{+\infty} h(y-\beta, t_n+\Delta t-\tau)G(x,\tau,\beta)d\beta$$

$$+\frac{1}{\pi}\int_{\frac{V\Delta t-x}{2\sqrt{D\Delta t}}}^{+\infty} d\xi \int_{-\infty}^{+\infty} f(x-V\Delta t+2\sqrt{D\Delta t}\xi, y-W\Delta t+2\sqrt{D\Delta t}\tau)\mathrm{e}^{-\xi^2-\tau^2}d\tau$$

$$-\frac{\mathrm{e}^{\frac{Vx}{D}}}{\pi}\int_{\frac{V\Delta t+x}{2\sqrt{D\Delta t}}}^{+\infty} d\xi \int_{-\infty}^{+\infty} f(-x-V\Delta t+2\sqrt{D\Delta t}\xi, y-W\Delta t+2\sqrt{D\Delta t}\tau)\mathrm{e}^{-\xi^2-\tau^2}d\tau$$

We denote U_{jk}^n the approximations to the values $u(x_j, y_k, t_n)$ at the mesh points

$$(x_j, y_k) = (j\Delta x, k\Delta y), \quad j = 0,1,2,\ldots; k \in \mathbb{Z}.$$

Considering $h(y,t) = 0$ if we approximate f by a polynomial around the point (x_j, y_k)

$$p_{jk}(x,y) = \sum_{r=0}^{K}\sum_{s=0}^{K} b_{rs}(x-x_j)^r(y-y_k)^s$$

and using the exact evolutionary operator, then the approximation U_{jk}^{n+1} will be given by

$$
\begin{aligned}
U_{jk}^{n+1} &= \sum_{r,s=0}^{K} \frac{b_{rs}}{\pi} \left\{ \int_{\frac{V\Delta t - x_j}{2\sqrt{D\Delta t}}}^{+\infty} d\xi \right. \\
&\int_{-\infty}^{+\infty} (-V\Delta t + 2\sqrt{D\Delta t}\xi)^r (-W\Delta t + 2\sqrt{D\Delta t}\tau)^s \mathrm{e}^{-\xi^2-\tau^2} d\tau \\
&-\mathrm{e}^{\frac{Vx_j}{D}} \int_{\frac{V\Delta t + x_j}{2\sqrt{D\Delta t}}}^{+\infty} d\xi \int_{-\infty}^{+\infty} (-2x_j - V\Delta t + 2\sqrt{D\Delta t}\xi)^r \\
&\left. (-W\Delta t + 2\sqrt{D\Delta t}\tau)^s \mathrm{e}^{-\xi^2-\tau^2} d\tau \right\}
\end{aligned}
$$

We can write,

$$
\begin{aligned}
U_{jk}^{n+1} &= b_{00}[\frac{1}{2}\mathrm{Erfc}(\frac{\nu_x - j}{2\sqrt{\mu_x}}) - \frac{1}{2}\mathrm{e}^{\frac{\nu_x j}{\mu_x}} \mathrm{Erfc}(\frac{\nu_x + j}{2\sqrt{\mu_x}})] \\
&+b_{10}[-V\Delta t\frac{1}{2}\mathrm{Erfc}(\frac{\nu_x - j}{2\sqrt{\mu_x}}) + (2x_j + V\Delta t)\frac{1}{2}\mathrm{e}^{\frac{\nu_x j}{\mu_x}} \mathrm{Erfc}(\frac{\nu_x + j}{2\sqrt{\mu_x}})] \\
&+b_{01}[-W\Delta t(\frac{1}{2}\mathrm{Erfc}(\frac{\nu_x - j}{2\sqrt{\mu_x}}) - \frac{1}{2}\mathrm{e}^{\frac{\nu_x j}{\mu_x}} \mathrm{Erfc}(\frac{\nu_x + j}{2\sqrt{\mu_x}}))] \\
&+b_{20}[(V^2(\Delta t)^2 + 2D\Delta t)\frac{1}{2}\mathrm{Erfc}(\frac{\nu_x - j}{2\sqrt{\mu_x}}) \\
&-((2x_j + V\Delta t)^2 + 2D\Delta t)\frac{\mathrm{e}^{\frac{\nu_x j}{\mu_x}}}{2}\mathrm{Erfc}(\frac{\nu_x + j}{2\sqrt{\mu_x}}) + 2\frac{\sqrt{D\Delta t}}{\sqrt{\pi}} x_j \mathrm{e}^{-(\nu_x - j)^2/4\mu_x}] \\
&+b_{02}[(W^2(\Delta t)^2 + 2D\Delta t)(\frac{1}{2}\mathrm{Erfc}(\frac{\nu_x - j}{2\sqrt{\mu_x}}) - \frac{1}{2}\mathrm{e}^{\frac{\nu_x j}{\mu_x}} \mathrm{Erfc}(\frac{\nu_x + j}{2\sqrt{\mu_x}}))] \\
&+b_{11}[WV(\Delta t)^2\frac{1}{2}\mathrm{Erfc}(\frac{\nu_x - j}{2\sqrt{\mu_x}}) - W\Delta t(2x_j + V\Delta t)\frac{1}{2}\mathrm{e}^{\frac{\nu_x j}{\mu_x}} \mathrm{Erfc}(\frac{\nu_x + j}{2\sqrt{\mu_x}})] \\
&+b_{12}[(W^2(\Delta t)^2 + 2D\Delta t)[-V\Delta t\frac{1}{2}\mathrm{Erfc}(\frac{\nu_x - j}{2\sqrt{\mu_x}}) \\
&+(2x_j + V\Delta t)\frac{1}{2}\mathrm{e}^{\frac{\nu_x j}{\mu_x}} \mathrm{Erfc}(\frac{\nu_x + j}{2\sqrt{\mu_x}})]] \\
&+b_{21}[-W\Delta t[(V^2(\Delta t)^2 + 2D\Delta t)\frac{1}{2}\mathrm{Erfc}(\frac{\nu_x - j}{2\sqrt{\mu_x}}) \\
&-((2x_j + V\Delta t)^2 + 2D\Delta t)\frac{\mathrm{e}^{\frac{\nu_x j}{\mu_x}}}{2}\mathrm{Erfc}(\frac{\nu_x + j}{2\sqrt{\mu_x}}) + 2\frac{\sqrt{D\Delta t}}{\sqrt{\pi}} x_j \mathrm{e}^{-(\nu_x - j)^2/4\mu_x}]]
\end{aligned}
$$

$$+b_{03}[(-W^3\Delta t^3 - 6WD\Delta t^2)[\frac{1}{2}\mathrm{Erfc}(\frac{\nu_x - j}{2\sqrt{\mu_x}}) - \frac{1}{2}\mathrm{e}^{\frac{\nu_x j}{\mu_x}}\mathrm{Erfc}(\frac{\nu_x + j}{2\sqrt{\mu_x}})]]$$

$$+b_{30}[-(V^3(\Delta t)^3 + 6VD(\Delta t)^2)\frac{1}{2}\mathrm{Erfc}(\frac{\nu_x - j}{2\sqrt{\mu_x}})$$

$$+(2x_j + V\Delta t)((2x_j + V\Delta t)^2 + 6D\Delta t)\frac{1}{2}\mathrm{e}^{\frac{\nu_x j}{\mu_x}}\mathrm{Erfc}(\frac{\nu_x + j}{2\sqrt{\mu_x}})$$

$$-2(2V\Delta t + 3x_j)\frac{\sqrt{D\Delta t}}{\sqrt{\pi}}x_j\mathrm{e}^{-(\nu_x - j)^2/4\mu_x}] + \ldots.$$

Within this general framework we can now obtain finite difference schemes by interpolation on a uniform mesh. We use the usual operators, central, second difference, backward and forward respectively:

$$\begin{aligned}
\Delta_{x0}U_{jk} &= \frac{U_{j+1k} - U_{j-1k}}{2}, \\
\delta_x^2 U_{jk} &= U_{j+1k} - 2U_{jk} + U_{j-1k}, \\
\Delta_{x-}U_{jk} &= U_{jk} - U_{j-1k}, \\
\Delta_{x+}U_{jk} &= U_{j+1k} - U_{jk}.
\end{aligned}$$

The operators $\Delta_{y0}, \delta_y^2, \Delta_{y-}, \Delta_{y+}$ are defined similarly.

Quadratic polynomial interpolation

Assuming that V and W are positive, we choose the six interpolation points to be a five-point star around (x_j, y_k) plus (x_{j-1}, y_{k-1}) then

$$b_{00} = U_{jk}^n, \qquad b_{10} = \frac{\Delta_{x0}U_{jk}^n}{\Delta x}, \qquad b_{01} = \frac{\Delta_{y0}U_{jk}^n}{\Delta y},$$

$$b_{02} = \frac{1}{2}\frac{\delta_x^2 U_{jk}^n}{\Delta x^2}, \quad b_{20} = \frac{1}{2}\frac{\delta_y^2 U_{jk}^n}{\Delta y^2}, \quad b_{11} = \frac{\Delta_{x-}\Delta_{y-}U_{jk}^n}{\Delta x \Delta y}.$$

The approximation is given by

$$\begin{aligned}
U_{jk}^n &= a_x(j)[1 - \nu_x\Delta_{x0} + \nu_x\nu_y\Delta_{x-}\Delta_{y-} + (\frac{1}{2}\nu_x^2 + \mu_x)\delta_x^2 + (\frac{1}{2}\nu_y^2 + \mu_y)\delta_y^2]U_{jk}^n \\
&\quad + b_x(j)[\Delta_{y0} - \nu_y\Delta_{x-}\Delta_{y-}]U_{jk}^n + c_x(j)\delta_x^2 U_{jk}^n,
\end{aligned}$$

where

$$a_x(j) = \frac{1}{2}\mathrm{Erfc}(\frac{\nu_x - j}{2\sqrt{\mu_x}}) - \frac{1}{2}\mathrm{e}^{\frac{\nu_x j}{\mu_x}}\mathrm{Erfc}(\frac{\nu_x + j}{2\sqrt{\mu_x}})$$

$$b_x(j) = j\mathrm{e}^{\frac{\nu_x j}{\mu_x}}\mathrm{Erfc}(\frac{\nu_x + j}{2\sqrt{\mu_x}})$$

$$c_x(j) = -j(j+\nu_x)\mathrm{e}^{\frac{\nu_x j}{\mu_x}}\mathrm{Erfc}(\frac{\nu_x + j}{2\sqrt{\mu_x}}) + Z_x(j)$$

where $Z_x(j) = \frac{\sqrt{\mu_x}}{\sqrt{\pi}} j\mathrm{e}^{-(\nu_x - j)^2/4\mu_x}$.

In a similar way, by choosing ten interpolation points we can deduce a cubic polynomial interpolation. See the computational molecule below.

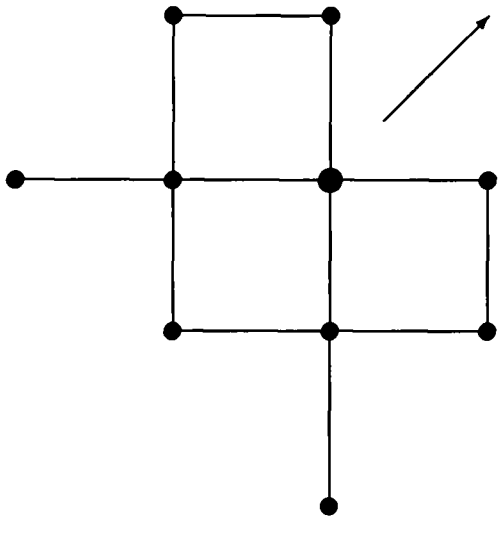


Figure 5.5: Computational molecule for the numerical scheme derived using a cubic polynomial interpolation for V and W positive; The larger circle denotes the central node and predominant flow direction is shown by arrow.

5.4 Concluding remarks

While the new schemes discussed in this chapter are theoretically interesting because of the general framework in which these schemes were derived, they seem not to provide a substantial advantage in terms of accuracy when compared with the Lax-Wendroff scheme and Quickest scheme studied in the previous chapters. We observed some advantages in the stability for the Modified Quickest scheme associated with the suggested numerical boundary conditions and concluded that this gain in stability is associated with the choice of the numerical boundary condition at the first point of the mesh and not necessarily with the Modified Quickest scheme. The problem with the Modified Quickest scheme is that for $j \to \infty$ it becomes identical with the Quickest scheme and consequently for most practical purposes it is only in the first point that we could expect some significant difference. This leads us again to the problem of numerical boundary

conditions. By using the Quickest scheme, rather than the Modified Quickest scheme, but now associated with the numerical boundary conditions which we proposed in this chapter, we have found gains in stability.

Chapter 6

Convection-diffusion in two dimensions

At the heart of computational fluid dynamics, including computation of laminar and turbulent flow and of heat and mass transfer, lies a deceptively simple balance of convection and diffusion. Despite its simplicity, this balance is very difficult to simulate without artificial effects such as increased dispersion or oscillations degrading the solution accuracy. These effects take on increased importance in two and three dimensional flows because of the difficulty in resolving all possible length scales. Great advances in simulating convection-diffusion have occurred in the last two decades and it is now possible to devise schemes of arbitrary accuracy for constant velocity convection in an unbounded domain. We have an understanding of how schemes which previously might be classed finite element or finite difference can be reconciled in a relatively general evolutionary operator framework (see Morton and Sobey [49] and Morton [50]).

In this chapter we deduce a new family of Lax-Wendroff schemes and Quickest schemes by using the evolutionary operator in an unbounded domain for the two-dimensional convection-diffusion problem.

The Lax-Wendroff schemes are a class of schemes which have attained considerable stature in theoretical studies of difference schemes. The essential property of the Lax-Wendroff schemes lies in the combination of time and space-centred discretisations. Their popularity is due to their second-order accuracy and simplicity, although their behaviour around discontinuities is not fully satisfactory.

The Quickest scheme was first generalised to two dimensions by Davis and Moore [13]. When generalising the method they ignored some of the cross-derivatives and that reduced the temporal accuracy of the scheme.

Although the Lax-Wendroff and Quickest schemes are unique for the one dimensional case, they have many variants in the two-dimensional case. Their ambiguity in two dimensions is connected with the fact that different combinations of local nodal values are equally able to model local behaviour. In this chapter, we deduce the analytical solution for a two-dimensional convection-diffusion problem and use it as the source for obtaining Lax-Wendroff and Quickest schemes not yet studied in the literature. The new Quickest schemes are expected to be more accurate in time than the Quickest scheme deduced by Davis and Moore [2], since we take into account the cross-derivatives. Additionally we study in detail the stability of those Lax-Wendroff and Quickest schemes, a crucial property for convergence of numerical schemes.

To analyse the practical stability of the numerical schemes we use von Neumann analysis since we are considering a problem in an unbounded domain. We observe that interesting differences occur between the stability regions of the different numerical schemes. For a clear visualisation of the stability regions we plot the sufficient and necessary stability conditions in three-dimensional space, in which the coordinates involve the convection coefficients and the diffusion.

6.1 Analytic Solution

In the previous chapters we have derived schemes using the exact solution of a one dimensional convection-diffusion problem. Applying the same idea, in this section we deduce the analytic solution for a two-dimensional convection-diffusion problem and use that as the source for obtaining finite difference schemes.

Consider the convection-diffusion equation with coefficient $D > 0$:

$$\frac{\partial u}{\partial t}(x,y,t) + V\frac{\partial u}{\partial x}(x,y,t) + W\frac{\partial u}{\partial y}(x,y,t) = D\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)(x,y,t) \qquad (6.1)$$

and the initial condition

$$u(x,y,0) = u_0(x,y). \qquad (6.2)$$

The diffusion coefficient is taken to be positive since a negative coefficient is a physical impossibility. We take a two-dimensional Fourier transform, denoted $\hat{u}(l,m)$ where l and m are transform variables in the x and y directions. Thus

$$\hat{u}(l,m,t) = u_0(l,m)\mathrm{e}^{-[Dl^2+Dm^2]t+\mathrm{i}[Vl+Wm]t}.$$

Writing this as a convolution integral, we obtain

$$u(x,y,t) = \frac{1}{\pi}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} u_0(x - Vt + 2\sqrt{Dt}\xi, y - Wt + 2\sqrt{Dt}\tau)\mathrm{e}^{-\xi^2-\tau^2}d\xi d\tau. \tag{6.3}$$

This is a two-dimensional evolution operator. In a similar manner, to the one-dimensional case, it defines a Green's function $G(x,y,\Delta t)$ which gives the evolution over a single time-step:

$$u(x,y,t_n+\Delta t) = \frac{1}{\pi}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} u(\xi,\eta,t_n)G(x-\xi,y-\eta,\Delta t)d\xi d\eta \tag{6.4}$$

where

$$G(s,p,t) = \frac{1}{4Dt\pi}\mathrm{e}^{-(s-Vt)^2/4Dt}\mathrm{e}^{-(p-Wt)^2/4Dt}.$$

In the next section we deduce finite difference schemes for an unbounded domain using the evolutionary operator (6.4), following the same procedure as in one dimension.

6.2 Finite differences schemes

The generalisation of finite difference schemes for a convection diffusion equation to multidimensions is not just the sum of the individual one-dimension contributions, since the simple addition of individual finite differences in x and y without appropriate cross terms can lead to instability.

We denote by U_{jk}^n the approximations to the values $u(x_j,y_k,t_n)$ at the mesh points

$$(x_j,y_k) = (j\Delta x, k\Delta y), \quad j,k = 0,\pm1,\pm2,\ldots.$$

We use the central, second difference, backward and forward difference operators defined in the previous chapter. Also in the previous chapter we defined the important quantities ν_x, ν_y, μ_x and μ_y, that we shall be using in what follows.

In the next sections we derive a family of Lax-Wendroff schemes and a family of Quickest schemes in two dimensions. Firstly we derive the schemes based on quadratic and cubic polynomial interpolation of the function $u(\xi,\eta,t_n)$ that appears in (6.4). Secondly we also deduce schemes by using a Taylor approximation of order two and three, of the same function.

6.2.1 Polynomial approximation

In this section we obtain finite difference schemes by approximating $u(x, y, t_n)$ in (6.4) by a local polynomial around the point (x_j, y_k), namely

$$p_{jk}(x, y) = \sum_{r=0}^{K} \sum_{s=0}^{K} b_{rs}(x - x_j)^r (y - y_k)^s.$$

Using the exact evolution operator, the approximation U_{jk}^{n+1} is given by

$$U_{jk}^{n+1} = \sum_{r,s=0}^{K} \frac{b_{rs}}{\pi} \int_{-\infty}^{+\infty} (-V\Delta t + 2\sqrt{D\Delta t}\xi)^r \mathrm{e}^{-\xi^2} d\xi$$

$$\times \int_{-\infty}^{+\infty} (-W\Delta t + 2\sqrt{D\Delta t}\tau)^s \mathrm{e}^{-\tau^2} d\tau.$$

If the power terms are expanded then all the integrals can be determined and then we can write,

$$\begin{aligned} U_{jk}^{n+1} \;=\; & b_{00} - b_{10}V\Delta t - b_{01}W\Delta t + b_{11}VW(\Delta t)^2 \\ & + b_{20}(2D\Delta t + V^2(\Delta t)^2) + b_{02}(2D\Delta t + W^2(\Delta t)^2) \\ & - b_{30}(6DV(\Delta t)^2 + V^3(\Delta t)^3) - b_{03}(6DW(\Delta t)^2 + W^3(\Delta t)^3) \\ & - b_{21}W\Delta t(2D\Delta t + V^2(\Delta t)^2) - b_{12}V\Delta t(2D\Delta t + W^2(\Delta t)^2) \\ & + \ldots. \end{aligned}$$

Through this formula we obtain second and third-order finite difference schemes by using quadratic interpolation or cubic interpolation.

Quadratic polynomial interpolation

If we use quadratic interpolation we need to choose six interpolation points to determine the six coefficients $b_{00}, b_{01}, b_{10}, b_{20}, b_{11}, b_{02}$. We obtain a different method for each choice of points. We choose the points, so that they form a five-point star around (x_j, y_k) and the sixth point is selected according to the direction of the velocity. Assume that V and W are positive. We choose the sixth interpolation point to be (x_{j-1}, y_{k-1}). Then we have

$$b_{00} = U_{jk}^n, \quad b_{10} = \frac{\Delta_{x0} U_{jk}^n}{\Delta x}, \quad b_{01} = \frac{\Delta_{y0} U_{jk}^n}{\Delta y},$$

$$b_{11} = \frac{\Delta_{x-}\Delta_{y-} U_{jk}^n}{\Delta x \Delta y}, \quad b_{02} = \frac{1}{2}\frac{\delta_x^2 U_{jk}^n}{\Delta x^2}, \quad b_{20} = \frac{1}{2}\frac{\delta_y^2 U_{jk}^n}{\Delta y^2}.$$

	$V \geq 0, W \geq 0$	$V \geq 0, W \leq 0$	$V \leq 0, W \geq 0$	$V \leq 0, W \leq 0$
b_{11}	$\Delta_{x-}\Delta_{y-}$	$\Delta_{x-}\Delta_{y+}$	$\Delta_{x+}\Delta_{y-}$	$\Delta_{x+}\Delta_{y+}$

Table 6.1: Polynomial Lax-Wendroff scheme.

This gives the formula,

$$\begin{aligned} U_{jk}^{n+1} &= [1 - (\nu_x \Delta_{x0} + \nu_y \Delta_{y0}) + (\frac{1}{2}\nu_x^2 + \mu_x)\delta_x^2 \\ &\quad +(\frac{1}{2}\nu_y^2 + \mu_y)\delta_y^2 + \nu_x \nu_y \Delta_{x-}\Delta_{y-}]U_{jk}^n. \end{aligned} \tag{6.5}$$

We call this the Polynomial Lax-Wendroff scheme.

Clearly there are three other configurations depending on various combinations of the sign of V and W. For V negative and W positive, we choose the sixth point as (x_{j+1}, y_{k-1}). If V is positive and W negative, we consider the point (x_{j-1}, y_{k+1}) and for V and W negative we choose the point (x_{j+1}, y_{k+1}). The different possibilities can give us a different coefficient b_{11}. We describe in the table 6.1 how the operators that define the coefficient b_{11} change according to the changes of the signs of the velocity components.

We illustrate the different computational molecules in figure 6.1 together with the flow directions.

Cubic polynomial interpolation

Next we turn to cubic interpolation. One advantage of using high-order methods is that numerical diffusion and dispersion errors are relatively smaller than in low order methods.

The procedure is illustrated for two-dimensional flow with positive velocity components V and W as in the previous case. We need to use 10 points to perform this interpolation. Using the 10 points $U_{j-2,k}$, $U_{j-1,k-1}$, $U_{j-1,k}$, $U_{j-1,k+1}$, $U_{j,k-2}$, $U_{j,k-1}$, $U_{j,k}$, $U_{j,k+1}$, $U_{j+1,k-1}$ and $U_{j+1,k}$ to evaluate b_{rs}, $r = 0, 1, 2, 3$; $s =$

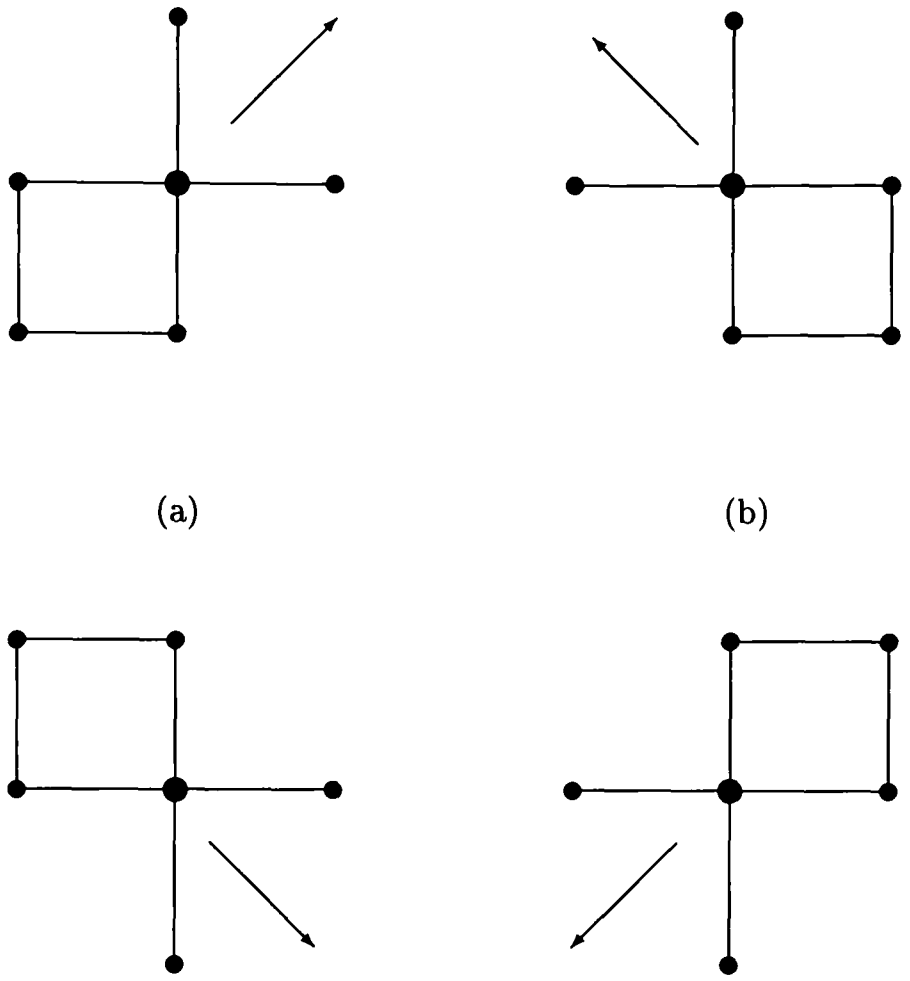


Figure 6.1: Computational molecule for the Polynomial Lax-Wendroff scheme for the different signs of the velocity components: (a) V and W positive; (b) V negative and W positive; (c) V positive and W negative; (d) V and W negative. The larger circle denotes the central node and arrows show typical flow directions.

	b_{11}	b_{12}	b_{21}	b_{30}	b_{03}
$V \geq 0, W \geq 0$	$\Delta_{y+}\Delta_{x-} + \Delta_{x+}\Delta_{y-}$	$\delta_y^2\Delta_{x-}$	$\delta_x^2\Delta_{y-}$	$\delta_y^2\Delta_{x-}$	$\delta_y^2\Delta_{y-}$
$V \leq 0, W \leq 0$	$\Delta_{y+}\Delta_{x-} + \Delta_{x+}\Delta_{y-}$	$\delta_y^2\Delta_{x+}$	$\delta_x^2\Delta_{y+}$	$\delta_y^2\Delta_{x+}$	$\delta_y^2\Delta_{y+}$
$V \geq 0, W \leq 0$	$\Delta_{y-}\Delta_{x-} + \Delta_{x+}\Delta_{y+}$	$\delta_y^2\Delta_{x-}$	$\delta_x^2\Delta_{y+}$	$\delta_y^2\Delta_{x-}$	$\delta_y^2\Delta_{y+}$
$V \leq 0, W \geq 0$	$\Delta_{y-}\Delta_{x-} + \Delta_{x+}\Delta_{y+}$	$\delta_y^2\Delta_{x+}$	$\delta_x^2\Delta_{y-}$	$\delta_y^2\Delta_{x+}$	$\delta_y^2\Delta_{y-}$

Table 6.2: Polynomial Quickest scheme

$0, 1, 2, 3$ we find

$$
\begin{aligned}
b_{00} &= U_{jk}, & b_{10} &= \frac{\Delta_{x0}U_{jk}}{\Delta x} - \frac{\delta_x^2\Delta_{x-}U_{jk}}{6\Delta x},\\
b_{20} &= \frac{\delta_x^2 U_{jk}}{2\Delta x^2}, & b_{30} &= \frac{\delta_x^2\Delta_{x-}U_{jk}}{6\Delta x^3},\\
b_{01} &= \frac{\Delta_{y0}U_{jk}}{\Delta y} - \frac{\delta_y^2\Delta_{y-}U_{jk}}{6\Delta y}, & b_{21} &= \frac{\delta_x^2\Delta_{y-}U_{jk}}{2\Delta x^2\Delta y},\\
b_{02} &= \frac{\delta_y^2 U_{jk}}{2\Delta y^2}, & b_{03} &= \frac{\delta_y^2\Delta_{y-}U_{jk}}{6\Delta y^3},\\
b_{12} &= \frac{\delta_y^2\Delta_{x-}U_{jk}}{2\Delta y^2\Delta x}, & b_{11} &= \frac{\Delta_{y+}\Delta_{x-}U_{jk}}{2\Delta x\Delta y} + \frac{\Delta_{x+}\Delta_{y-}U_{jk}}{2\Delta x\Delta y}.
\end{aligned}
$$

Now we can write the scheme,

$$
\begin{aligned}
U_{jk}^{n+1} &= U_{jk}^n - \nu_x\Delta_{x0}U_{jk}^n - \nu_y\Delta_{y0}U_{jk}^n\\
&\quad +(\frac{1}{2}\nu_x^2 + \mu_x)\delta_x^2U_{jk}^n + (\frac{1}{2}\nu_y^2 + \mu_y)\delta_y^2U_{jk}^n\\
&\quad +\frac{1}{2}\nu_x\nu_y\Delta_{y+}\Delta_{x-}U_{jk}^n + \frac{1}{2}\nu_x\nu_y\Delta_{x+}\Delta_{y-}U_{jk}^n\\
&\quad +\frac{1}{6}\nu_x(1 - \nu_x^2 - 6\mu_x)\delta_x^2\Delta_{x-}U_{jk}^n + \frac{1}{6}\nu_y(1 - \nu_y^2 - 6\mu_y)\delta_y^2\Delta_{y-}U_{jk}^n\\
&\quad -\nu_y(\mu_x + \frac{1}{2}\nu_x^2)\delta_x^2\Delta_{y-}U_{jk}^n - \nu_x(\mu_y + \frac{1}{2}\nu_y^2)\delta_y^2\Delta_{x-}U_{jk}^n. \qquad (6.6)
\end{aligned}
$$

This scheme is called Polynomial Quickest scheme. As with the Lax-Wendroff schemes, we can change the choice of the mesh points, depending on the direction of the velocities. The changes that occur in the scheme (6.6) according to the sign of the velocity components involve changes in the coefficients $b_{11}, b_{12}, b_{21}, b_{30}$ and b_{03} that we describe in table 6.2

We illustrate the different computational molecules in figure 6.2 together with the flow directions.

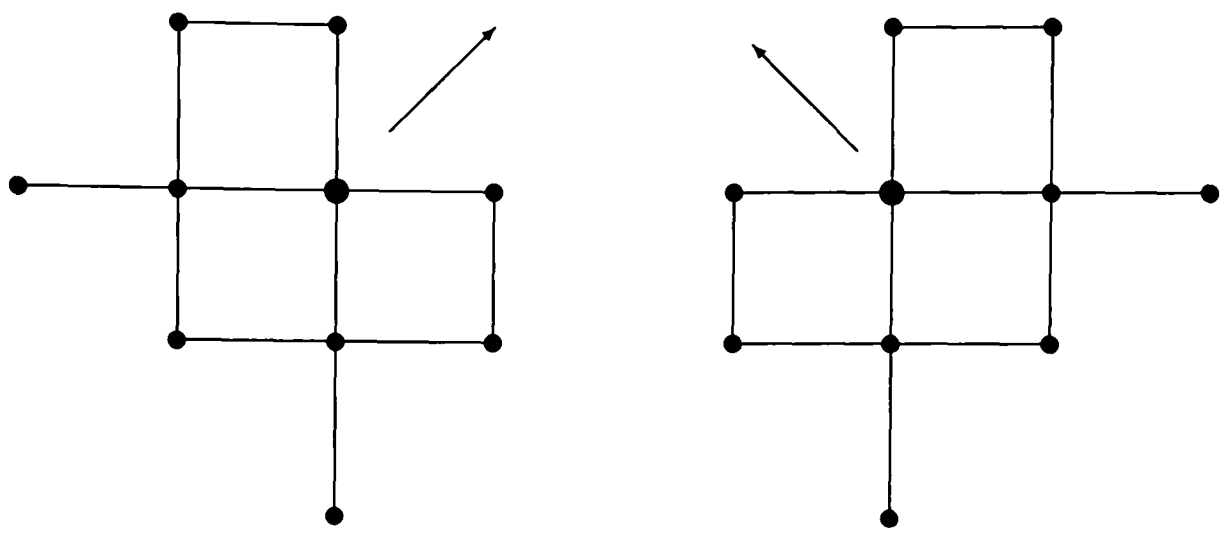


Figure 6.2: Computational molecule for the Polynomial Quickest scheme for the different signs of the velocity components: (a) V and W positive; (b) V negative and W positive; (c) V positive and W negative; (d) V and W negative. The larger circle denotes the central node and predominant flow directions are shown by arrows.

6.2.2 Taylor approximation

In the previous section we considered a local polynomial interpolation based on some selected points in a neighbourhood of (x_j, y_k). Since the local interpolation requires only a small number of neighbouring points, we used the flow directions to choose which neighbouring points to use. Now we use an alternative idea and approximate $u(x, y, t_n)$ by a truncated Taylor series around (x_j, y_k):

$$t_{jk}(x,y) = \sum_{r=0}^{K}\sum_{s=0}^{K} b_{rs}(x-x_j)^r(y-y_k)^s,$$

where $b_{rs} = \frac{u_{x^r y^s}}{r!s!}$. Using the evolutionary operator as in the previous section, depending on the order of the expansion we can obtain the numerical schemes described below.

Second order Taylor expansion

For the second order accurate Taylor expansion the b_{rs} coefficients are:

$$\begin{aligned}
b_{00} &= U_{jk}, & b_{10} &= u_x := \frac{\Delta_{x0}U_{jk}}{\Delta x},\\
b_{01} &= u_y := \frac{\Delta_{y0}U_{jk}}{\Delta y}, & b_{20} &= \frac{1}{2}u_{xx} := \frac{\delta_x^2 U_{jk}}{2\Delta x^2},\\
b_{02} &= \frac{1}{2}u_{yy} := \frac{\delta_y^2 U_{jk}}{2\Delta y^2}, & b_{11} &= u_{xy} := \frac{\Delta_{x0}\Delta_{y0}U_{jk}}{\Delta x \Delta y}.
\end{aligned}$$

This scheme uses a 9 point stencil and has the form:

$$\begin{aligned}
U_{jk}^{n+1} &= [1-(\nu_x\Delta_{x0}+\nu_y\Delta_{y0})+(\frac{1}{2}\nu_x^2+\mu_x)\delta_x^2\\
&\quad +(\frac{1}{2}\nu_y^2+\mu_y)\delta_y^2+\nu_x\nu_y\Delta_{x0}\Delta_{y0}]U_{jk}^n. \qquad (6.7)
\end{aligned}$$

We call this the Taylor Lax-Wendroff scheme. This formula is illustrated by the computational molecule in figure 6.3. This molecule can be used independently of the directions of the velocity components.

Third order Taylor expansion

When we derive different Lax-Wendroff schemes they differ because of the way we discretise the mixed derivatives in the Taylor expansion. The same is true in the case of the two different Quickest schemes. We discretise the mixed

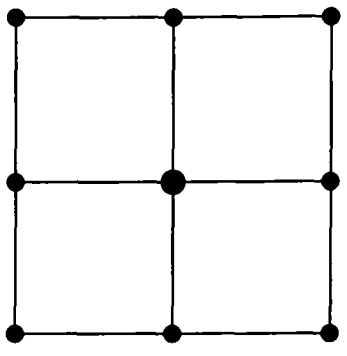


Figure 6.3: Computational molecule for the Taylor Lax-Wendroff scheme. The larger circle denotes the central node.

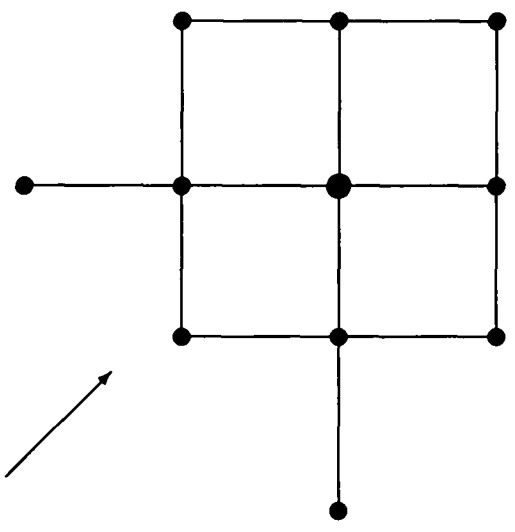


Figure 6.4: Computational molecule for the Taylor Quickest scheme. The larger circle denotes the central node and predominant flow direction is shown by arrow.

	$V \geq 0, W \geq 0$	$V \geq 0, W \leq 0$	$V \leq 0, W \geq 0$	$V \leq 0, W \leq 0$
b_{30}	$\delta_x^2 \Delta_{x-}$	$\delta_x^2 \Delta_{x-}$	$\delta_x^2 \Delta_{x+}$	$\delta_x^2 \Delta_{x+}$
b_{03}	$\delta_y^2 \Delta_{y-}$	$\delta_y^2 \Delta_{y+}$	$\delta_y^2 \Delta_{y-}$	$\delta_y^2 \Delta_{y+}$

Table 6.3: Taylor Quickest scheme.

derivatives u_{xy}, u_{xyy} and u_{yyx} in a different way in both schemes or, to put it differently, we choose in a different way the coefficients b_{11}, b_{12} and b_{21}.

For the third-order accurate Taylor expansion, taking in consideration that V and W are positive, we use the 11 point stencil, $U_{j-2,k}$, $U_{j-1,k-1}$, $U_{j-1,k}$ $U_{j-1,k+1}$, $U_{j,k-2}$, $U_{j,k-1}$, $U_{j,k}$, $U_{j,k+1}$ $U_{j+1,k-1}$, $U_{j+1,k}$, and $U_{j+1,k+1}$. It follows:

$$
\begin{aligned}
b_{00} &= U_{jk}, & b_{10} &= u_x := \frac{\Delta_{x0} U_{jk}}{\Delta x} - \frac{\delta_x^2 \Delta_{x-} U_{jk}}{6\Delta x}, \\
b_{20} &= \frac{1}{2} u_{xx} := \frac{\delta_x^2 U_{jk}}{2\Delta x^2}, & b_{30} &= \frac{1}{6} u_{xxx} := \frac{\delta_x^2 \Delta_{x-} U_{jk}}{6\Delta x^3}, \\
b_{01} &= u_y := \frac{\Delta_{y0} U_{jk}}{\Delta y} - \frac{\delta_y^2 \Delta_{y-} U_{jk}}{6\Delta y}, & b_{21} &= \frac{1}{2} u_{xxy} := \frac{\delta_x^2 \Delta_{y0} U_{jk}}{2\Delta x^2 \Delta y}, \\
b_{02} &= \frac{1}{2} u_{yy} := \frac{\delta_y^2 U_{jk}}{2\Delta y^2}, & b_{03} &= \frac{1}{6} u_{yyy} := \frac{\delta_y^2 \Delta_{y-} U_{jk}}{6\Delta y^3}, \\
b_{12} &= \frac{1}{2} u_{yyx} := \frac{\delta_y^2 \Delta_{x0} U_{jk}}{2\Delta y^2 \Delta x}, & b_{11} &= u_{xy} := \frac{\Delta_{x0} \Delta_{y0} U_{jk}}{\Delta x \Delta y}.
\end{aligned}
$$

We have the numerical method,

$$
\begin{aligned}
U_{jk}^{n+1} \;=\; & U_{jk}^n - \nu_x \Delta_{x0} U_{jk}^n - \nu_y \Delta_{y0} U_{jk}^n \\
& + (\frac{1}{2}\nu_x^2 + \mu_x)\delta_x^2 U_{jk}^n + (\frac{1}{2}\nu_y^2 + \mu_y)\delta_y^2 U_{jk}^n \\
& + \nu_x \nu_y \Delta_{x0} \Delta_{y0} U_{jk}^n \\
& + \frac{1}{6}\nu_x (1 - \nu_x^2 - 6\mu_x)\delta_x^2 \Delta_{x-} U_{jk}^n + \frac{1}{6}\nu_y (1 - \nu_y^2 - 6\mu_y)\delta_y^2 \Delta_{y-} U_{jk}^n \\
& - \nu_y (\mu_x + \frac{1}{2}\nu_x^2)\delta_x^2 \Delta_{y0} U_{jk}^n - \nu_x (\mu_y + \frac{1}{2}\nu_y^2)\delta_y^2 \Delta_{x0} U_{jk}^n. \qquad (6.8)
\end{aligned}
$$

Similarly to the previous schemes, we call this scheme the Taylor Quickest scheme. This formula is illustrated by the computational molecule in figure 6.4. The choice of the mesh points, to approximate the third derivatives, depends on the direction of the velocities and it affects the values of the coefficients b_{30} and b_{03}. The changes are shown in table 6.3.

In the next section we use a von Neumann analysis to determine the stability region of these schemes.

6.3 Von Neumann stability analysis

The von Neumann analysis in two dimensions is a straightforward generalisation of the one-dimensional case. The discrete Fourier decomposition in two dimensions consists of the decomposition of the function into a Fourier series as

$$U_{jk}^n = \sum_{\xi_x,\xi_y} \kappa^n \mathrm{e}^{\mathrm{i}\xi_x j\Delta x} \mathrm{e}^{\mathrm{i}\xi_y k\Delta y},$$

where the range ξ_x, ξ_y is defined separately for each direction, as in the one dimensional case. The amplification factor is still κ as in the one-dimensional case. The products $\xi_x\Delta x$ and $\xi_y\Delta y$ are often represented as phase angles, namely:

$$\theta_x = \xi_x\Delta x, \quad \theta_y = \xi_y\Delta y.$$

To obtain a von Neumann stability condition we insert the Fourier mode

$$\kappa^n \mathrm{e}^{\mathrm{i}j\theta_x} \mathrm{e}^{\mathrm{i}k\theta_y}$$

into the numerical scheme. The amplification factor is said to satisfy the von Neumann condition if there is a constant K such that

$$|\kappa(\theta_x,\theta_y)| \leq 1 + K\Delta t, \quad \forall\, \theta_x,\theta_y \in [0,2\pi]. \tag{6.9}$$

As in the one-dimensional case , in practice we use the stronger condition

$$|\kappa(\theta_x,\theta_y)| \leq 1, \quad \forall\, \theta_x,\theta_y \in [0,2\pi] \tag{6.10}$$

and the discrete scheme that meets this condition, we refer to it as practically stable.

For our finite difference schemes we derive mostly analytical necessary conditions. Nevertheless we plot conditions, determined numerically, that are necessary and sufficient for stability.

First we perform Fourier stability analysis for the convective problem, i.e., for $D = 0$, although we are mainly interested in problems with diffusion. Afterwards we study the fully convective and diffusive discrete scheme.

6.3.1 Practical stability regions for $D = 0$

In this section we consider the convective problem and analyse the different stability regions obtained for the Lax-Wendroff schemes and the Quickest schemes. We first consider the Lax-Wendroff schemes.

Lax-Wendroff schemes

The next result is related with the scheme (6.5) derived using a quadratic polynomial interpolation and called Polynomial Lax-Wendroff scheme. Although we were not able to prove the necessary and sufficient condition plotted in figure 6.5 for the scheme (6.5), we prove analytical necessary conditions.

Lemma 6.1 *Necessary conditions for the practical stability of the Polynomial Lax-Wendroff scheme, when* $D = 0$, $\nu_x \geq 0$ *and* $\nu_y \geq 0$ *are:*

$$(\nu_x - \nu_y) \leq 1, \tag{6.11}$$

$$\nu_x \leq 1, \quad \nu_y \leq 1. \tag{6.12}$$

Proof: The amplification factor for the method (6.5), when $D = 0$ is given by:

$$\begin{aligned}\kappa(\theta_x, \theta_y) &= 1 - \frac{\nu_x}{2}(e^{i\theta_x} - e^{-i\theta_x}) - \frac{\nu_y}{2}(e^{i\theta_y} - e^{-i\theta_y}) \\ &\quad + \frac{\nu_x^2}{2}(e^{i\theta_x} - 2 + e^{-i\theta_x}) + \frac{\nu_y^2}{2}(e^{i\theta_y} - 2 + e^{-i\theta_y}) \\ &\quad + \nu_x\nu_y(1 - e^{-i\theta_x} - e^{-i\theta_y} + e^{-i\theta_x - i\theta_y}).\end{aligned}$$

Then we have,

$$\begin{aligned}\kappa(\theta_x, \theta_y) &= 1 - i\nu_x \sin\theta_x - i\nu_y \sin\theta_y \\ &\quad + \nu_x^2(-1 + \cos\theta_x) + \nu_y^2(-1 + \cos\theta_y) \\ &\quad + \nu_x\nu_y[1 - \cos\theta_x + i\sin\theta_x - \cos\theta_y + i\sin\theta_y \\ &\quad + \cos(\theta_x + \theta_y) - i\sin(\theta_x + \theta_y)].\end{aligned}$$

Consequently,

$$\begin{aligned}|\kappa(\theta_x, \theta_y)|^2 &= [1 - \nu_x^2(1 - \cos\theta_x) - \nu_y^2(1 - \cos\theta_y) \\ &\quad + \nu_x\nu_y((1 - \cos\theta_x)(1 - \cos\theta_y) - \sin\theta_x \sin\theta_y)]^2 \\ &\quad + [-\nu_x \sin\theta_x - \nu_y \sin\theta_y \\ &\quad + \nu_x\nu_y(\sin\theta_x(1 - \cos\theta_y) + \sin\theta_y(1 - \cos\theta_x))]^2\end{aligned}$$

In particular,

$$\begin{aligned}|\kappa(\pi, \pi)| &= |1 - 2\nu_x^2 - 2\nu_y^2 + 4\nu_x\nu_y|, \\ &= |1 - 2(\nu_x - \nu_y)^2|\end{aligned}$$

$$
\begin{aligned}
|\kappa(0,\pi)| &= |1-2\nu_y^2|, \\
|\kappa(\pi,0)| &= |1-2\nu_x^2|.
\end{aligned}
$$

If we have $|\kappa(\pi,\pi)| \leq 1$, $|\kappa(0,\pi)| \leq 1$ and $|\kappa(\pi,0)| \leq 1$ then (6.11) and (6.12) are verified. □

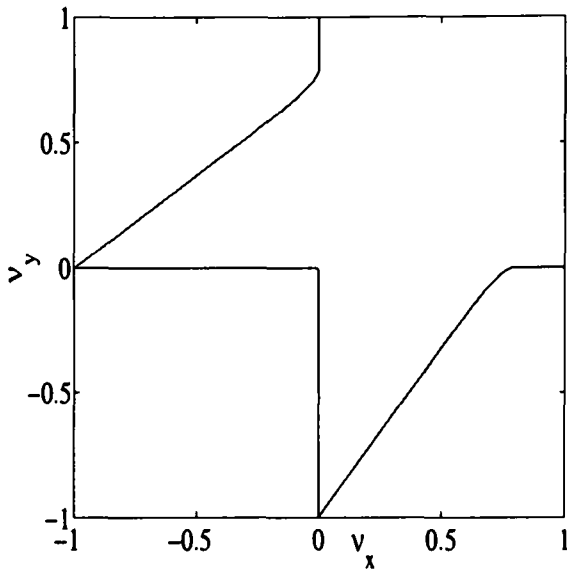


Figure 6.5: Practical stability region for the Polynomial Lax-Wendroff scheme for $\mu_x = \mu_y = 0$ (the scheme was deduced for both velocities positive).

The stability of the scheme (6.7), here called Taylor Lax-Wendroff scheme, was already studied in the literature for $D = 0$. The next result is due to Turkel [86].

Theorem 6.2 *For $D = 0$ the scheme (6.7) is practically stable if and only if*

$$|\nu_x|^{2/3} + |\nu_y|^{2/3} \leq 1. \tag{6.13}$$

Proof: See Turkel [86]. □

The condition (6.13) of the Theorem 6.2 is plotted in figure 6.6.

Quickest schemes

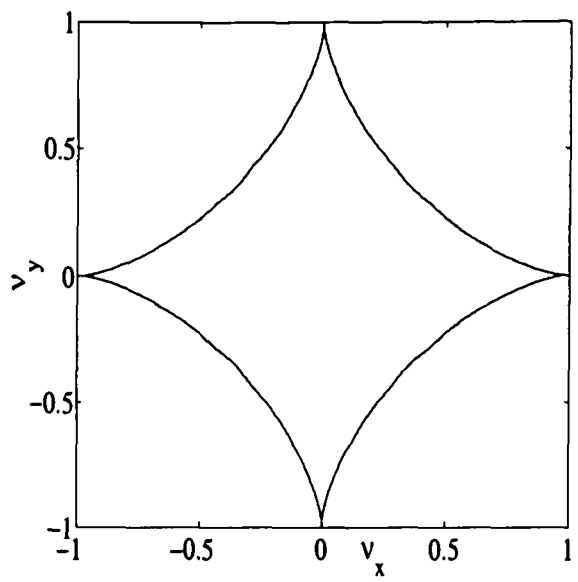


Figure 6.6: Practical stability region for the Taylor Lax-Wendroff scheme for $\mu_x = \mu_y = 0$, deduced independently of the signs of both velocities.

First we consider the Polynomial Quickest scheme and we present a necessary condition for this scheme. This necessary condition is also a sufficient condition although we do not prove it analytically. We plot numerically the sufficient and necessary stability condition for the Polynomial Quickest scheme in figure 6.7.

Lemma 6.3 *If the Polynomial Quickest scheme (6.6) is practically stable, for* $D = 0, \ 0 \leq \nu_x \leq 1$ *and* $0 \leq \nu_y \leq 1$ *then*

$$\nu_x + \nu_y \leq 1. \tag{6.14}$$

Proof: The amplification factor for the Polynomial Quickest scheme is given by:

$$\begin{aligned}
\kappa(\theta_x, \theta_y) \ = \ & 1 - \frac{\nu_x}{2}(e^{i\theta_x} - e^{-i\theta_x}) - \frac{\nu_y}{2}(e^{i\theta_y} - e^{-i\theta_y}) \\
& + \frac{\nu_x^2}{2}(e^{i\theta_x} - 2 + e^{-i\theta_x}) + \frac{\nu_y^2}{2}(e^{i\theta_y} - 2 + e^{-i\theta_y}) \\
& + \frac{\nu_x\nu_y}{2}(e^{i\theta_y} - 1)(1 - e^{-i\theta_x}) + \frac{\nu_x\nu_y}{2}(e^{i\theta_x} - 1)(1 - e^{-i\theta_y}) \\
& + \frac{1}{6}\nu_x(1 - \nu_x^2)(e^{i\theta_x} - 2 + e^{-i\theta_x})(1 - e^{-i\theta_x}) \\
& + \frac{1}{6}\nu_y(1 - \nu_y^2)(e^{i\theta_y} - 2 + e^{-i\theta_y})(1 - e^{-i\theta_y})
\end{aligned}$$

$$-\frac{1}{2}\nu_y\nu_x^2(e^{i\theta_x}-2+e^{-i\theta_x})(1-e^{-i\theta_y})$$
$$-\frac{1}{2}\nu_x\nu_y^2(e^{i\theta_y}-2+e^{-i\theta_y})(1-e^{-i\theta_x}).$$

We obtain,

$$\begin{aligned}\kappa(\theta_x,\theta_y) &= 1-i\nu_x\sin\theta_x-i\nu_y\sin\theta_y+\nu_x^2(-1+\cos\theta_x)+\nu_y^2(-1+\cos\theta_y)\\ &\quad -\nu_x\nu_y((1-\cos\theta_x)(1-\cos\theta_y)+\sin\theta_x\sin\theta_y)\\ &\quad +\frac{1}{6}\nu_x(1-\nu_x^2)[-2(1-\cos\theta_x)^2+2i\sin\theta_x(-1+\cos\theta_x)]\\ &\quad +\frac{1}{6}\nu_y(1-\nu_y^2)[-2(1-\cos\theta_y)^2+2i\sin\theta_y(-1+\cos\theta_y)]\\ &\quad -\frac{1}{2}\nu_y\nu_x^2[-2(1-\cos\theta_y)(1-\cos\theta_x)+2i\sin\theta_y(-1+\cos\theta_x)]\\ &\quad -\frac{1}{2}\nu_x\nu_y^2[-2(1-\cos\theta_x)(1-\cos\theta_y)+2i\sin\theta_x(-1+\cos\theta_y)].\end{aligned}$$

In particular we have,

$$\begin{aligned}\kappa(\pi,\pi) &= 1-2\nu_x^2-2\nu_y^2-4\nu_x\nu_y-\frac{4}{3}\nu_x(1-\nu_x^2)\\ &\quad -\frac{4}{3}\nu_y(1-\nu_y^2)+4\nu_x\nu_y^2+4\nu_y\nu_x^2\\ &= 1-2(\nu_x+\nu_y)^2-\frac{4}{3}(\nu_x+\nu_y)+\frac{4}{3}(\nu_x+\nu_y)^3.\end{aligned}$$

To have $|\kappa(\pi,\pi)|\le 1$ implies $\nu_x+\nu_y\le 1$ or $\nu_x+\nu_y\ge 3/2$. From this fact we obtain (6.14). □

The next lemma is a necessary condition for the Taylor Quickest scheme. We plot the necessary and sufficient conditions for this scheme in figure 6.8.

Lemma 6.4 *If the Taylor Quickest scheme (6.8) is practically stable, for* $D=0$, $0\le\nu_x\le 1$ *and* $0\le\nu_y\le 1$ *then*

$$\nu_x+\nu_y\le 1. \tag{6.15}$$

Proof: The amplification factor for the Taylor Quickest scheme is

$$\begin{aligned}\kappa(\theta_x,\theta_y) &= 1-\frac{\nu_x}{2}(e^{i\theta_x}-e^{-i\theta_x})-\frac{\nu_y}{2}(e^{i\theta_y}-e^{-i\theta_y})\\ &\quad +\frac{\nu_x^2}{2}(e^{i\theta_x}-2+e^{-i\theta_x})+\frac{\nu_y^2}{2}(e^{i\theta_y}-2+e^{-i\theta_y})\end{aligned}$$

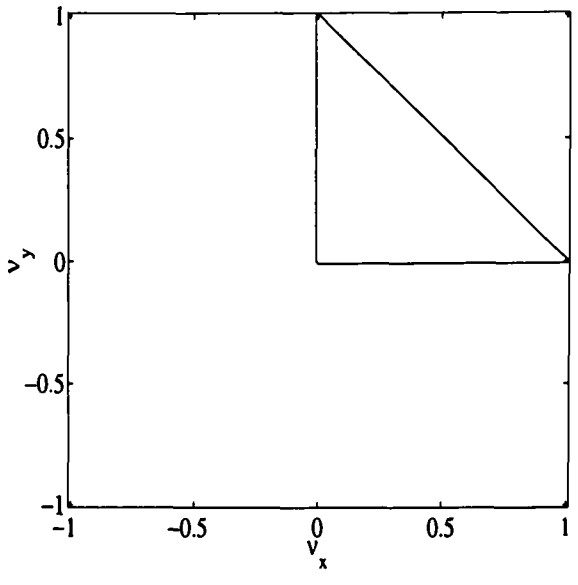


Figure 6.7: Practical stability region for the Polynomial Quickest scheme for $\mu_x = \mu_y = 0$, deduced when both velocities are considered positive and consequently ν_x and ν_y are positive.

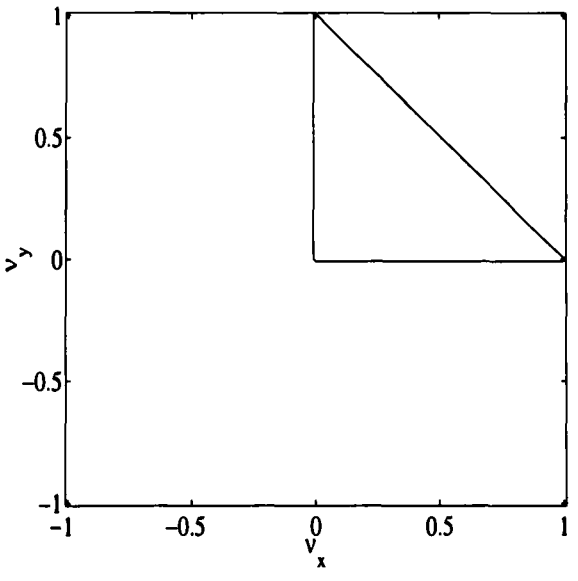


Figure 6.8: Practical stability region for the Taylor Quickest scheme for $\mu_x = \mu_y = 0$, deduced when both velocities are considered positive and consequently ν_x and ν_y are positive.

$$
\begin{aligned}
&+\frac{\nu_x\nu_y}{4}(e^{i\theta_x}-e^{-i\theta_x})(e^{i\theta_y}-e^{-i\theta_y})\\
&+\frac{1}{6}\nu_x(1-\nu_x^2)(e^{i\theta_x}-2+e^{-i\theta_x})(1-e^{-i\theta_x})\\
&+\frac{1}{6}\nu_y(1-\nu_y^2)(e^{i\theta_y}-2+e^{-i\theta_y})(1-e^{-i\theta_y})\\
&-\frac{\nu_y}{4}\nu_x^2(e^{i\theta_x}-2+e^{-i\theta_x})(e^{i\theta_y}-e^{-i\theta_y})\\
&-\frac{\nu_x}{4}\nu_y^2(e^{i\theta_y}-2+e^{-i\theta_y})(e^{i\theta_x}-e^{-i\theta_x}).
\end{aligned}
$$

Then we have,

$$
\begin{aligned}
\kappa(\theta_x,\theta_y) \;=\;& 1-i\nu_x\sin\theta_x-i\nu_y\sin\theta_y\\
&+\nu_x^2(-1+\cos\theta_x)+\nu_y^2(-1+\cos\theta_y)\\
&-\nu_x\nu_y\sin\theta_x\sin\theta_y\\
&-\frac{1}{3}\nu_x(1-\nu_x^2)[(1-\cos\theta_x)^2+i\sin\theta_x(1-\cos\theta_x)]\\
&-\frac{1}{3}\nu_y(1-\nu_y^2)[(1-\cos\theta_y)^2+i\sin\theta_y(1-\cos\theta_y)]\\
&+\nu_y\nu_x^2 i\sin\theta_y(1-\cos\theta_x)+\nu_x\nu_y^2 i\sin\theta_x(1-\cos\theta_y).
\end{aligned}
$$

As in the previous cases we consider in particular,

$$
\begin{aligned}
\kappa(\pi,\pi) \;=\;& 1-2\nu_x^2-2\nu_y^2-\frac{4}{3}\nu_x(1-\nu_x^2)-\frac{4}{3}\nu_y(1-\nu_y^2)\\
=\;& 1-2[(\nu_x+\nu_y)^2+\frac{2}{3}(\nu_x+\nu_y)-\frac{2}{3}(\nu_x+\nu_y)^3\\
&-2\nu_x\nu_y(1-\nu_x-\nu_y)]
\end{aligned}
$$

and if $|\kappa(\pi,\pi)|\leq 1$ then we have (6.15) . □

In this section we assumed $D=0$. In the next section we analyse the stability regions for the schemes considered in this section but with the diffusive coefficient $D>0$.

6.3.2 Practical stability regions for $D>0$

In this section we consider the parabolic problem that is simultaneously convective and diffusive. We analyse the stability analytically, obtaining mainly necessary conditions. We plot numerically the sufficient and necessary stability regions in the three dimensional space (ν_x,ν_y,μ), where we assume $\mu=\mu_x=\mu_y$.

A finite difference scheme for the two-dimensional convection diffusion equation (6.1) which is quite well known in the literature is the central scheme:

$$U_{jk}^{n+1} = U_{jk}^n - \nu_x \Delta_{x0} U_{jk}^n - \nu_y \Delta_{y0} U_{jk}^n + \mu_x \delta_x^2 U_{jk}^n + \mu_y \delta_y^2 U_{jk}^n. \tag{6.16}$$

The von Neumann stability analysis of this scheme was studied by Hindmarsh and Gresho [29] for a multidimensional problem.

We show the practical stability region of the central scheme in order to compare it with the Lax-Wendroff schemes and the Quickest schemes that we are studying. The von Neumann necessary and sufficient conditions for stability, for the central scheme (6.16), are given in the next theorem due to Hindmarsh and Gresho [29]. They are also plotted in figure 6.9, for $\mu = \mu_x = \mu_y$.

Theorem 6.5 *The scheme (6.16) is practically stable if and only if*

$$2\mu_x + 2\mu_y \leq 1, \tag{6.17}$$

$$\frac{\nu_x^2}{2\mu_x} + \frac{\nu_y^2}{2\mu_y} \leq 1. \tag{6.18}$$

Proof: See Hindmarsh and Gresho [29]. □

In figure 6.9b and figure 6.9c we show the projections of figure 6.9a in two different planes to give us a more accurate idea of the stability region.

For the next finite difference schemes we derive mostly analytical necessary conditions. Nevertheless we plot sufficient and necessary conditions for stability determined numerically.

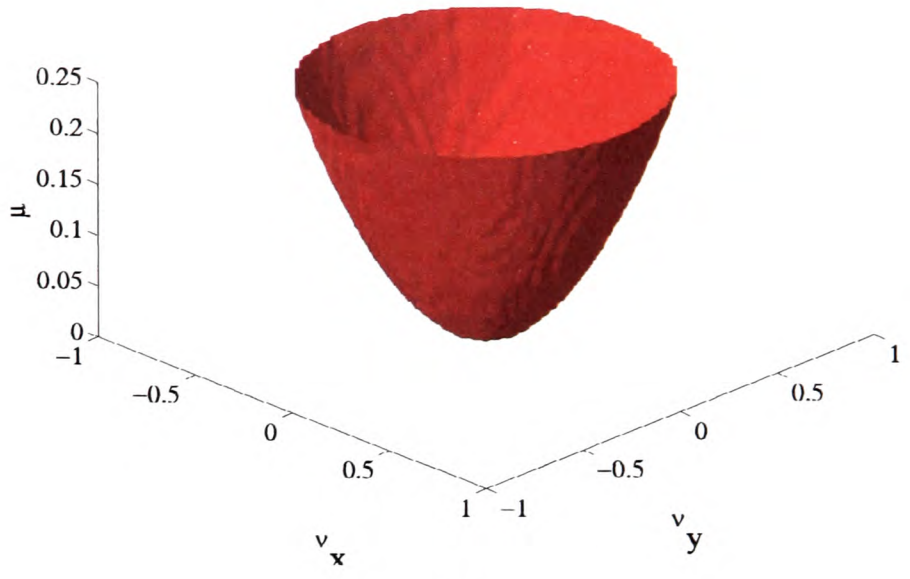
Lax-Wendroff schemes

We start to analyse the Polynomial Lax-Wendroff scheme by giving some necessary conditions for the stability of this scheme.

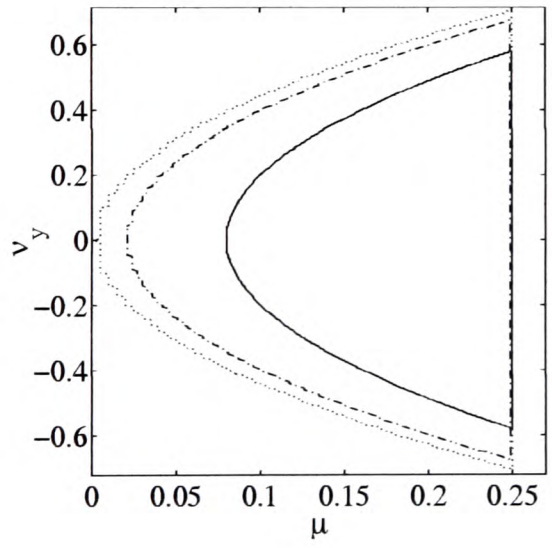
Lemma 6.6 *Necessary conditions for the Polynomial Lax-Wendroff scheme (6.5) to be practically stable are*

$$2(\mu_x + \mu_y) \leq 1 \tag{6.19}$$

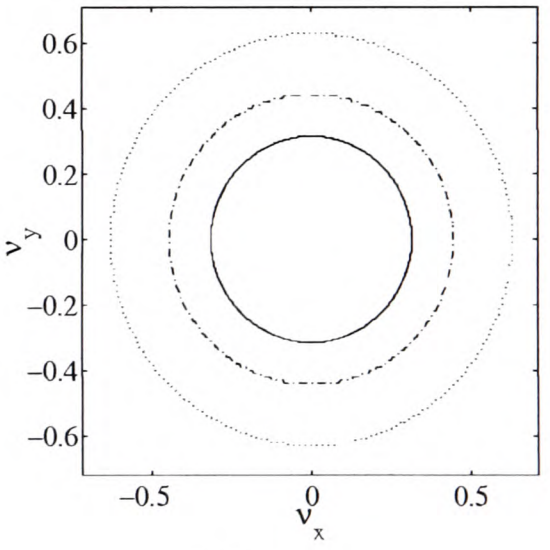
$$(\nu_x - \nu_y)^2 \leq 1 - 2(\mu_x + \mu_y). \tag{6.20}$$



(a)



(b)



(c)

Figure 6.9: (a) Practical stability analysis for the central scheme; (b) projection of the figure (a) on the plane $\mu \circ \nu_y$: $\nu_x = 0$ $(\cdots)$; $\nu_x = 0.2$ $(- \cdot -)$; $\nu_x = 0.4$ $(-)$; (c) projection of the figure (a) on the plane $\nu_x \circ \nu_y$: $\mu = 0.05$ $(-)$; $\mu = 0.1$ $(- \cdot -)$; $\mu = 0.2$ $(\cdots)$.

Proof: The amplification factor for the Polynomial Lax-Wendroff scheme is given by:

$$\begin{aligned}\kappa(\theta_x,\theta_y) &= 1-\frac{\nu_x}{2}(e^{i\theta_x}-e^{-i\theta_x})-\frac{\nu_y}{2}(e^{i\theta_y}-e^{-i\theta_y})\\ &\quad+(\frac{\nu_x^2}{2}+\mu_x)(e^{i\theta_x}-2+e^{-i\theta_x})+(\frac{\nu_y^2}{2}+\mu_y)(e^{i\theta_y}-2+e^{-i\theta_y})\\ &\quad+\nu_x\nu_y(1-e^{-i\theta_x}-e^{-i\theta_y}+e^{-i\theta_x-i\theta_y}).\end{aligned}$$

The amplification factor can be written in the form,

$$\begin{aligned}\kappa(\theta_x,\theta_y) &= 1-i\nu_x\sin\theta_x-i\nu_y\sin\theta_y\\ &\quad+(\nu_x^2+2\mu_x)(-1+\cos\theta_x)+(\nu_y^2+2\mu_y)(-1+\cos\theta_y)\\ &\quad+\nu_x\nu_y[1-\cos\theta_x+i\sin\theta_x-\cos\theta_y+i\sin\theta_y\\ &\quad+\cos(\theta_x+\theta_y)-i\sin(\theta_x+\theta_y)].\end{aligned}$$

Consequently the square of the modulus of the amplification factor is,

$$\begin{aligned}|\kappa(\theta_x,\theta_y)|^2 &= [1-(\nu_x^2+2\mu_x)(1-\cos\theta_x)-(\nu_y^2+2\mu_y)(1-\cos\theta_y)\\ &\quad+\nu_x\nu_y((1-\cos\theta_x)(1-\cos\theta_y)-\sin\theta_x\sin\theta_y)]^2\\ &\quad+[-\nu_x\sin\theta_x-\nu_y\sin\theta_y\\ &\quad+\nu_x\nu_y(\sin\theta_x(1-\cos\theta_y)+\sin\theta_y(1-\cos\theta_x))]^2.\end{aligned}$$

For the limiting case $\theta_x\to 0$ and $\theta_y\to 0$ with $|\theta_x|\le\theta$ and $|\theta_y|\le\theta$, we can write

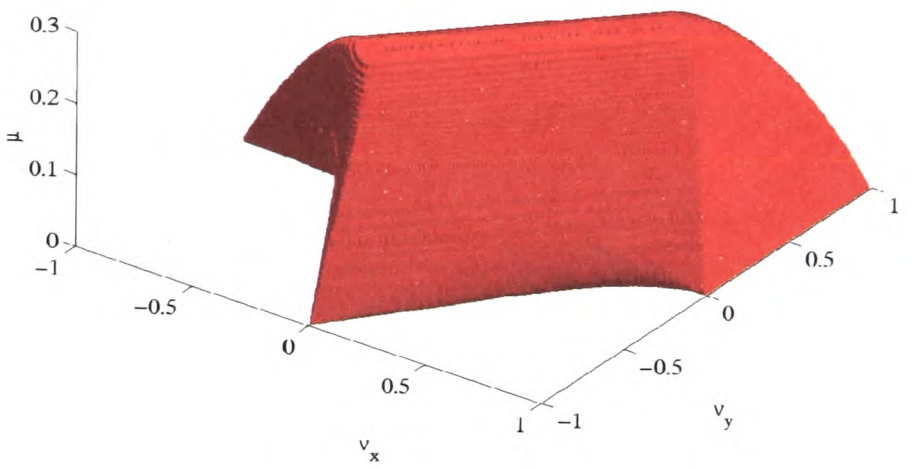
$$\begin{aligned}|\kappa(\theta_x,\theta_y)|^2 &= [1-(\nu_x^2+2\mu_x)\frac{\theta_x^2}{2}-(\nu_y^2+2\mu_y)\frac{\theta_y^2}{2}\\ &\quad+\nu_x\nu_y(\frac{\theta_x^2}{2}\frac{\theta_y^2}{2}-\theta_x\theta_y)+O(\theta^4)]^2\\ &\quad+[-\nu_x\theta_x-\nu_y\theta_y+\nu_x\nu_y(\theta_x\frac{\theta_y^2}{2}+\theta_y\frac{\theta_x^2}{2})+O(\theta^3)]^2.\end{aligned}$$

After some algebraic calculations we obtain,

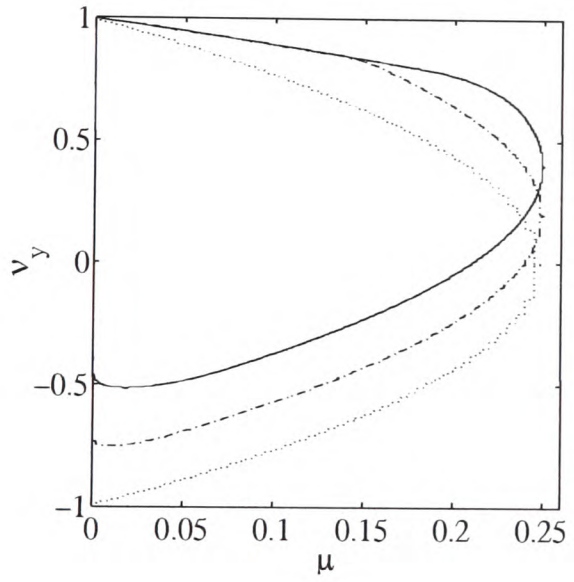
$$\begin{aligned}|\kappa(\theta_x,\theta_y)|^2 &= 1-(\nu_x^2+2\mu_x)\theta_x^2-(\nu_y^2+2\mu_y)\theta_y^2-2\nu_x\nu_y\theta_x\theta_y\\ &\quad+(\nu_x\theta_x+\nu_y\theta_y)^2+O(\theta^4)\\ &= 1-(2\mu_x+2\mu_y)+O(\theta^4).\end{aligned}$$

In order to have $|\kappa(\theta_x,\theta_y)|\le 1$ for all θ_x, θ_y we need to have (6.19). For the particular case $\theta_x=\theta_y=\pi$ we have,

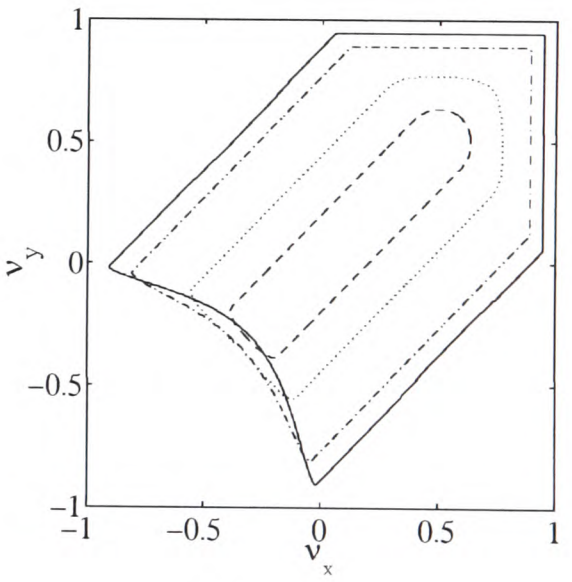
$$\kappa(\pi,\pi)=1-2(\nu_x^2+2\mu_x)-2(\nu_y^2+2\mu_y)+4\nu_x\nu_y.$$



(a)



(b)



(c)

Figure 6.10: (a) Practical stability analysis for the Polynomial Lax-Wendroff scheme; (b) projection of the figure (a) on the plane $\mu \circ \nu_y$: $\nu_x = 0$ $(\cdots)$; $\nu_x = 0.2$ $(- \cdot -)$; $\nu_x = 0.4$ $(-)$; (c) projection of the figure (a) on the plane $\nu_x \circ \nu_y$: $\mu = 0.05$ $(-)$; $\mu = 0.1$ $(- \cdot -)$; $\mu = 0.2$ $(\cdots)$; $\mu = 0.24$ $(--)$.

Then to have $|\kappa(\pi,\pi)| \leq 1$, we need to have (6.20). □

The condition (6.20) is associated with the diagonal lines shown in figure 6.10c, determining that the stability region is between the two lines as we see in the figure 6.10c. The condition (6.19) gives us a limit for the diffusion parameter. We observe that this scheme is still stable for simultaneous big values of ν_x and ν_y, when μ is small. Although we plot the stability region for $(\nu_x, \nu_y) \in [-1,1] \times [-1,1]$, note that for this scheme we are assuming that both velocity components are positive, this is that, ν_x and ν_y are both positive.

The next result is for the Taylor Lax-Wendroff scheme. This scheme was deduced independently of the signs of the velocities.

Lemma 6.7 *Necessary conditions for the Taylor Lax-Wendroff scheme (6.7) to be practically stable are*

$$2(\mu_x + \mu_y) \leq 1 \tag{6.21}$$

$$\nu_x^2 + \nu_y^2 \leq 1 - 2(\mu_x + \mu_y). \tag{6.22}$$

Proof: The amplification factor for the Taylor Lax-Wendroff scheme is given by:

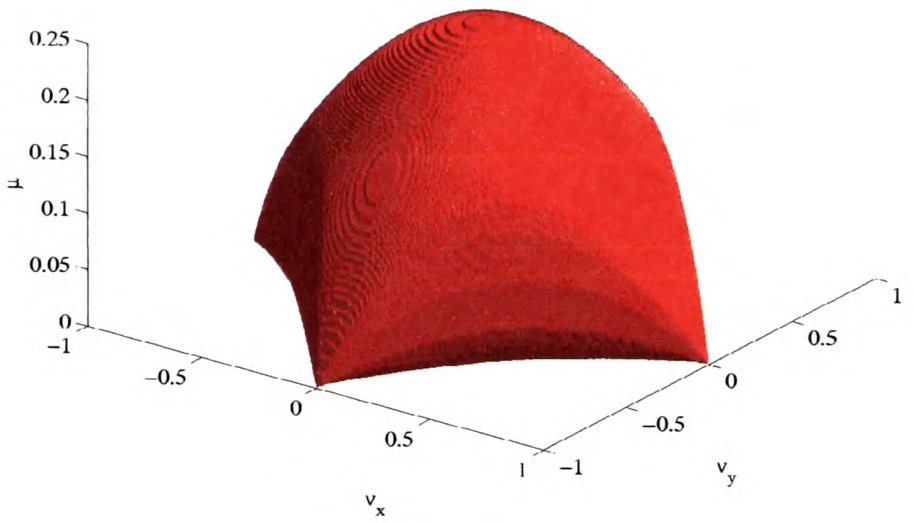
$$\begin{aligned}\kappa(\theta_x,\theta_y) &= 1 - \frac{\nu_x}{2}(e^{i\theta_x} - e^{-i\theta_x}) - \frac{\nu_y}{2}(e^{i\theta_y} - e^{-i\theta_y}) \\ &\quad +(\frac{\nu_x^2}{2} + \mu_x)(e^{i\theta_x} - 2 + e^{-i\theta_x}) + (\frac{\nu_y^2}{2} + \mu_y)(e^{i\theta_y} - 2 + e^{-i\theta_y}) \\ &\quad +\frac{\nu_x\nu_y}{4}(e^{i\theta_x+i\theta_y} - e^{-i\theta_x+i\theta_y} - e^{i\theta_x-i\theta_y} + e^{-i\theta_x-i\theta_y}).\end{aligned}$$

This can be written as,

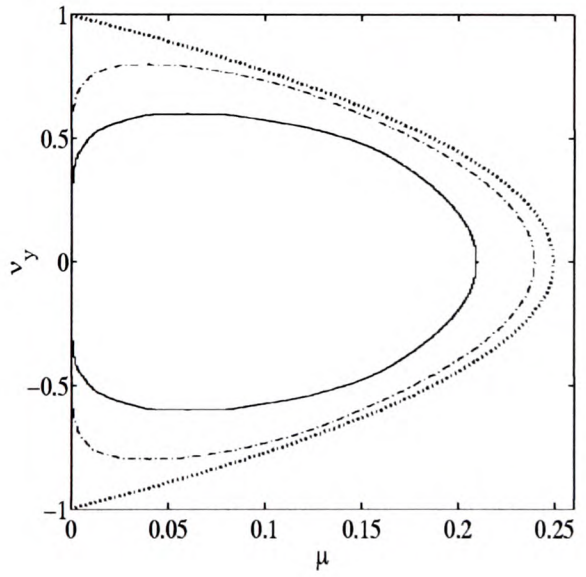
$$\begin{aligned}\kappa(\theta_x,\theta_y) &= 1 - i\nu_x \sin\theta_x - i\nu_y \sin\theta_y \\ &\quad +(\nu_x^2 + 2\mu_x)(-1 + \cos\theta_x) + (\nu_y^2 + 2\mu_y)(-1 + \cos\theta_y) \\ &\quad -\nu_x\nu_y \sin\theta_x \sin\theta_y\end{aligned}$$

Consequently, we can write,

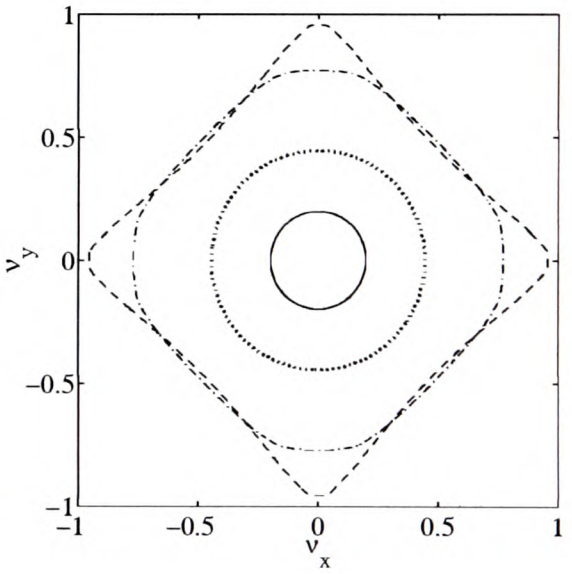
$$\begin{aligned}|\kappa(\theta_x,\theta_y)|^2 &= [1 - (\nu_x^2 + 2\mu_x)(1 - \cos\theta_x) - (\nu_y^2 + 2\mu_y)(1 - \cos\theta_y) \\ &\quad -\nu_x\nu_y \sin\theta_x \sin\theta_y]^2 \\ &\quad +[\nu_x \sin\theta_x + \nu_y \sin\theta_y]^2.\end{aligned}$$



(a)



(b)



(c)

Figure 6.11: (a) Practical stability analysis for the Taylor Lax-Wendroff scheme; (b) projection of the figure (a) on the plane $\mu \circ \nu_y$: $\nu_x = 0$ $(\cdots)$; $\nu_x = 0.2$ $(- \cdot -)$; $\nu_x = 0.4$ (- -); (c) projection of the figure (a) on the plane $\nu_x \circ \nu_y$: $\mu = 0.02$ $(--)$; $\mu = 0.1$ $(- \cdot -)$; $\mu = 0.2$ $(\cdots)$; $\mu = 0.24$ $(-)$.

For the limiting case $\theta_x \to 0$ and $\theta_y \to 0$ with $|\theta_x| \leq \theta$ and $|\theta_y| \leq \theta$, we can write,

$$\begin{aligned} |\kappa(\theta_x,\theta_y)|^2 &= [1-(\nu_x^2+2\mu_x)\frac{\theta_x^2}{2}-(\nu_y^2+2\mu_y)\frac{\theta_y^2}{2}-\nu_x\nu_y\theta_x\theta_y+O(\theta^4)]^2 \\ &\quad +[(\nu_x\theta_x+\nu_y\theta_y)+O(\theta^3)]^2. \end{aligned}$$

After some calculations,

$$\begin{aligned} |\kappa(\theta_x,\theta_y)|^2 &= 1-(\nu_x^2+2\mu_x)\theta_x^2-(\nu_y^2+2\mu_y)\theta_y^2-2\nu_x\nu_y\theta_x\theta_y \\ &\quad +(\nu_x\theta_x+\nu_y\theta_y)^2+O(\theta^4) \\ &= 1-(2\mu_x+2\mu_y)+O(\theta^4). \end{aligned}$$

In order to have $|\kappa(\theta_x,\theta_y)| \leq 1$ for all θ_x, θ_y the condition (6.21) needs to hold. For the particular case $\theta_x = \theta_y = \pi$ we have,

$$|\kappa(\pi,\pi)| = |1-2(\nu_x^2+2\mu_x)-2(\nu_y^2+2\mu_y)|$$

and $|\kappa(\pi,\pi)| \leq 1$ is equivalent to (6.22). □

This scheme has a smaller region of stability for small μ compared with the Polynomial Lax-Wendroff scheme (6.5). However this scheme has the advantage that it can be used independently of the signs of the velocities. In figure 6.11 we plot the necessary and sufficient practical stability conditions for the Taylor Lax-Wendroff scheme (6.7).

Observing the figure 6.11c, the necessary and sufficient condition for this scheme is the condition (6.22) associated with some other condition that seems to change as μ increases from $|\nu_x|^{2/3}+|\nu_y|^{2/3} \leq 1$ to $|\nu_x|+|\nu_y| \leq 1$.

Quickest schemes

In the one-dimensional case we saw that for the Quickest scheme to find analytical sufficient and necessary conditions involves cumbersome expressions that make it difficult to have a clear understanding of the region. In the two dimensional case we should expect to find even more complexity.

We consider first the Polynomial Quickest scheme. The next lemma is about a necessary condition for the stability of this scheme. We plot the sufficient and necessary conditions in figure 6.12.

Lemma 6.8 *A necessary condition for the Polynomial Quickest scheme (6.6) to be practically stable is*

$$\begin{aligned}(\nu_x^2+2\mu_x)(1-2\nu_y)+(\nu_y^2+2\mu_y)(1-2\nu_x)+2\nu_x\nu_y \\ +\frac{2}{3}\nu_x(1-\nu_x^2-6\mu_x)+\frac{2}{3}\nu_y(1-\nu_y^2-6\mu_y) \;\leq\; 1\end{aligned} \tag{6.23}$$

Proof: The amplification factor for the method is given by:

$$\begin{aligned}\kappa(\theta_x,\theta_y) \;=\; & 1-\frac{\nu_x}{2}(e^{i\theta_x}-e^{-i\theta_x})-\frac{\nu_y}{2}(e^{i\theta_y}-e^{-i\theta_y}) \\ & +(\frac{\nu_x^2}{2}+\mu_x)(e^{i\theta_x}-2+e^{-i\theta_x})+(\frac{\nu_y^2}{2}+\mu_y)(e^{i\theta_y}-2+e^{-i\theta_y}) \\ & +\frac{\nu_x\nu_y}{2}(e^{i\theta_y}-1)(1-e^{-i\theta_x})+\frac{\nu_x\nu_y}{2}(e^{i\theta_x}-1)(1-e^{-i\theta_y}) \\ & +\frac{1}{6}\nu_x(1-\nu_x^2-6\mu_x)(e^{i\theta_x}-2+e^{-i\theta_x})(1-e^{-i\theta_x}) \\ & +\frac{1}{6}\nu_y(1-\nu_y^2-6\mu_y)(e^{i\theta_y}-2+e^{-i\theta_y})(1-e^{-i\theta_y}) \\ & -\nu_y(\mu_x+\frac{1}{2}\nu_x^2)(e^{i\theta_x}-2+e^{-i\theta_x})(1-e^{-i\theta_y}) \\ & -\nu_x(\mu_y+\frac{1}{2}\nu_y^2)(e^{i\theta_y}-2+e^{-i\theta_y})(1-e^{-i\theta_x}).\end{aligned}$$

We can write the amplification factor in the form,

$$\begin{aligned}\kappa(\theta_x,\theta_y) \;=\; & 1-i\nu_x\sin\theta_x-i\nu_y\sin\theta_y \\ & +(\nu_x^2+2\mu_x)(-1+\cos\theta_x)+(\nu_y^2+2\mu_y)(-1+\cos\theta_y) \\ & -\nu_x\nu_y((1-\cos\theta_x)(1-\cos\theta_y)+\sin\theta_x\sin\theta_y) \\ & +\frac{1}{6}\nu_x(1-\nu_x^2-6\mu_x)[-2(1-\cos\theta_x)^2+2i\sin\theta_x(-1+\cos\theta_x)] \\ & +\frac{1}{6}\nu_y(1-\nu_y^2-6\mu_y)[-2(1-\cos\theta_y)^2+2i\sin\theta_y(-1+\cos\theta_y)] \\ & -\nu_y(\mu_x+\frac{1}{2}\nu_x^2)[-2(1-\cos\theta_y)(1-\cos\theta_x) \\ & +2i\sin\theta_y(-1+\cos\theta_x)] \\ & -\nu_x(\mu_y+\frac{1}{2}\nu_y^2)[-2(1-\cos\theta_x)(1-\cos\theta_y) \\ & +2i\sin\theta_x(-1+\cos\theta_y)].\end{aligned}$$

For the particular case $\theta_x=\theta_y=\pi$ we have

$$\begin{aligned}\kappa(\pi,\pi) \;=\; & 1-2(\nu_x^2+2\mu_x)-2(\nu_y^2+2\mu_y)-4\nu_x\nu_y \\ & -\frac{4}{3}\nu_x(1-\nu_x^2-6\mu_x)-\frac{4}{3}\nu_y(1-\nu_y^2-6\mu_y) \\ & +4\nu_x(2\mu_y+\nu_y^2)+4\nu_y(2\mu_x+\nu_x^2)\end{aligned}$$

and to have $|\kappa(\pi,\pi)| \leq 1$ we need to have (6.23) . □

We can tell from the complexity of the expression for the amplification factor in the previous lemma, that to derive sufficient conditions here is, if not an almost impossible task, then a very hard one.

We notice that on the three dimensional surface displayed in figure 6.12, we have $\mu \leq 9/16$. This value corresponds to $(\nu_x, \nu_y) = (1/4, 1/4)$.

Next we provide a necessary condition for the stability of the Taylor Quickest scheme.

Lemma 6.9 *A necessary condition for the Taylor Quickest scheme (6.8) to be practically stable is*

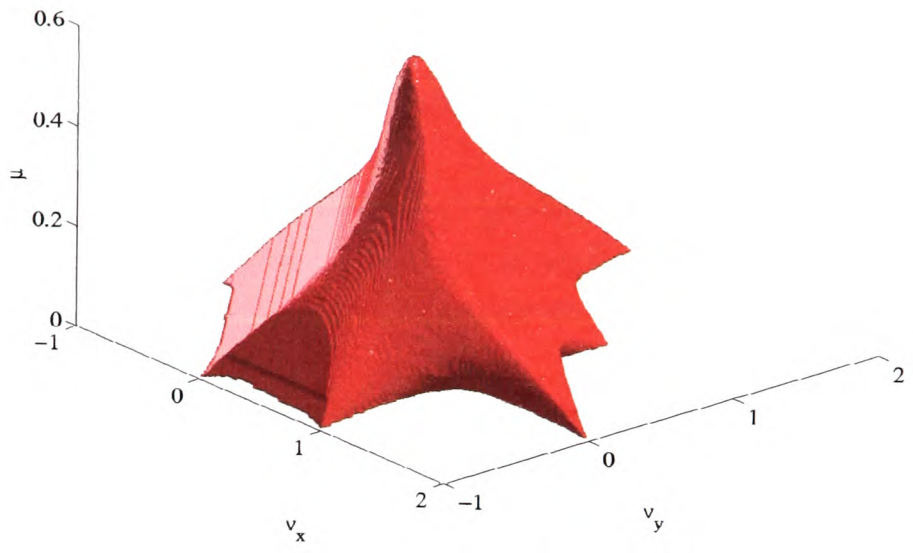
$$(\nu_x^2 + 2\mu_x) + (\nu_y^2 + 2\mu_y) + \frac{2}{3}\nu_x(1 - \nu_x^2 - 6\mu_x) + \frac{2}{3}\nu_y(1 - \nu_y^2 - 6\mu_y) \leq 1 \quad (6.24)$$

Proof: The amplification factor is given by:

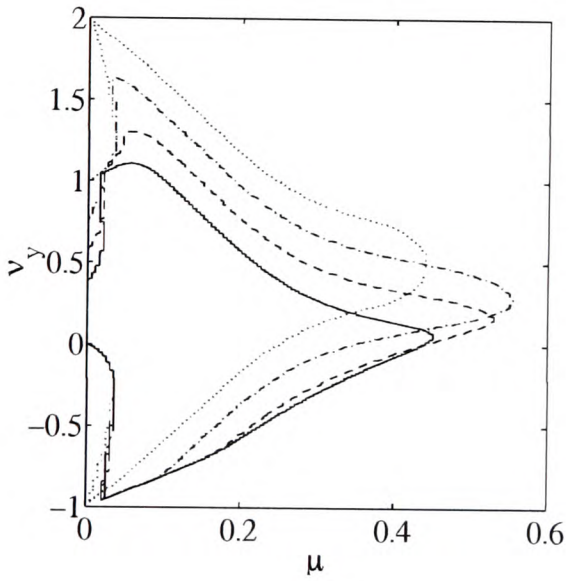
$$\begin{aligned}
\kappa(\theta_x,\theta_y) \;=\; & 1 - \frac{\nu_x}{2}(e^{i\theta_x} - e^{-i\theta_x}) - \frac{\nu_y}{2}(e^{i\theta_y} - e^{-i\theta_y}) \\
& + (\frac{\nu_x^2}{2} + \mu_x)(e^{i\theta_x} - 2 + e^{-i\theta_x}) + (\frac{\nu_y^2}{2} + \mu_y)(e^{i\theta_y} - 2 + e^{-i\theta_y}) \\
& + \frac{\nu_x\nu_y}{4}(e^{i\theta_x} - e^{-i\theta_x})(e^{i\theta_y} - e^{-i\theta_y}) \\
& + \frac{1}{6}\nu_x(1 - \nu_x^2 - 6\mu_x)(e^{i\theta_x} - 2 + e^{-i\theta_x})(1 - e^{-i\theta_x}) \\
& + \frac{1}{6}\nu_y(1 - \nu_y^2 - 6\mu_y)(e^{i\theta_y} - 2 + e^{-i\theta_y})(1 - e^{-i\theta_y}) \\
& - \frac{\nu_y}{2}(\mu_x + \frac{1}{2}\nu_x^2)(e^{i\theta_x} - 2 + e^{-i\theta_x})(e^{i\theta_y} - e^{-i\theta_y}) \\
& - \frac{\nu_x}{2}(\mu_y + \frac{1}{2}\nu_y^2)(e^{i\theta_y} - 2 + e^{-i\theta_y})(e^{i\theta_x} - e^{-i\theta_x})
\end{aligned}$$

Simplifying the previous expression for the amplification factor, we obtain,

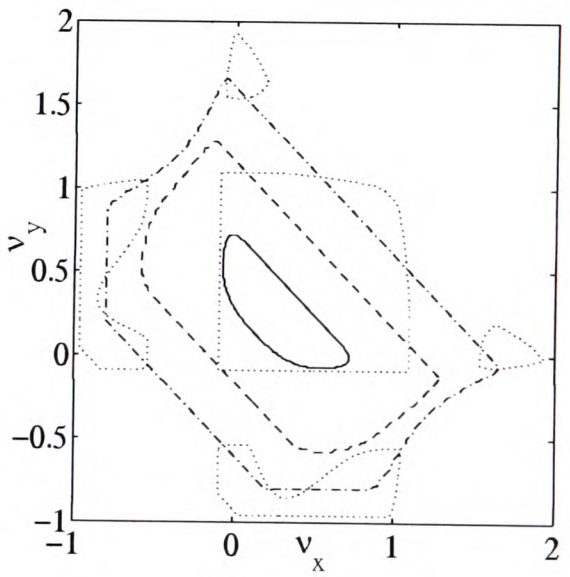
$$\begin{aligned}
\kappa(\theta_x,\theta_y) \;=\; & 1 - i\nu_x \sin\theta_x - i\nu_y \sin\theta_y \\
& + (\nu_x^2 + 2\mu_x)(-1 + \cos\theta_x) + (\nu_y^2 + 2\mu_y)(-1 + \cos\theta_y) \\
& - \nu_x\nu_y \sin\theta_x \sin\theta_y \\
& - \frac{1}{3}\nu_x(1 - \nu_x^2 - 6\mu_x)[(1 - \cos\theta_x)^2 + i\sin\theta_x(1 - \cos\theta_x)] \\
& - \frac{1}{3}\nu_y(1 - \nu_y^2 - 6\mu_y)[(1 - \cos\theta_y)^2 + i\sin\theta_y(1 - \cos\theta_y)] \\
& - 2\nu_y(\mu_x + \frac{1}{2}\nu_x^2)i\sin\theta_y(1 - \cos\theta_x)]
\end{aligned}$$



(a)



(b)



(c)

Figure 6.12: (a) Practical stability analysis for the Polynomial Quickest scheme; (b) projection of the figure (a) on the plane $\mu \circ \nu_y$: $\nu_x = 0$ $(\cdots)$; $\nu_x = 0.2$ $(- \cdot -)$; $\nu_x = 0.4$ $(--)$; $\nu_x = 0.6$ $(-)$; (c) projection of the figure (a) on the plane $\nu_x \circ \nu_y$: $\mu = 0.02$ $(\cdots)$; $\mu = 0.1$ $(- \cdot -)$; $\mu = 0.2$ $(--)$; $\mu = 0.4$ $(-)$.

$$-2\nu_x(\mu_y + \frac{1}{2}\nu_y^2)\mathrm{i}\sin\theta_x(1-\cos\theta_y)].$$

For the particular case $\theta_x = \theta_y = \pi$ we have,

$$\begin{aligned}\kappa(\pi,\pi) &= 1 - 2(\nu_x^2 + 2\mu_x) - 2(\nu_y^2 + 2\mu_y) \\ &\quad -\frac{4}{3}\nu_x(1-\nu_x^2-6\mu_x) - \frac{4}{3}\nu_y(1-\nu_y^2-6\mu_y).\end{aligned}$$

To have $|\kappa(\pi,\pi)| \leq 1$ we need to have (6.24) . □

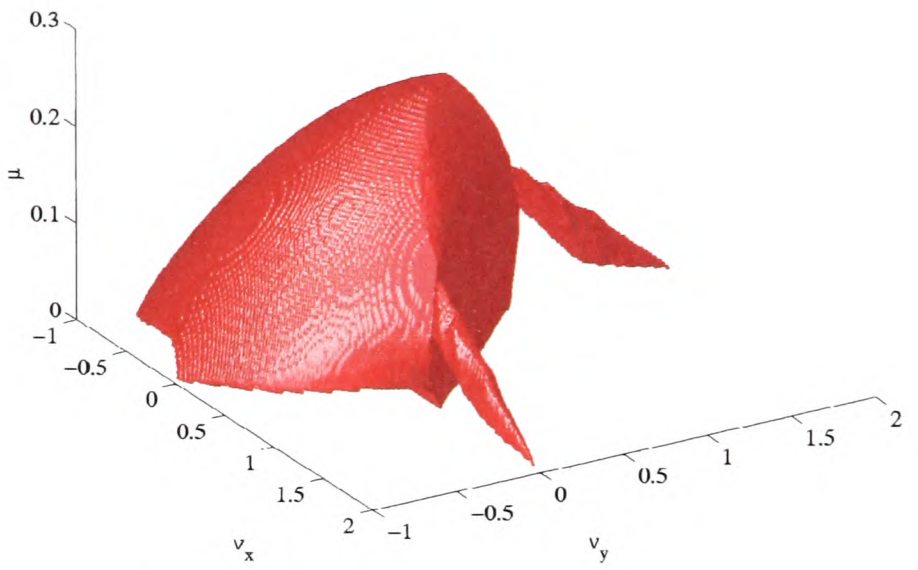
We plot the sufficient and necessary conditions for the Taylor Quickest scheme in figure 6.13. By assuming $\mu_x = \mu_y = \mu$ and $\nu_x = \nu_y = 1/4$ in (6.24) we have the condition $\mu \leq 9/32$. This is the maximum value that μ can take inside the stable region (see figure 6.13).

We turn now to the Quickest scheme suggested by Davis and Moore [13], which is interesting to compare with the Quickest schemes we have devised in this chapter. Their scheme is as follows:

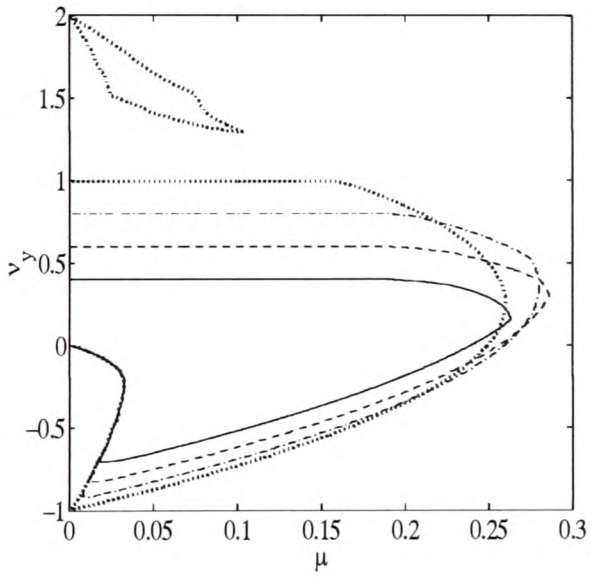
$$\begin{aligned}U_{jk}^{n+1} &= U_{jk}^n - \nu_x\Delta_{x0}U_{jk}^n - \nu_y\Delta_{y0}U_{jk}^n \\ &\quad +(\frac{1}{2}\nu_x^2 + \mu_x)\delta_x^2U_{jk}^n + (\frac{1}{2}\nu_y^2 + \mu_y)\delta_y^2U_{jk}^n \\ &\quad +\frac{1}{6}\nu_x(1-\nu_x^2-6\mu_x)\delta_x^2\Delta_{x-}U_{jk}^n + \frac{1}{6}\nu_y(1-\nu_y^2-6\mu_y)\delta_y^2\Delta_{y-}U_{jk}^n.\end{aligned} \tag{6.25}$$

Davis and Moore [13] generalised the one dimensional Quickest scheme to two dimensions, although in forming the numerical scheme some $O(\Delta t^2)$ spatial cross derivatives have been omitted to create a simpler algorithm. Formally this reduces the temporal accuracy of the scheme to $O(\Delta t)$. Consequently Davis and Moore employed small time-steps to minimise the error associated with the neglected $O(\Delta t^2)$ term.

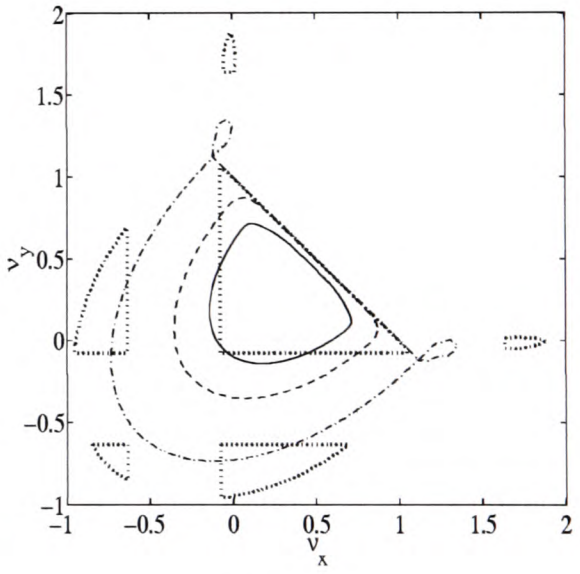
We analyse the stability for this scheme using the von Neumann analysis in the same way as for the previous Quickest schemes. To find a necessary condition for the Davis and Moore Quickest scheme, we evaluate the amplification factor for the phase angles of high frequency $\theta_x = \theta_y = \pi$. In that way we obtain the same necessary condition as for the Taylor Quickest scheme, given by (6.13). The stability region for this scheme is plotted in figure 6.14. According to this figure it seems that another necessary condition for stability is that for $\mu \leq 1/4$, we need to have $\nu_x + \nu_y \leq 1$. For small μ the stable region seems to be quite small, compared with the previously studied schemes.



(a)

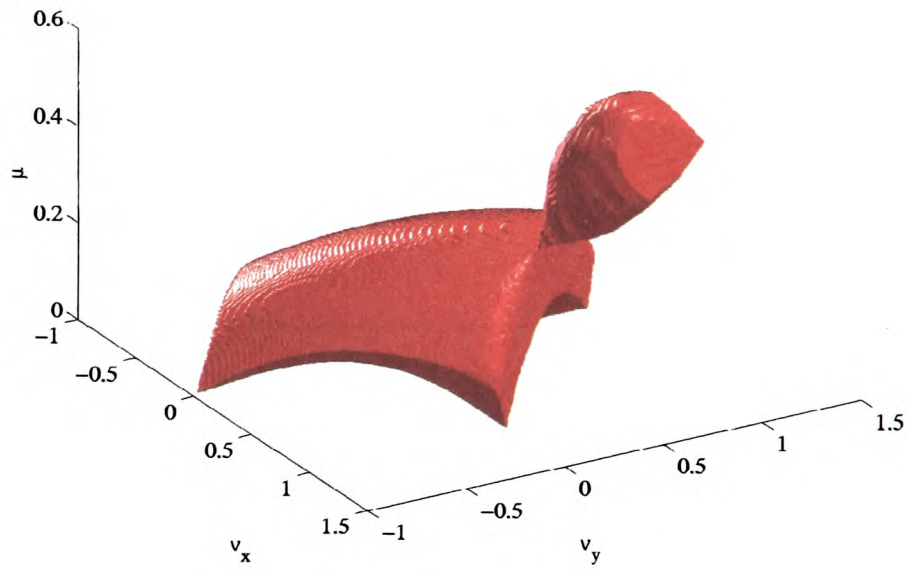


(b)

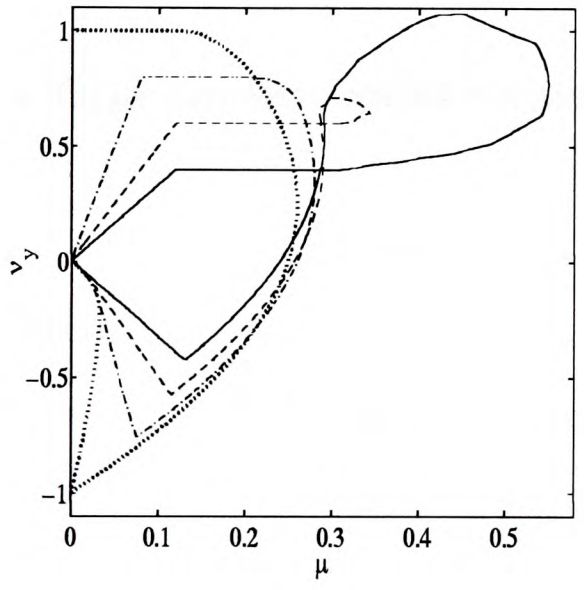


(c)

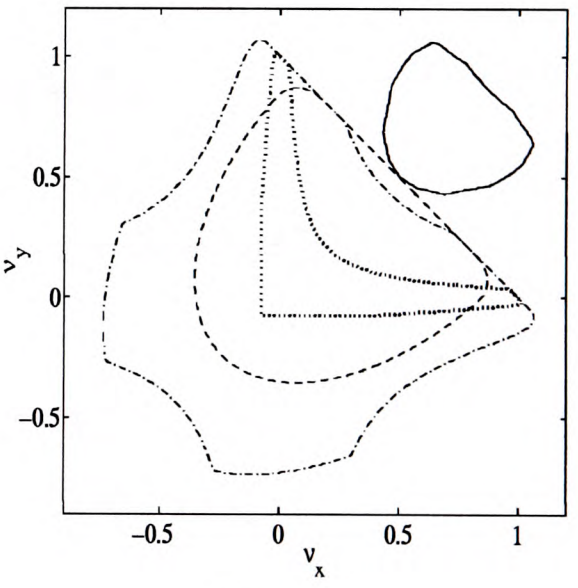
Figure 6.13: (a) Practical stability analysis for the Taylor Quickest scheme; (b) projection of the figure (a) on the plane $\mu \circ \nu_y$: $\nu_x = 0$ $(\cdots)$; $\nu_x = 0.2$ $(- \cdot -)$; $\nu_x = 0.4$ $(--)$; $\nu_x = 0.6$ $(-)$; (c) projection of the figure (a) on the plane $\nu_x \circ \nu_y$: $\mu = 0.02$ $(\cdots)$; $\mu = 0.1$ $(- \cdot -)$; $\mu = 0.2$ $(--)$; $\mu = 0.24$ $(-)$.



(a)



(b)



(c)

Figure 6.14: (a) Practical stability analysis for the Davis and Moore Quickest scheme; (b) projection of the figure (a) on the plane $\mu \circ \nu_y$: $\nu_x = 0$ $(\cdots)$; $\nu_x = 0.2$ $(- \cdot -)$; $\nu_x = 0.4$ $(--)$; $\nu_x = 0.6$ $(-)$; (c) projection of the figure (a) on the plane $\nu_x \circ \nu_y$: $\mu = 0.02$ $(\cdots)$; $\mu = 0.1$ $(- \cdot -)$; $\mu = 0.2$ $(--)$; $\mu = 0.24$ $(-)$.

In this chapter we provided stability regions for various finite difference schemes.

The Lax-Wendroff schemes considered present regions of stability that are sufficient and necessary conditions, with a shape that appears to have a simple form, although we found considerable difficulties, when we attempted to find these conditions analytically. The main source of these difficulties was related to the majorisation of the Fourier terms associated with the mixed derivatives.

The Quickest schemes present more awkward regions and they do not seem to show an obvious regularity leading us to conjecture that to provide the analytical necessary and sufficient conditions is an extremely difficult task.

To conclude this chapter we summarise the numerical schemes derived here and that we use in the following chapters (in this summary we assume both velocities positive).

- Polynomial Lax-Wendroff scheme (see figure 6.1)

$$U_{jk}^{n+1} = [1 + \mathcal{P}_x + \mathcal{P}_y + \nu_x \nu_y \Delta_{x-} \Delta_{y-}] U_{jk}^n.$$

- Taylor Lax-Wendroff scheme (see figure 6.3)

$$U_{jk}^{n+1} = [1 + \mathcal{P}_x + \mathcal{P}_y + \nu_x \nu_y \Delta_{x0} \Delta_{y0}] U_{jk}^n.$$

where

$$\begin{aligned} \mathcal{P}_x &= -\nu_x \Delta_{x0} + (\frac{1}{2}\nu_x^2 + \mu_x)\delta_x^2 \\ \mathcal{P}_y &= -\nu_y \Delta_{y0} + (\frac{1}{2}\nu_y^2 + \mu_y)\delta_y^2 \end{aligned}$$

- Polynomial Quickest scheme (see figure 6.2)

$$\begin{aligned} U_{jk}^{n+1} &= [1 + \mathcal{Q}_x + \mathcal{Q}_y + \frac{1}{2}\nu_x \nu_y (\Delta_{y+}\Delta_{x-} + \Delta_{x+}\Delta_{y-}) \\ &\quad -\nu_y(\mu_x + \frac{1}{2}\nu_x^2)\delta_x^2 \Delta_{y-} - \nu_x(\mu_y + \frac{1}{2}\nu_y^2)\delta_y^2 \Delta_{x-}] U_{jk}^n. \end{aligned}$$

• Taylor Quickest scheme (see figure 6.4)

$$\begin{aligned} U_{jk}^{n+1} &= [1 + \mathcal{Q}_x + \mathcal{Q}_y + \nu_x\nu_y\Delta_{x0}\Delta_{y0} \\ &\quad -\nu_y(\mu_x + \frac{1}{2}\nu_x^2)\delta_x^2\Delta_{y0} - \nu_x(\mu_y + \frac{1}{2}\nu_y^2)\delta_y^2\Delta_{x0}]U_{jk}^n. \end{aligned}$$

where

$$\begin{aligned} \mathcal{Q}_x &= \mathcal{P}_x + \frac{1}{6}\nu_x(1 - \nu_x^2 - 6\mu_x)\delta_x^2\Delta_{x-} \\ \mathcal{Q}_y &= \mathcal{P}_y + \frac{1}{6}\nu_y(1 - \nu_y^2 - 6\mu_y)\delta_y^2\Delta_{y-} \end{aligned}$$

In the next chapter we consider two-dimensional convection-diffusion problems with physical boundary conditions. We know that the presence of boundary conditions interferes with the stability of a finite difference scheme. Although in the presence of boundaries that are not periodic the von Neumann condition is no longer a sufficient condition, it is still quite an important necessary condition.

Chapter 7

The effect on stability of boundaries in two dimensions

In the foregoing chapter we applied the von Neumann method to analyse the stability of Lax-Wendroff and Quickest schemes in a two-dimensional unbounded domain. In this chapter we analyse the stability of those same schemes when used to approximate convection-diffusion problems on a bounded domain with two different boundary conditions on the underlying flow field.

For the Lax-Wendroff schemes, in the case of a bounded domain, it is sufficient to consider only the physical boundary conditions in order to calculate the approximate solution. However, the development of high accuracy schemes, such as Quickest schemes, is accomplished by increasing the local spatial domain needed to compute the time evolution at each point. Of course real flows do not occur in unbounded domains and as soon as boundaries are present it is not possible to use large local domains to compute evolution near boundaries nor is the correct evolution given by schemes derived from an infinite domain. Thus, as in the one-dimensional case, a different numerical approximation may have to be carried out near boundaries, compromising the stability of the higher order scheme used elsewhere. Hence a core issue in the development of practical high order schemes for real flows is an ability to deal with bounded domains.

As we have done in the third chapter for a one-dimensional problem, in this chapter we provide numerical boundary conditions for high order Quickest schemes and examine their effect on the stability of the general scheme.

We also investigate the stability regions for Lax-Wendroff schemes and Quickest schemes when applied to two problems. First, we consider a convection-diffusion problem on a quarter plane where we assume both components of the

convective velocity to be positive and constant. Secondly, we consider a problem defined in a square with non-uniform flow velocity which models a driven cavity, a problem for which we investigate a Navier-Stokes solver in the next chapter.

7.1 Numerical boundary conditions

If we want to discretise the two-dimensional convection-diffusion equation with a high-order method such as a Quickest scheme then, in the same way as in one dimension, we need to introduce numerical boundary conditions in addition to the physical boundary conditions. In this section we introduce the numerical boundary conditions that we shall use with the Quickest schemes in the subsequent sections. To clearly illustrate the numerical boundary conditions, we start by considering a convection-diffusion problem defined in the domain $x \geq 0,\ y \geq 0$.

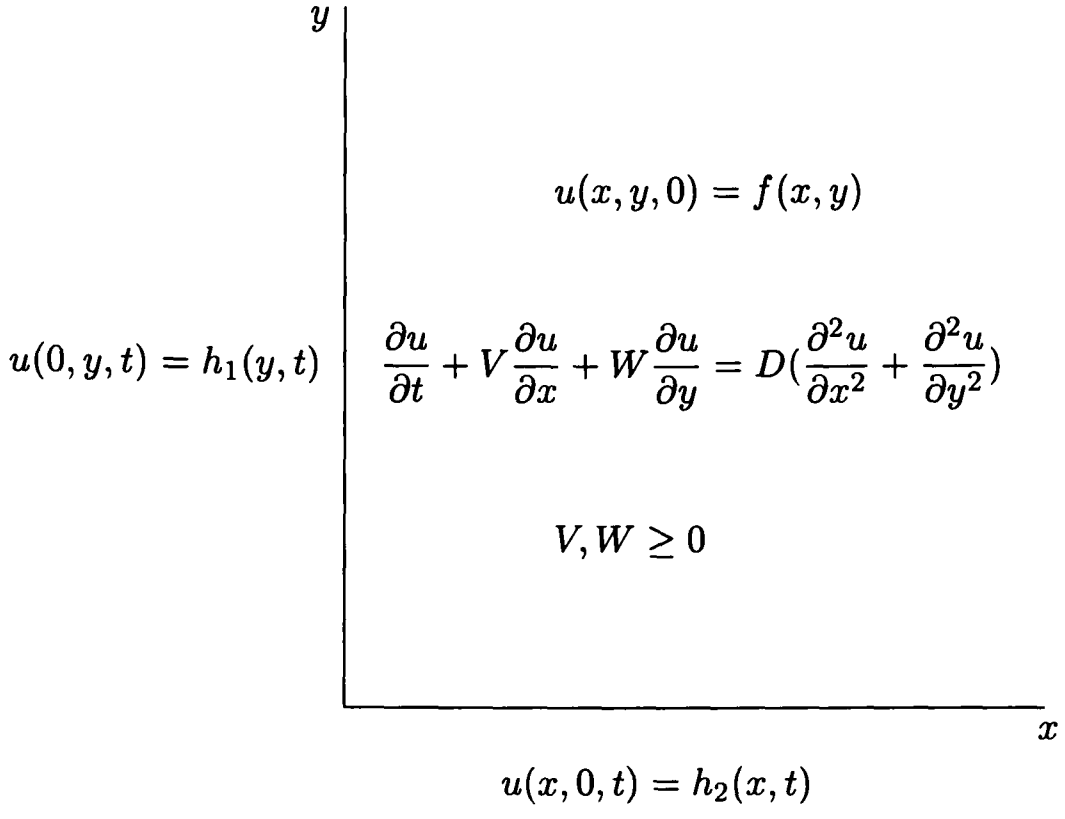
Consider the convection-diffusion equation (6.1) defined on the quarter-plane $x \geq 0,\ y \geq 0$, for $t > 0$, with the initial condition,

$$u(x, y, 0) = f(x, y), \quad x \geq 0,\ y \geq 0,$$

and the wall conditions,

$$u(0, y, t) = h_1(y, t),\ y \geq 0,\ t \geq 0, \tag{7.1}$$

$$u(x, 0, t) = h_2(x, t),\ x \geq 0,\ t \geq 0. \tag{7.2}$$



When applying a Quickest scheme to the two-dimensional problem illustrated above we are assuming that both velocity components V and W are positive and constant. Both Quickest schemes, the Polynomial Quickest scheme and the Taylor Quickest scheme, are generated according to the direction of the velocity, as described in the sixth chapter.

We denote the points of the discrete mesh as

$$\{U_{jk}^n, \quad j \in \mathbb{N},\ k \in \mathbb{N}\}.$$

The discrete boundary conditions associated with the physical boundary conditions (7.1) and (7.2) are

$$U_{0k}^{n+1} = h_1(k\Delta y, (n+1)\Delta t), \quad k \in \mathbb{N}, \tag{7.3}$$

$$U_{j0}^{n+1} = h_2(j\Delta x, (n+1)\Delta t), \quad j \in \mathbb{N}. \tag{7.4}$$

Using this velocity field, the family of values of the approximate solution $\{U_{1k}^{n+1}, k \in \mathbb{N}\}$ and $\{U_{j1}^{n+1}, j \in \mathbb{N}\}$, cannot be obtained using the same discretisation that is used to evaluate the approximate solution at the other interior mesh points since at the first interior points of the mesh it is necessary to have the approximated solution evaluated at some nonexistent points of the mesh, namely

$\{(x_{-1}, y_k), k \in \mathbb{N}\}$ and $\{(x_j, y_{-1}), j \in \mathbb{N}\}$. Therefore additionally to the boundary conditions (7.3) and (7.4), we must provide numerical boundary conditions at the first layer of interior mesh points.

In the next sections we describe the numerical boundary conditions that we consider at those points. They can be interpreted as generalisations of the numerical boundary conditions provided in the third chapter for the one-dimensional case.

7.1.1 Downwind third difference

One implementation for the Polynomial Quickest scheme (6.6) is a third difference which does not track the flow direction: at the first interior points of the mesh, the points immediately after the physical boundaries, we bring a forward third-order difference in x or y instead of a backward third-order difference for the approximate solution. In other words it corresponds to using the operators $\delta_x^2\Delta_{x+}$ and $\delta_y^2\Delta_{y+}$ when evaluating the approximate solution $U_{1k}^n, k \in \mathbb{N}$ and $U_{j1}^n, j \in \mathbb{N}$ respectively, instead of the operators $\delta_x^2\Delta_{x-}$ and $\delta_y^2\Delta_{y-}$ that are applied to evaluate the approximate solution elsewhere.

The numerical boundary conditions, to evaluate the grid function $\{U_{1k}^{n+1},\ k = 2, 3, \ldots\}$, are given by

$$\begin{aligned} U_{1k}^{n+1} &= U_{1k}^n - \nu_x\Delta_{x0}U_{1k}^n - \nu_y\Delta_{y0}U_{1k}^n \\ &\quad +(\frac{1}{2}\nu_x^2 + \mu_x)\delta_x^2U_{1k}^n + (\frac{1}{2}\nu_y^2 + \mu_y)\delta_y^2U_{1k}^n \\ &\quad +\frac{1}{2}\nu_x\nu_y\Delta_{y+}\Delta_{x-}U_{1k}^n + \frac{1}{2}\nu_x\nu_y\Delta_{x+}\Delta_{y-}U_{1k}^n \\ &\quad +\frac{1}{6}\nu_x(1 - \nu_x^2 - 6\mu_x)\delta_x^2\Delta_{x+}U_{1k}^n + \frac{1}{6}\nu_y(1 - \nu_y^2 - 6\mu_y)\delta_y^2\Delta_{y-}U_{1k}^n \\ &\quad -\nu_y(\mu_x + \frac{1}{2}\nu_x^2)\delta_x^2\Delta_{y-}U_{1k}^n - \nu_x(\mu_y + \frac{1}{2}\nu_y^2)\delta_y^2\Delta_{x-}U_{1k}^n. \end{aligned} \tag{7.5}$$

The numerical boundary conditions (7.5) are obtained by performing a forward third-order difference in x instead of the backward third-order difference done at the other interior mesh points. Similarly the numerical boundary conditions to evaluate the grid function $\{U_{j1}^{n+1},\ j = 2, 3, \ldots\}$ are obtained by performing a forward third-order difference in y,

$$\begin{aligned} U_{j1}^{n+1} &= U_{j1}^n - \nu_x\Delta_{x0}U_{j1}^n - \nu_y\Delta_{y0}U_{j1}^n \\ &\quad +(\frac{1}{2}\nu_x^2 + \mu_x)\delta_x^2U_{j1}^n + (\frac{1}{2}\nu_y^2 + \mu_y)\delta_y^2U_{j1}^n \end{aligned}$$

$$
\begin{aligned}
&+\frac{1}{2}\nu_x\nu_y\Delta_{y+}\Delta_{x-}U^n_{j1}+\frac{1}{2}\nu_x\nu_y\Delta_{x+}\Delta_{y-}U^n_{j1}\\
&+\frac{1}{6}\nu_x(1-\nu_x^2-6\mu_x)\delta_x^2\Delta_{x-}U^n_{j1}+\frac{1}{6}\nu_y(1-\nu_y^2-6\mu_y)\delta_y^2\Delta_{y+}U^n_{j1}\\
&-\nu_y(\mu_x+\frac{1}{2}\nu_x^2)\delta_x^2\Delta_{y-}U^n_{j1}-\nu_x(\mu_y+\frac{1}{2}\nu_y^2)\delta_y^2\Delta_{x-}U^n_{j1}. \qquad (7.6)
\end{aligned}
$$

The numerical boundary condition for U^{n+1}_{11} is obtained by applying simultaneously a forward third order difference in x and y,

$$
\begin{aligned}
U^{n+1}_{11} \;=\;& U^n_{11}-\nu_x\Delta_{x0}U^n_{11}-\nu_y\Delta_{y0}U^n_{11}\\
&+(\frac{1}{2}\nu_x^2+\mu_x)\delta_x^2U^n_{11}+(\frac{1}{2}\nu_y^2+\mu_y)\delta_y^2U^n_{11}\\
&+\frac{1}{2}\nu_x\nu_y\Delta_{y+}\Delta_{x-}U^n_{11}+\frac{1}{2}\nu_x\nu_y\Delta_{x+}\Delta_{y-}U^n_{11}\\
&+\frac{1}{6}\nu_x(1-\nu_x^2-6\mu_x)\delta_x^2\Delta_{x+}U^n_{11}+\frac{1}{6}\nu_y(1-\nu_y^2-6\mu_y)\delta_y^2\Delta_{y+}U^n_{11}\\
&-\nu_y(\mu_x+\frac{1}{2}\nu_x^2)\delta_x^2\Delta_{y-}U^n_{11}-\nu_x(\mu_y+\frac{1}{2}\nu_y^2)\delta_y^2\Delta_{x-}U^n_{11}. \qquad (7.7)
\end{aligned}
$$

The numerical boundary conditions that we have described for the Polynomial Quickest scheme (6.6) are easily generalised for the Taylor Quickest scheme (6.8). For the latter we also use a forward third-order difference in x to calculate the values $\{U^n_{1k}, k\in\mathbb{N}\}$ and in y to calculate the values $\{U^n_{j1}, j\in\mathbb{N}\}$ instead of a backward third-order difference that we apply at the other interior points for x and y respectively. To calculate the value U^n_{11} we apply the forward third order difference in x and y as in the case of the Polynomial Quickest scheme described above.

When the directions of either of the velocities are negative, we use a backward third-order difference instead of a forward third-order difference.

7.1.2 Lax-Wendroff

At the first points of the mesh we use quadratic interpolation, instead of cubic interpolation, for which we do not have the necessary points, obtaining a Lax-Wendroff scheme instead of a Quickest scheme. For the approximate solution values $\{U^{n+1}_{1k},\ k\in\mathbb{N}\}$ we have,

$$
\begin{aligned}
U^{n+1}_{1k} \;=\;& [1-(\nu_x\Delta_{x0}+\nu_y\Delta_{y0})+(\frac{1}{2}\nu_x^2+\mu_x)\delta_x^2\\
&+(\frac{1}{2}\nu_y^2+\mu_y)\delta_y^2+\nu_x\nu_y\Delta_{x-}\Delta_{y-}]U^n_{1k}, \qquad (7.8)
\end{aligned}
$$

and for the approximate solution values $\{U_{j1}^{n+1},\ j \in \mathbb{N}\}$,

$$\begin{aligned} U_{j1}^{n+1} &= [1 - (\nu_x \Delta_{x0} + \nu_y \Delta_{y0}) + (\frac{1}{2}\nu_x^2 + \mu_x)\delta_x^2 \\ &\quad +(\frac{1}{2}\nu_y^2 + \mu_y)\delta_y^2 + \nu_x \nu_y \Delta_{x-}\Delta_{y-}]U_{j1}^n. \end{aligned} \tag{7.9}$$

We can consider a different Lax-Wendroff scheme at the first points of the interior scheme. Apart from the Lax-Wendroff scheme above we also consider other types of Lax-Wendroff schemes as numerical boundary conditions in further sections.

7.1.3 The fictitious points

In $\mathbb{R}^2$ the exact solution of the problem (6.1), (6.2) is given by:

$$u(x,y,t) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} u(\xi,\tau,0)G(x-\xi, y-\tau, t)d\xi d\tau, \tag{7.10}$$

where

$$G(s,p,t) = \frac{1}{4Dt\pi}\mathrm{e}^{-(s-Vt)^2/4Dt}\mathrm{e}^{-(s-Wt)^2/4Dt}.$$

For the problem under consideration we have the conditions,

$$u(0,y,t) = h_1(y,t), \quad u(x,0,t) = h_2(x,t),$$

and therefore

$$h_1(y,t) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} u(\xi,\tau,0)G(-\xi, y-\tau, t)d\xi d\tau, \tag{7.11}$$

$$h_2(x,t) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} u(\xi,\tau,0)G(x-\xi, -\tau, t)d\xi d\tau. \tag{7.12}$$

As already mentioned, in order to calculate the approximate solution values U_{1k}^n and U_{j1}^n we need to have the noexisting values of the approximated solution $U_{-1,k}^n$ and $U_{j,-1}^n$. We approximate the solution u in (7.11) and (7.12) by performing a certain polynomial interpolation around the points (x_j, y_0) in (7.12) and (x_0, y_k) in (7.11).

The values $\{U_{-1,k}^n,\ k \in \mathbb{N}\}$ are obtained from using the Lax-Wendroff scheme to evaluate $\{U_{0k}^n,\ k \in \mathbb{N}\}$,

$$\begin{aligned} U_{0k}^{n+1} &= [1 - \nu_x \Delta_{x0} - \nu_y \Delta_{y0} + (\frac{1}{2}\nu_x^2 + \mu_x)\delta_x^2 + (\frac{1}{2}\nu_y^2 + \mu_y)\delta_y^2 \\ &\quad +\nu_x \nu_y \Delta_{x+}\Delta_{y-}]U_{0k}^n. \end{aligned} \tag{7.13}$$

Similarly the values $\{U_{j,-1},\ j \in \mathbb{N}\}$ are obtained from using the Lax-Wendroff scheme to evaluate $\{U_{j0},\ j \in \mathbb{N}\}$,

$$\begin{aligned} U_{j0}^{n+1} &= [1 - \nu_x\Delta_{x0} - \nu_y\Delta_{y0} + (\frac{1}{2}\nu_x^2 + \mu_x)\delta_x^2 + (\frac{1}{2}\nu_y^2 + \mu_y)\delta_y^2 \\ &\quad +\nu_x\nu_y\Delta_{x-}\Delta_{y+}]U_{j0}^n. \end{aligned} \tag{7.14}$$

To calculate $\{U_{1k}^n, k = 2, 3, \ldots\}$ and $\{U_{j1}^n, j = 2, 3, \ldots\}$ we use the fictitious points $\{U_{-1,k}^n, k = 2, 3, \ldots\}$ and $\{U_{j,-1}^n, j = 2, 3, \ldots\}$ given by the equations (7.13) and (7.14) respectively. Note that the calculation of U_{11}^n requires the use of both fictitious points $U_{-1,1}$ and $U_{1,-1}$ and consequently we need to apply (7.13) and (7.14) simultaneously.

The different Lax-Wendroff schemes obtained above were based in the interpolation at six points where the sixth point is chosen according to the sign of the flow velocity, in particular this determines the fact that we use the operator $\Delta_{x+}\Delta_{y-}$ in (7.13) and $\Delta_{x-}\Delta_{y+}$ in (7.14).

7.1.4 Summary

The numerical boundary conditions suggested above are for a problem with left and bottom physical boundaries and positive and constant velocities. However, it is not difficult to adjust these ideas to a slightly different problem, namely a problem with a different flow velocity direction. Later we also apply similar kinds of numerical boundary conditions to a problem that involves a flow with non-uniform velocity. This non-uniform problem has four physical boundary conditions with flow velocity changing direction at each time-step. This requires that we choose a numerical boundary condition according to the direction of the flow velocity at each local mesh point.

In the next section we apply the numerical boundary conditions to the problem we considered here, with a constant and positive flow velocity in both coordinates x and y, and we investigate the stability of the general scheme.

7.2 A problem with constant flow velocity

Consider the two-dimensional flow described in section 7.1, with positive velocity components V and W. Let $\Delta x = \Delta y$, then we have $\mu_x = \mu_y = \mu$. Suppose we have the simplified form of the wall conditions,

$$u(0, y, t) = 0, \tag{7.15}$$

$$u(x, 0, t) = 0. \tag{7.16}$$

Then, for the discrete problem we have the Dirichlet boundary conditions,

$$U_{0k}^{n+1} = 0, \quad U_{j0}^{n+1} = 0, \quad j,k = 1,\ldots,N, \quad n = 0,1,\ldots.$$

In order to build the discrete evolution operators associated with the Lax-Wendroff schemes and the Quickest schemes, we consider the boundary conditions

$$U_{Nk}^{n+1} = 0, \quad U_{jN}^{n+1} = 0, \quad j,k = 1,\ldots,N, \quad n = 0,1,\ldots.$$

The schemes can now be written as matrix equations

$$U^{n+1} = BU^n, \quad n = 0,1\ldots, \tag{7.17}$$

where $U^n = \{U_{11}^n,\ldots,U_{1N}^n,\ldots,U_{21}^n,\ldots,U_{2N}^n,\ldots,U_{N1}^n,\ldots,U_{NN}^n\}$, and B is an $N^2 \times N^2$ matrix that depends on the scheme used.

For the Lax-Wendroff schemes, the region of stability is given by the von Neumann method since we consider periodic boundary conditions. Therefore the stability regions for the Lax-Wendroff schemes are the regions plotted in figure 6.10 and figure 6.11. These regions can also be determined by the condition $||B|| \leq 1$, where B is the iterative matrix in (7.17) for the respective Lax-Wendroff scheme and $||\cdot||$ is the L_2-norm.

For the Quickest schemes we need to consider additional numerical boundary conditions as already mentioned and therefore the von Neumann stability analysis is no longer a sufficient and necessary condition for stability. However, as we have seen in the one-dimensional case, the von Neumann condition is still a necessary condition for the stability of the scheme.

The stability analysis of the Polynomial Quickest scheme and Taylor Quickest scheme in the next sections is based on the analysis of the norm and spectrum of the iterative matrix B in (7.17). As mentioned in the third chapter, a sufficient condition for the stability of the scheme is given by

$$||B|| \leq 1,$$

and a necessary condition is given by

$$\rho(B) \leq 1.$$

In the following sections we investigate the stability for the Polynomial Quickest scheme and Taylor Quickest scheme when the different types of numerical boundary conditions are applied, observing their effects on the stability of the general scheme. The matrix size we had selected to illustrate the stability regions is such that larger sizes would not lead to significant changes in the results.

7.2.1 Polynomial Quickest scheme

In this section we consider the Polynomial Quickest scheme with the numerical boundary conditions described in section 7.1, observing the differences between them. We consider the Polynomial Quickest scheme with positive velocities and therefore the figures that we plot below are for ν_x and ν_y positive.

First, we plot in figure 7.1 the stability region given by the von Neumann method for the Polynomial Quickest scheme since it is still a necessary condition for the stability of the scheme when associated with numerical boundary conditions.

Downwind numerical boundary condition

We start to consider the Polynomial Quickest scheme with the numerical boundary condition described in section 7.1.1, the downwind third difference. It consists in bringing a forward third order difference instead of a backward third order difference in x to calculate the values $U^n_{1k}, k \in \mathbb{N}$ and in y to calculate the values $U^n_{j1}, j \in \mathbb{N}$. We have a sufficient condition for the stability of this scheme plotted in figure 7.2a, $||B|| \leq 1$ and a necessary condition plotted in figure 7.2b, $\rho(B) \leq 1$. These regions are the regions inside the lines plotted in the pictures. Therefore the stability region for the scheme considered contains the region plotted in figure 7.2a and is contained in the region plotted in figure 7.2b. Additionally we take in consideration the von Neumann stability region plotted in figure 7.1, since the stability region also needs to be inside that region.

Note that in the one dimensional case the region for very small μ and close to zero, where $||B|| \geq 1$ (see figure 7.2a), did not cause instability problems when experiments were run.

A necessary condition that it plausibly also a sufficient condition for the stability of the Polynomial Quickest scheme with the downwind numerical boundary condition, is given by the intersection of the von Neumann condition (see figure 7.1) and the condition $\rho(B) \leq 1$ (see figure 7.2b). Comparing the resulting stability region with the region given by the von Neumann method, it seems that when we choose this boundary condition we essentially loose some of the stability region for ν_x and ν_y bigger than one.

It is also worth noting that in the one-dimensional case, the presence of this numerical boundary condition seems to lead to a more significant loss of stability. Also, in the two-dimensional problem we do not apply the forward third-order difference in x and y simultaneously except for the approximated solution value

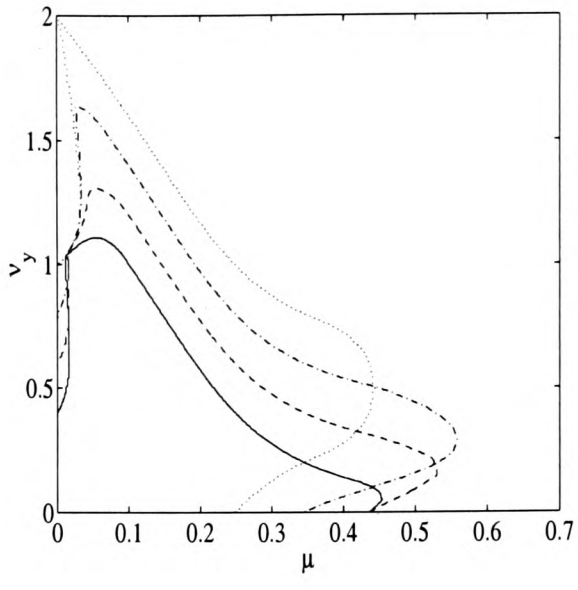


Figure 7.1: Von Neumann stability condition for the Polynomial Quickest scheme for $\nu_x = 0$ $(\cdots)$; $\nu_x = 0.2$ $(- \cdot -)$; $\nu_x = 0.4$ $(--)$; $\nu_x = 0.6$ $(-)$.

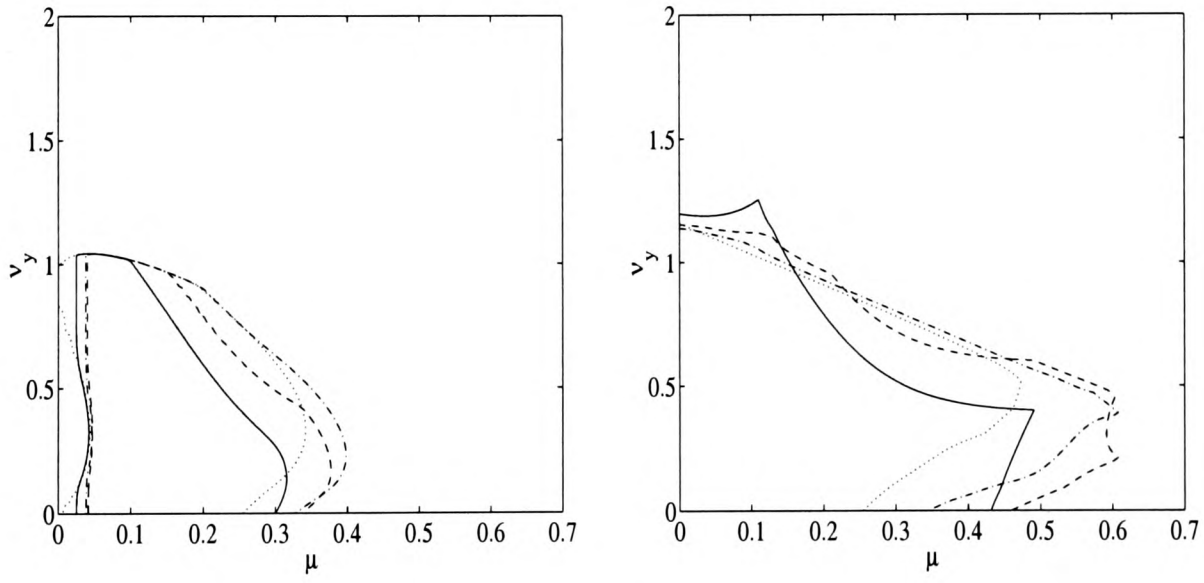


Figure 7.2: Polynomial Quickest scheme with the downwind numerical boundary condition for a matrix of size $10^2 \times 10^2$. Lines are plotted for $\nu_x = 0$ $(\cdots)$; $\nu_x = 0.2$ $(- \cdot -)$; $\nu_x = 0.4$ $(--)$; $\nu_x = 0.6$ $(-)$: (a) $||B|| \leq 1$; (b) $\rho(B) \leq 1$

U_{11}^n. At the other points we keep the upwind third difference in the y direction and x direction respectively.

Lax-Wendroff numerical boundary condition

We consider the Lax-Wendroff numerical boundary condition mentioned in section 7.1.2. As in the previous case we plot a sufficient condition in figure 7.3a, $||B|| \leq 1$, and a necessary condition in figure 7.3b., $\rho(B) \leq 1$. Considering the stability region that is given by the intersection of the von Neumann condition for the Polynomial Quickest scheme in figure 7.1 with the condition $\rho(B) \leq 1$ displayed in figure 7.3b we observe that we have a smaller stability region with this numerical boundary condition than with the numerical boundary condition considered previously.

The loss of stability region when compared with the stability region given by the von Neumann method seems to be of the same magnitude of loss we have observed in the one dimensional case. We also notice that the value that μ can take for the scheme to be stable is half of the value that μ could take in one dimension. This is also true when the stability regions given by the von Neumann method for the one dimensional Quickest scheme and for the two dimensional Polynomial Quickest scheme are compared.

Numerical boundary condition with fictitious points

Now, we proceed to investigate the stability of the Polynomial Quickest scheme with the numerical boundary condition involving the fictitious points, obtained by the conditions in section 7.1.3. Taking in consideration the physical boundary conditions (7.15) and (7.16) it follows that

$$\begin{aligned} 0 = \quad U_{j0}^{n+1} = \quad & \frac{1}{2}(\nu_y + \nu_y^2 + 2\mu)U_{j,-1}^n + \frac{1}{2}(-\nu_y + \nu_y^2 + 2\mu + 2\nu_x\nu_y)U_{j1}^n \\ & -\nu_x\nu_y U_{j-11}^n; \end{aligned} \tag{7.18}$$

$$\begin{aligned} 0 = \quad U_{0k}^{n+1} = \quad & \frac{1}{2}(\nu_x + \nu_x^2 + 2\mu)U_{-1k}^n + \frac{1}{2}(-\nu_x + \nu_x^2 + 2\mu + 2\nu_x\nu_y)U_{1k}^n \\ & -\nu_x\nu_y U_{1k-1}^n. \end{aligned} \tag{7.19}$$

From the previous equations we evaluate the fictitious points that we need to use when calculating the approximate solution at the mesh points next to the boundaries.

We plot the sufficient condition $||B|| \leq 1$ in figure 7.4a and the necessary

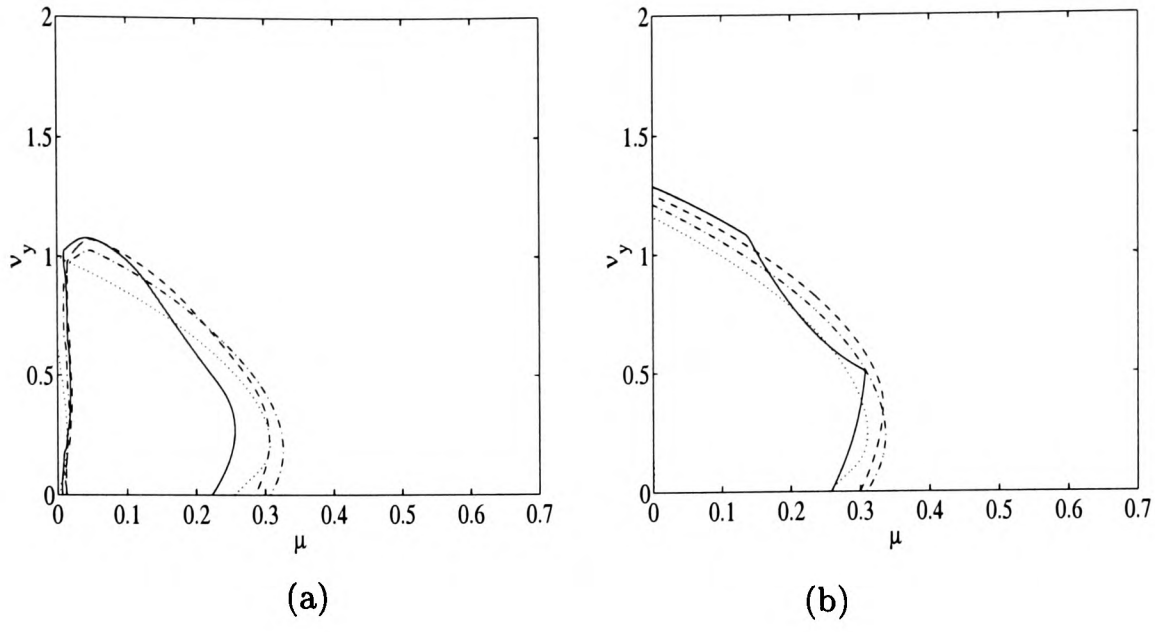


Figure 7.3: Polynomial Quickest scheme with Lax-Wendroff numerical boundary condition for a matrix of size $10^2 \times 10^2$. Line are plotted for $\nu_x = 0$ $(\cdots)$; $\nu_x = 0.2$ $(- \cdot -)$; $\nu_x = 0.4$ $(--)$; $\nu_x = 0.6$ $(-)$. (a) $||B|| \leq 1$; (b) $\rho(B) \leq 1$.

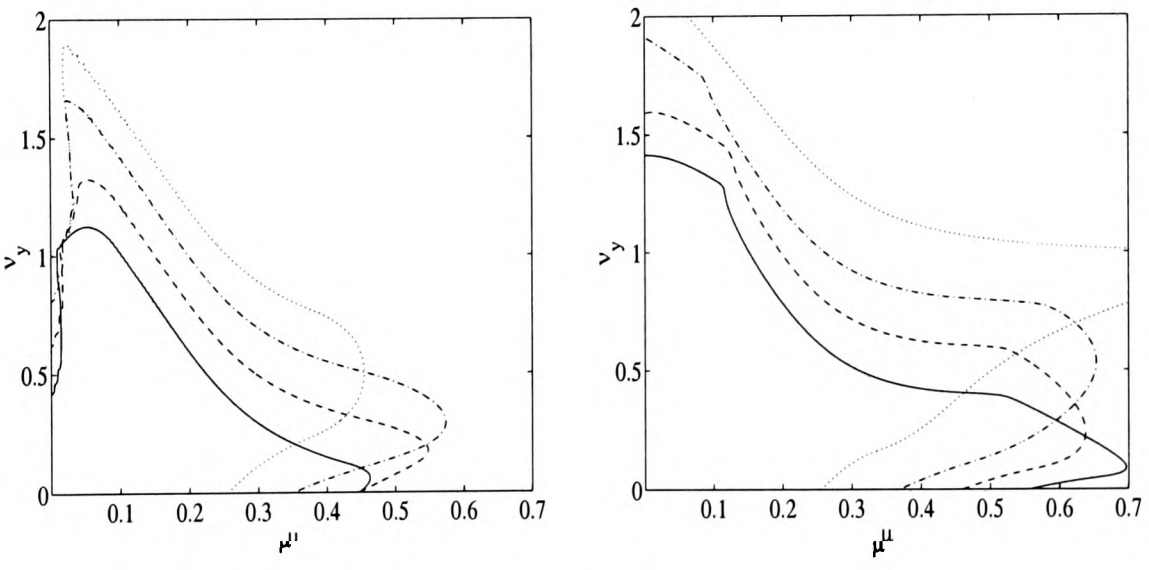


Figure 7.4: Polynomial Quickest scheme with fictitious point numerical boundary condition for a matrix of size $8^2 \times 8^2$. Lines are plotted for $\nu_x = 0$ $(\cdots)$; $\nu_x = 0.2$ $(- \cdot -)$; $\nu_x = 0.4$ $(--)$; $\nu_x = 0.6$ $(-)$ (a) $||B|| \leq 1$; (b) $\rho(B) \leq 1$.

condition $\rho(B) \leq 1$ in figure 7.4b. We can observe some similarity in the figure 7.4a and the von Neumann stability region given in figure 7.1. Consequently we can conclude that the stability region in this case is determined by the region $||B|| \leq 1$ as had happened for the one-dimensional case. This is the numerical boundary condition that offers a larger region of stability.

In the next section we consider the Taylor Quickest scheme associated with the same numerical boundary conditions.

7.2.2 Taylor Quickest scheme

In the previous chapter we observed that the von Neumann stability region for the Taylor Quickest scheme was smaller than the stability region for the Polynomial Quickest scheme. Therefore when we introduce some additional numerical boundary conditions we expect the stability regions to be smaller than the stability regions of the Polynomial Quickest scheme with numerical boundary conditions.

The downwind numerical boundary condition and the numerical boundary condition with the fictitious points for the Taylor Quickest scheme are not exactly the same as the boundaries we used under the same title for the Polynomial Quickest scheme, but are deduced in the same manner. The Lax-Wendroff numerical boundary condition is still exactly the same.

We first plot, in figure 7.5, the von Neumann stability region for the Taylor Quickest scheme, since it is a necessary stability condition for this scheme with numerical boundary conditions.

Downwind numerical boundary condition

The downwind numerical boundary condition is deduced in the way described in section 7.1.1 by taking a forward third order difference in x or y as necessary, that is, when there are no sufficient points to do a backward third order difference. This numerical boundary condition is not exactly the same as the one considered for the previous scheme since we keep the central difference, instead of the upwind difference, for the mixed derivatives.

In figure 7.6a we plot the sufficient condition $||B|| \leq 1$ and in figure 7.6b the necessary condition $\rho(B) \leq 1$. The necessary condition $\rho(B) \leq 1$ intersected with the von Neumann condition gives us a necessary stability condition that, as we expected, originates a smaller stability region than the stability region for the Polynomial Quickest scheme with the similar numerical boundary condition.

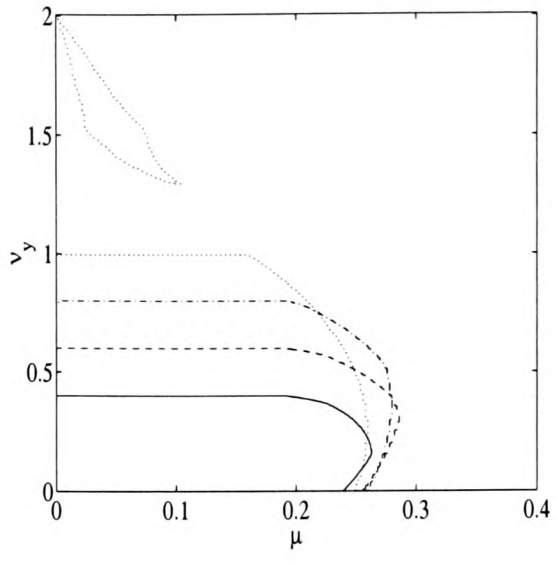


Figure 7.5: Von Neumann stability condition for the Taylor Quickest scheme for: $\nu_x = 0$ $(\cdots)$; $\nu_x = 0.2$ $(- \cdot -)$; $\nu_x = 0.4$ $(--)$; $\nu_x = 0.6$ $(-)$.

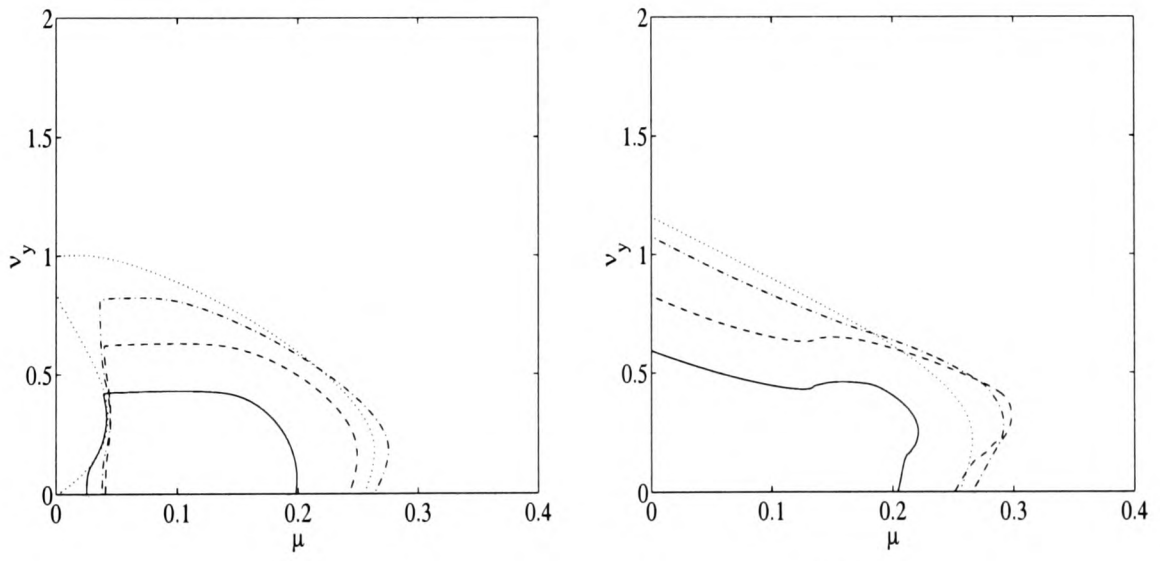


Figure 7.6: Taylor Quickest scheme with the downwind numerical boundary condition for a matrix of size $10^2 \times 10^2$. Lines are plotted for $\nu_x = 0$ $(\cdots)$; $\nu_x = 0.2$ $(- \cdot -)$; $\nu_x = 0.4$ $(--)$; $\nu_x = 0.6$ $(-)$. (a) $||B|| \leq 1$; (b) $\rho(B) \leq 1$.

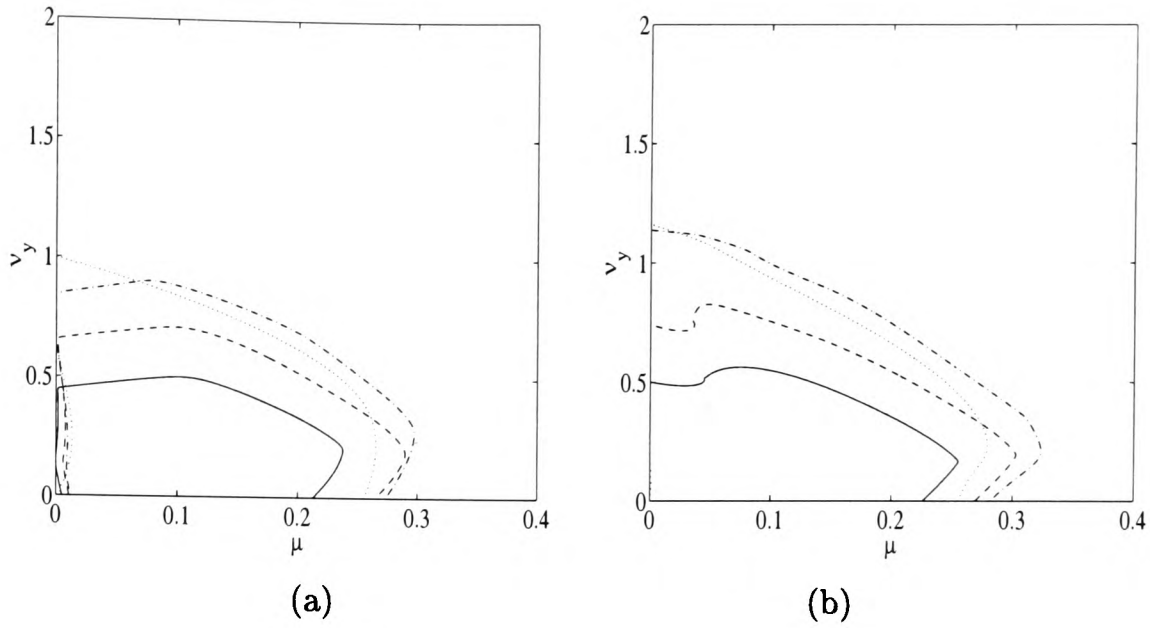


Figure 7.7: Taylor Quickest scheme with the Lax-Wendroff numerical boundary condition for a matrix of size $8^2 \times 8^2$. Lines are plotted for $\nu_x = 0$ $(\cdots)$; $\nu_x = 0.2$ $(- \cdot -)$; $\nu_x = 0.4$ $(--)$; $\nu_x = 0.6$ $(-)$. (a) $||B|| \leq 1$; (b) $\rho(B) \leq 1$.

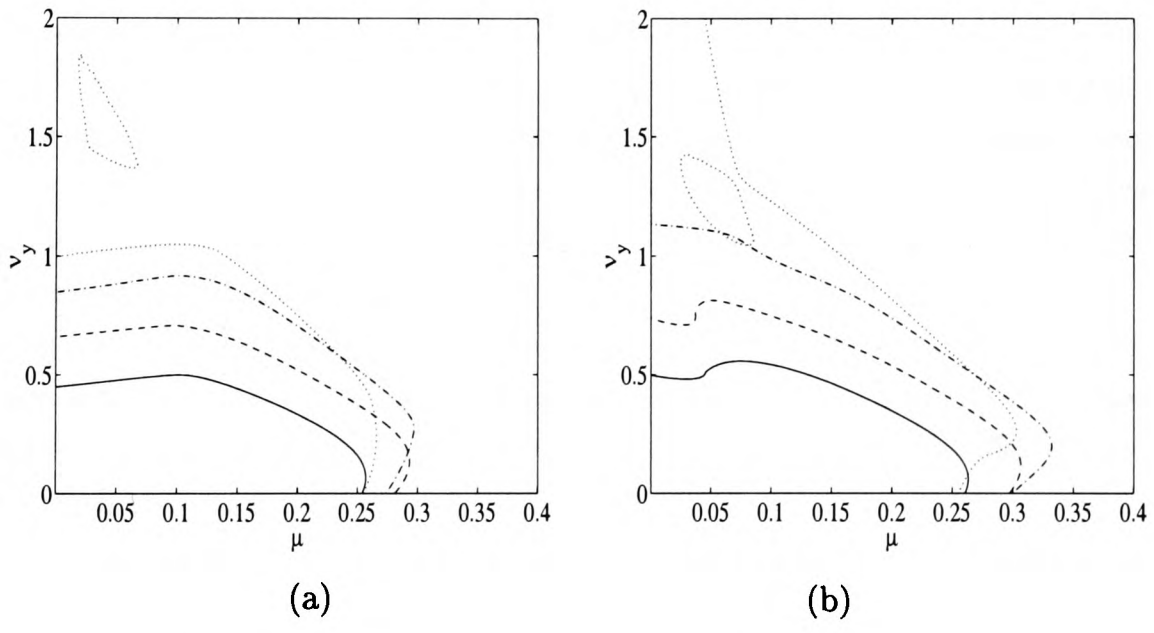


Figure 7.8: Taylor Quickest scheme with the fictitious point numerical boundary condition for a matrix of size $10^2 \times 10^2$. Lines are plotted for $\nu_x = 0$ $(\cdots)$; $\nu_x = 0.2$ $(- \cdot -)$; $\nu_x = 0.4$ $(--)$; $\nu_x = 0.6$ $(-)$. (a) $||B|| \leq 1$; (b) $\rho(B) \leq 1$.

Lax-Wendroff numerical boundary condition

The Lax-Wendroff numerical boundary condition that we apply is described in section 7.1.2. In figure 7.7a we plot a sufficient condition for the stability of the numerical scheme, $||B|| \leq 1$, and in figure 7.7b the necessary condition $\rho(B) \leq 1$. The regions determined by the conditions $||B|| \leq 1$ and $\rho(B) \leq 1$ for the scheme with the Lax-Wendroff numerical boundary condition are close to the regions determined for the downwind numerical boundary condition. For this case we also notice that the regions given by the condition $||B|| \leq 1$ are nearly the same as the regions given by the von Neumann method.

Numerical boundary condition with fictitious points

We consider the numerical boundary condition described in section 7.1.3 that involves the calculation of the fictitious points that we need to have when using a backward third order difference to calculate the approximate solution at the first interior points next to the physical boundaries. We have the physical boundary conditions (7.15) and (7.16) and therefore we also have the two equalities (7.18) and (7.19), that give us the value of the fictitious points that we need.

Applying the numerical boundary condition with the fictitious points we have a result that does not differ too much from the two previous numerical boundary conditions. This is not surprising since the results are similar to the von Neumann stability regions given in figure 7.5 which is a necessary condition for the stability of the Taylor Quickest scheme with any numerical boundary condition. Essentially the stability region is slightly larger than the previous ones for relatively big values of μ.

In figure 7.8a we show the sufficient condition $||B|| \leq 1$ for stability and in figure 7.8b we show the necessary condition $\rho(B) \leq 1$ that intersected with the von Neumann condition gives the stability regions for the Taylor Quickest scheme with this numerical boundary condition.

7.2.3 Summary

For the Quickest schemes with the numerical boundary conditions we suggest that it is quite safe in terms of stability to compute inside the region determined by the condition $\rho(B) \leq 1$ intersected with the von Neumann condition as had already happened for the one dimensional case. Although both conditions

are only necessary conditions for the stability of a finite difference scheme with numerical boundary conditions, the intersection of both conditions seems to give us a quite strong necessary condition if not one that is also sufficient. We also took into account the sufficient condition $||B|| \leq 1$, so that we would have some more additional information about the stability regions. We conclude that the numerical boundary condition involving the use of fictitious points gives us the largest stability region.

In the next section we consider a problem with non-uniform flow velocity. We apply the same kind of numerical boundary conditions that we have applied so far in this chapter, although with some minor changes.

7.3 A problem with non-uniform flow velocity

We consider a two dimensional unsteady convection-diffusion flow defined on the square $\{(x,y) \in \mathbb{R}^2 : (x,y) \in [0,1] \times [0,1]\}$ with non-uniform velocity, $(v(x,y), w(x,y))$,

$$\frac{\partial u}{\partial t} + v\frac{\partial u}{\partial x} + w\frac{\partial u}{\partial y} = D(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}) \tag{7.20}$$

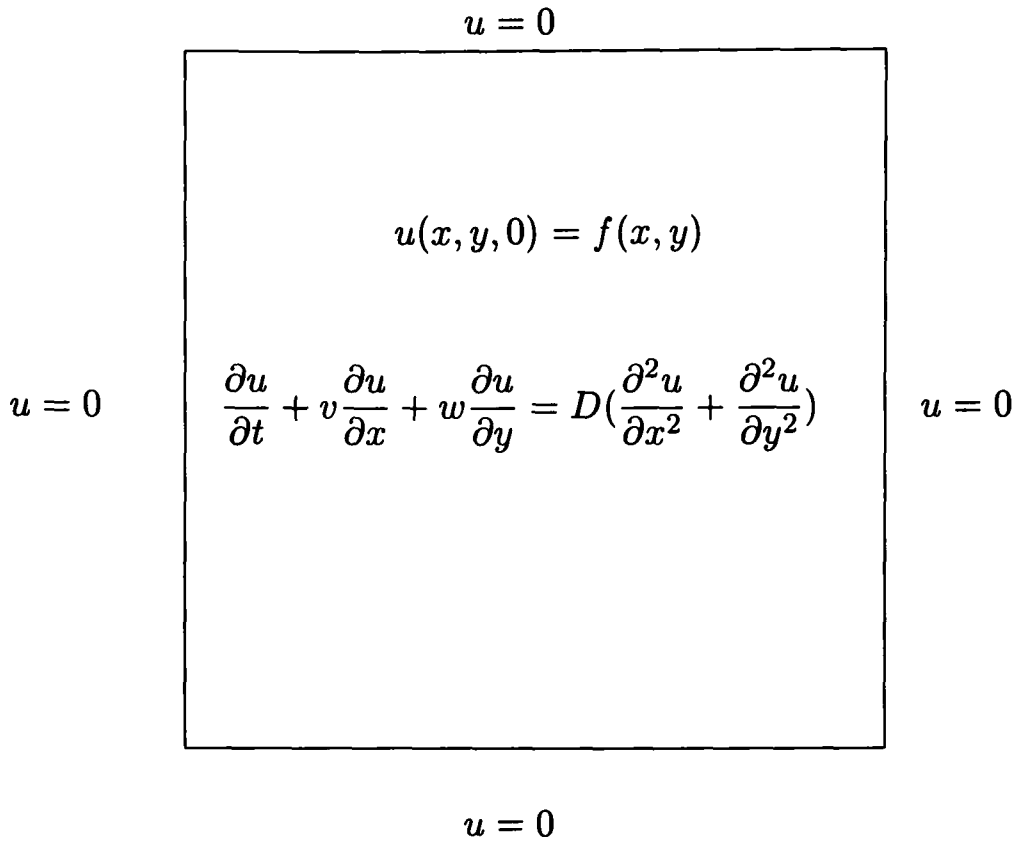
an initial condition given by

$$u(x,y,0) = f(x,y), \tag{7.21}$$

and the wall conditions,

$$u(0,y,t) = 0 \qquad u(1,y,t) = 0, \tag{7.22}$$

$$u(x,0,t) = 0 \qquad u(x,1,t) = 0. \tag{7.23}$$



We define the non-uniform velocity field by

$$v(x,y) = V\sin(p_1\pi x)\cos(p_2\pi y), \tag{7.24}$$

$$w(x,y) = -V\frac{p_1}{p_2}\cos(p_1\pi x)\sin(p_2\pi y), \tag{7.25}$$

where p_1 and p_2 can take the values one or two. The stream function associated with the velocity (7.24), (7.25) is given by:

$$\psi = \psi_0 \sin(p_1\pi x)\sin(p_2\pi y),$$

where

$$p_2\pi\psi_0 = V. \tag{7.26}$$

A variety of circular motions can be generated by these streamlines. The stream function is shown in figure 7.9 for different values of p_1 and p_2.

We assume $\Delta x = \Delta y$ and for $\nu = V\Delta t/\Delta x$ we have,

$$(\nu_x)_{jk} = \nu\sin(p_1\pi j\Delta x)\cos(p_2\pi k\Delta y) \tag{7.27}$$

$$(\nu_y)_{jk} = -\nu\frac{p_1}{p_2}\cos(p_1\pi j\Delta x)\sin(p_2\pi k\Delta y). \tag{7.28}$$

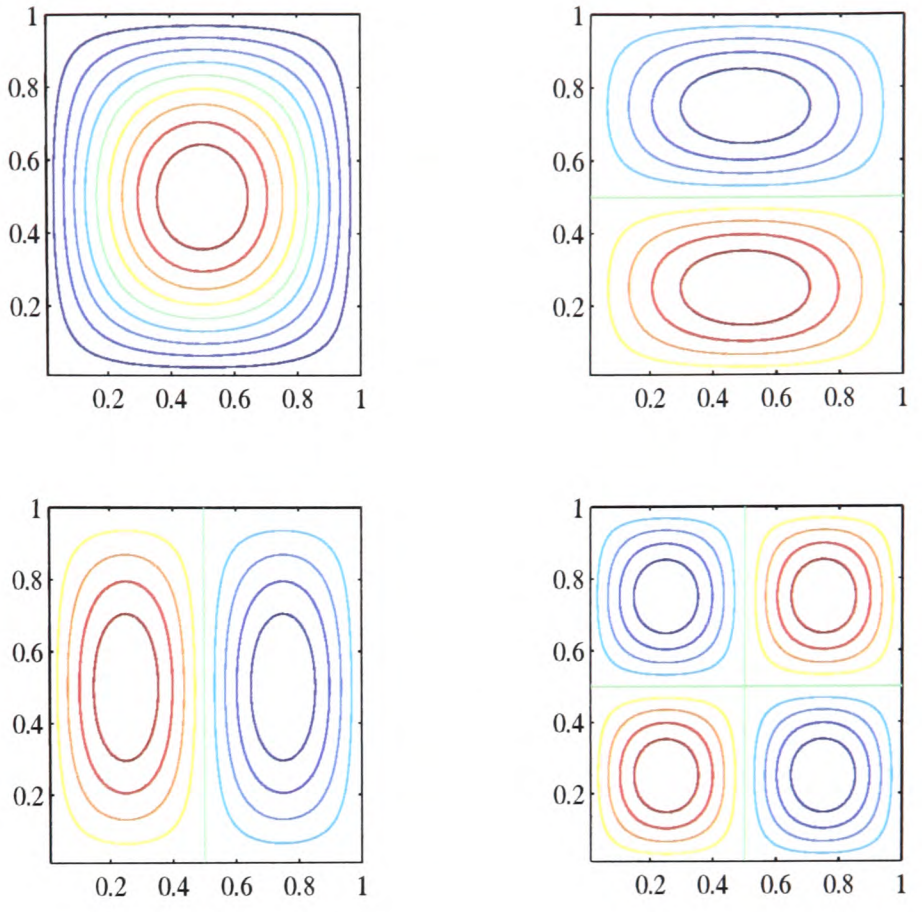


Figure 7.9: Stream function starting on the top left to the right: $(p_1, p_2) = (1, 1)$; $(p_1, p_2) = (1, 2)$; $(p_1, p_2) = (2, 1)$; $(p_1, p_2) = (2, 2)$.

We also consider $\mu = \mu_x = \mu_y$.

The discrete mesh points are given by

$$U_{jk}^n = u(x_j, y_k, t_n), \quad j, k = 0, \ldots, N+1; \; n = 0, 1, \ldots,$$

where $\Delta x = \Delta y = \dfrac{1}{N+1}$.

Associated with the physical boundary conditions (7.22) and (7.23), we consider the Dirichlet boundary conditions,

$$U_{0k}^{n+1} = 0, \quad U_{j0}^{n+1} = 0, \quad j, k = 0, \ldots, N+1, \quad n = 0, 1, \ldots.$$

$$U_{N+1k}^{n+1} = 0, \quad U_{jN+1}^{n+1} = 0, \quad j, k = 0, \ldots, N+1, \quad n = 0, 1, \ldots.$$

The Lax-Wendroff schemes and the Quickest schemes can be written as matrix equations

$$U^{n+1} = BU^n, \quad n = 0, 1 \ldots,$$

where $U^n = \{U^n_{11}, \ldots, U^n_{1N}, \ldots, U^n_{21}, \ldots, U^n_{2N}, \ldots, U^n_{N1}, \ldots, U^n_{NN}\}$, and B is an $N^2 \times N^2$ matrix and depends on the scheme used. The entries of the matrix B depend on ν_x and ν_y. This means that for a fixed μ, if $b_{\alpha\beta}$ denotes the entries of the matrix B, then the non-zero entries $b_{\alpha\beta}$ depend on $((\nu_x)_\alpha, (\nu_y)_\alpha)$. Since the velocity field is defined by (7.27) and (7.28) the study of stability can be carried out using the two parameters μ and ν.

In the next sections we apply the Lax-Wendroff schemes and the Quickest schemes to the flow with velocity (7.24) and (7.25) for $p_1 = p_2 = 1$. We investigate the stability of these schemes when applied to this problem. The stability analysis is mainly based on the observation of the spectrum of the matrix B for the scheme.

7.3.1 Lax-Wendroff schemes

We apply the Lax-Wendroff schemes to the flow with velocity given by (7.24) and (7.25) for $p_1 = p_2 = 1$ and which is represented in figure 7.9.

When applying the Polynomial Lax-Wendroff scheme, we need to take into consideration that this scheme changes according to the direction of the flow velocity, that is, when we interpolate the six needed points, although the five-point star is the same, the sixth point is chosen in agreement with the direction of the flow velocity. This particular fact makes the building of an algorithm that computes the approximate solution slightly more complex than if we could use the same finite difference scheme independently of the flow velocity. In the former case we need to test the direction of the flow velocity at each time step and then, according to the result, choose the correct finite difference formula.

The Taylor Lax-Wendroff scheme is used independently of the flow velocity and therefore its implementation is slightly simpler than for the polynomial Quickest scheme as explained above.

We plot the stability regions for both schemes in figure 7.10, given by the conditions $||B|| \leq 1$ and $\rho(B) \leq 1$. It seems that although for the problem with constant velocity the Polynomial Lax-Wendroff scheme offers a larger stability region, this fact is not so evident in the non-uniform case. We can observe that for μ close to 0.25 there is a larger range of ν values inside the stable region. The maximum value that μ takes in a stable region for the case with constant velocity is for both schemes 0.25 and this is still the case for the non-uniform velocity.

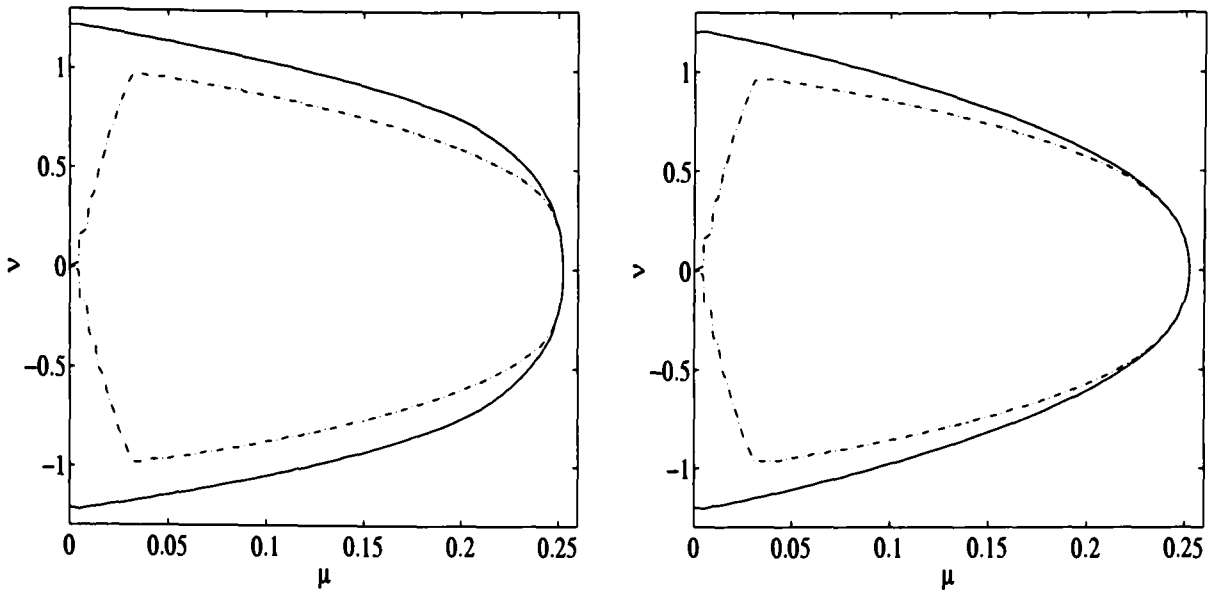


Figure 7.10: Matrices size $15^2 \times 15^2$; $\rho(B) \leq 1(-), ||B|| \leq 1(- \cdot -)$: (a) Polynomial Lax-Wendroff (b) Taylor Lax-Wendroff.

7.3.2 Quickest schemes

To apply any of the Quickest schemes to a flow with velocities that change directions and subject to various physical boundary conditions, requires careful work.

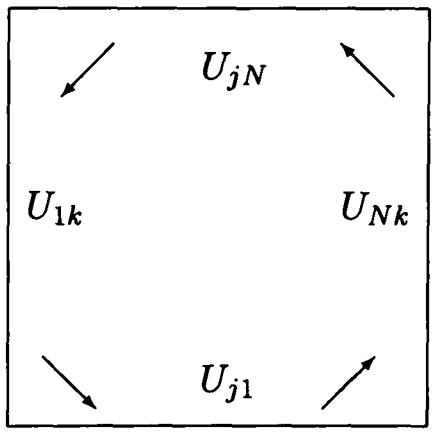


Figure 7.11: Arrows show the direction of the flow velocity and the first interior mesh points next to the walls are indicated.

There are two main facts that contribute to the difficulty. First we need to change the discrete points that we use at each time step, according to the direction of the flow velocity and, secondly and more importantly, we need to

have numerical boundary conditions which also depend on the direction of the flow velocity next to the walls.

In figure 7.11 we plot the direction of the flow inside the domain that we are considering and we denote the approximate solution at the first interior mesh points, next to the walls, that are calculated as numerical boundary conditions. Note that we are considering the flow with velocities (7.24) and (7.25) for $p_1 = p_2 = 1$ as represented in figure 7.9.

Taylor Quickest scheme - Downwind numerical boundary condition

First we describe the numerical boundary conditions applied to the Taylor Quickest scheme. When applying the downwind numerical boundary condition, as already mentioned, we need to take into account the signs of the velocity components when discretising the values next to the walls.

The numerical boundary conditions that we use to calculate the approximated solution values $\{U_{j1}, j = 2, \ldots, N-1\}$, consist of applying a forward third-order difference in the y direction at all the points of the set independently of the sign of w and a backward or forward third order difference in the x direction according to the sign of v. For the set of values $\{U_{jN}, j = 2, \ldots, N-1\}$, we apply a backward third order difference in the y direction at all the points independently of the sign of w and a backward or forward third order difference in the x direction according to the sign of v. Similarly for the set of values $\{U_{1k}, k = 2, \ldots, N-1\}$ we apply a forward third order difference in the x direction at all the points independently of the sign of v, and for the set $\{U_{Nk}, k = 2, \ldots, N-1\}$ we apply a backward third order difference in the x direction at all the points independently of the sign of v, and for both sets we apply backward or forward third order difference in the y direction according to the sign of x.

To calculate the approximate solution values at the corners of the domain, namely U_{11}, U_{1N}, U_{N1} and U_{NN}, we use respectively a forward third-order difference in the x direction and y direction; a forward third-order difference in the x direction and a backward third-order difference in the y direction; a backward third-order difference in the x direction and a forward third-order difference in the y direction and a backward third-order difference in the x direction and y direction.

We show the stability region of the Taylor Quickest scheme with the downwind numerical boundary condition in figure 7.12.

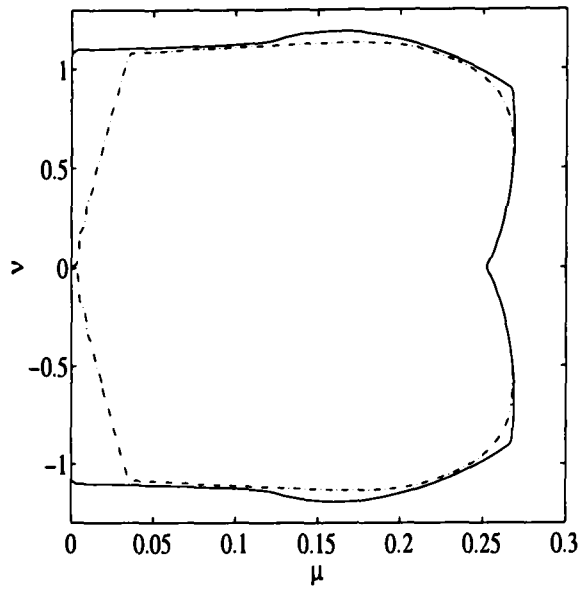


Figure 7.12: Matrix size $15^2 \times 15^2$: Taylor Quickest with downwind numerical boundary condition: $\rho(B) \leq 1(-)$, $||B|| \leq 1(- \cdot -)$

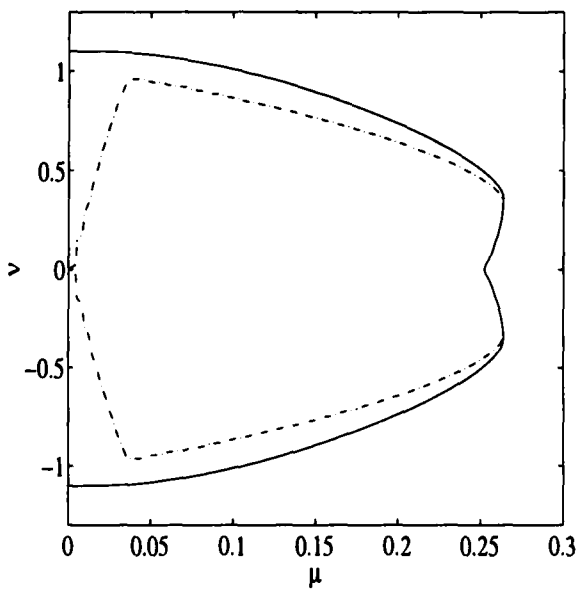


Figure 7.13: Matrix size $15^2 \times 15^2$: Taylor Quickest with Lax-Wendroff numerical boundary condition:$\rho(B) \leq 1(-)$, $||B|| \leq 1(- \cdot -)$:

Taylor Quickest scheme - Lax-Wendroff numerical boundary condition

The Lax-Wendroff numerical boundary condition consists of applying the Taylor Lax-Wendroff at all the discrete points next to the walls, namely $U_{1k}, k = 1,\ldots,N$, $U_{j1}, j = 1,\ldots,N$, $U_{Nk}, k = 1,\ldots,N$ and $U_{jN}, j = 1,\ldots,N$. The stability region is displayed in figure 7.13 and is a smaller region than the stability region for the scheme with the downwind numerical boundary condition.

Taylor Quickest scheme - Fictitious points numerical boundary condition

This numerical boundary condition consists of using fictitious points. We calculate the fictitious points by applying a Lax-Wendroff scheme that is obtained by performing an interpolation of six discrete points around the discrete points on the boundary. These interpolation points are chosen according to the direction of the velocities of the approximate solution next to the walls. The boundary values where the Lax-Wendroff scheme is applied are U^n_{j0}, U^n_{0k}, U^n_{N+1k} and U^n_{jN+1}. We explain this procedure in more detail in what follows. We define the following operator $\mathcal{P}$,

$$\mathcal{P}U^n_{j,k} = [1 - \nu_x\Delta_{x0} - \nu_y\Delta_{y0} + (\frac{1}{2}\nu_x^2 + \mu)\delta_x^2 + (\frac{1}{2}\nu_y^2 + \mu)\delta_y^2]U^n_{j,k}.$$

Taking into account the signs of the velocities next to the walls, that is, at the first interior mesh points, we obtain that for the points $U^n_{j,0}$ then, if w is positive,

$$U^{n+1}_{j,0} = \mathcal{P}U^n_{j,0} + \nu_x\nu_y\Delta_{x-}\Delta_{y+}U^n_{j,0}, \quad \text{if } v \text{ is positive,}$$

$$U^{n+1}_{j,0} = \mathcal{P}U^n_{j,0} + \nu_x\nu_y\Delta_{x+}\Delta_{y+}U^n_{j,0}, \quad \text{if } v \text{ is negative.}$$

For the points $U^n_{0,k}$ if v is positive then

$$U^{n+1}_{0,k} = \mathcal{P}U^n_{0,k} + \nu_x\nu_y\Delta_{x+}\Delta_{y-}U^n_{0,k} \quad \text{if } w \text{ is positive,}$$

$$U^{n+1}_{0,k} = \mathcal{P}U^n_{0,k} + \nu_x\nu_y\Delta_{x+}\Delta_{y+}U^n_{0,k} \quad \text{if } w \text{ is negative.}$$

For the points $U^n_{N+1,k}$ if v is negative then

$$U^{n+1}_{N+1,k} = \mathcal{P}U^n_{N+1,k} + \nu_x\nu_y\Delta_{x-}\Delta_{y+}U^n_{N+1,k}, \quad \text{if } w \text{ is negative,}$$

$$U_{N+1,k}^{n+1} = \mathcal{P}U_{N+1,k}^{n} + \nu_x\nu_y\Delta_{x-}\Delta_{y-}U_{N+1,k}^{n}, \quad \text{if } w \text{ is positive.}$$

For the points $U_{j,N+1}^{n}$ if w is negative then

$$U_{j,N+1}^{n+1} = \mathcal{P}U_{j,N+1}^{n} + \nu_x\nu_y\Delta_{x-}\Delta_{y+}U_{j,N+1}^{n}, \quad \text{if } v \text{ is positive,}$$

$$U_{j,N+1}^{n+1} = \mathcal{P}U_{j,N+1}^{n} + \nu_x\nu_y\Delta_{x+}\Delta_{y+}U_{j,N+1}^{n}, \quad \text{if } v \text{ is negative.}$$

Now, considering the fact that we have Dirichlet boundary conditions on the walls, after some algebra we obtain,

$$\begin{cases} 0 = & \frac{1}{2}(\nu_y + \nu_y^2 + 2\mu)U_{j,-1}^{n} + \frac{1}{2}(-\nu_y + \nu_y^2 + 2\mu + 2\nu_x\nu_y)U_{j,1}^{n} \\ & -\nu_x\nu_y U_{j-1,1}^{n} \\ 0 = & \frac{1}{2}(\nu_y + \nu_y^2 + 2\mu)U_{j,-1}^{n} + \frac{1}{2}(-\nu_y + \nu_y^2 + 2\mu - 2\nu_x\nu_y)U_{j,1}^{n} \\ & +\nu_x\nu_y U_{j+1,1}^{n} \end{cases} \tag{7.29}$$

$$\begin{cases} 0 = & \frac{1}{2}(\nu_x + \nu_x^2 + 2\mu)U_{-1,k}^{n} + \frac{1}{2}(-\nu_x + \nu_x^2 + 2\mu - 2\nu_x\nu_y)U_{1,k}^{n} \\ & +\nu_x\nu_y U_{1,k+1}^{n} \\ 0 = & \frac{1}{2}(\nu_x + \nu_x^2 + 2\mu)U_{-1,k}^{n} + \frac{1}{2}(-\nu_x + \nu_x^2 + 2\mu - 2\nu_x\nu_y)U_{1,k}^{n} \\ & -\nu_x\nu_y U_{1,k-1}^{n} \end{cases} \tag{7.30}$$

$$\begin{cases} 0 = & \frac{1}{2}(-\nu_x + \nu_x^2 + 2\mu)U_{N+2,k}^{n} + \frac{1}{2}(\nu_x + \nu_x^2 + 2\mu - 2\nu_x\nu_y)U_{N,k}^{n} \\ & -\nu_x\nu_y U_{N,k-1}^{n} \\ 0 = & \frac{1}{2}(-\nu_x + \nu_x^2 + 2\mu)U_{N+2,k}^{n} + \frac{1}{2}(\nu_x + \nu_x^2 + 2\mu + 2\nu_x\nu_y)U_{N,k}^{n} \\ & +\nu_x\nu_y U_{N,k+1}^{n} \end{cases} \tag{7.31}$$

$$\begin{cases} 0 = & \frac{1}{2}(-\nu_y + \nu_y^2 + 2\mu)U_{j,N+2}^{n} + \frac{1}{2}(\nu_y + \nu_y^2 + 2\mu + 2\nu_x\nu_y)U_{j,N}^{n} \\ & -\nu_x\nu_y U_{j+1,N}^{n} \\ 0 = & \frac{1}{2}(-\nu_y + \nu_y^2 + 2\mu)U_{j,N+2}^{n} + \frac{1}{2}(\nu_y + \nu_y^2 + 2\mu - 2\nu_x\nu_y)U_{j,N}^{n} \\ & +\nu_x\nu_y U_{j-1,N}^{n} \end{cases} \tag{7.32}$$

From the previous equations (7.29)–(7.32) we calculate the values of the fictitious points and use those values in the general formula of the Taylor Quickest scheme next to the walls where the fictitious points are needed.

The stability region for the Taylor Quickest scheme with the numerical boundary condition involving the fictitious points is displayed in figure 7.14.

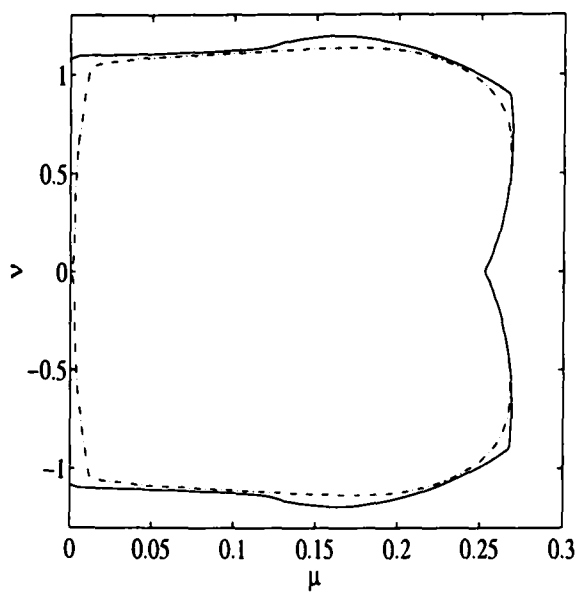


Figure 7.14: Matrix size $15^2 \times 15^2$: Taylor Quickest with fictitious point numerical boundary condition: $\rho(B) \leq 1(-)$, $||B|| \leq 1(- \cdot -)$.

This stability region is very similar to the stability region given by this scheme with the downwind numerical boundary condition (compare with figure 7.12).

Now we turn to the Polynomial Quickest scheme when numerical boundary conditions are applied. We give the same name to the numerical boundary conditions but it is important to note that they are different when applied to the different Quickest schemes. We use the same name since it is associated with the framework we use to deduce the numerical boundary conditions. The different Taylor Quickest schemes that we obtain by assuming different directions of the flow are only different in terms of the approximation of the third derivative. The same does not happen with the Polynomial Quickest scheme which also changes according to the way that the mixed derivatives are approximated. These differences make the numerical boundary conditions that we use with this scheme slightly different from the numerical boundary conditions that we had used with the Taylor Quickest scheme.

Polynomial Quickest scheme - Downwind numerical boundary condition

We consider the Polynomial Quickest scheme with the downwind numerical boundary condition. This numerical boundary condition consists of approximating the third derivative with a forward third-order difference instead of a backward third-order difference or vice-versa, as required. The stability region

is shown in figure 7.15. This stability region is larger than the stability region for the Taylor Quickest scheme with the similar numerical boundary condition.

Polynomial Quickest scheme - Lax-Wendroff numerical boundary condition

We consider the Lax-Wendroff numerical boundary condition and in this case we consider none of the previously described Lax-Wendroff schemes but the one that results naturally from the Polynomial Quickest scheme, that is, by considering the derivatives of order three equal to zero. By doing this we use two different Lax-Wendroff schemes according to the direction of the velocity components. When w and v have the same sign we have,

$$\begin{aligned} U_{jk}^{n+1} &= U_{jk}^n - \nu_x \Delta_{x0} U_{jk}^n - \nu_y \Delta_{y0} U_{jk}^n \\ &\quad + (\frac{1}{2}\nu_x^2 + \mu_x)\delta_x^2 U_{jk}^n + (\frac{1}{2}\nu_y^2 + \mu_y)\delta_y^2 U_{jk}^n \\ &\quad + \frac{1}{2}\nu_x \nu_y \Delta_{y+}\Delta_{x-} U_{jk}^n + \frac{1}{2}\nu_x \nu_y \Delta_{x+}\Delta_{y-} U_{jk}^n. \end{aligned} \tag{7.33}$$

When v and w have opposite signs we have,

$$\begin{aligned} U_{jk}^{n+1} &= U_{jk}^n - \nu_x \Delta_{x0} U_{jk}^n - \nu_y \Delta_{y0} U_{jk}^n \\ &\quad + (\frac{1}{2}\nu_x^2 + \mu_x)\delta_x^2 U_{jk}^n + (\frac{1}{2}\nu_y^2 + \mu_y)\delta_y^2 U_{jk}^n \\ &\quad + \frac{1}{2}\nu_x \nu_y \Delta_{y+}\Delta_{x+} U_{jk}^n + \frac{1}{2}\nu_x \nu_y \Delta_{x-}\Delta_{y-} U_{jk}^n. \end{aligned} \tag{7.34}$$

The stability region for this numerical boundary condition is shown in figure 7.16. This region is significantly smaller than the stability region computed for the previous numerical boundary condition, although slightly larger than the stability region for the Taylor Quickest scheme with the Lax-Wendroff numerical boundary condition.

Polynomial Quickest scheme - Fictitious points numerical boundary condition

The numerical boundary condition consisting of the calculation of fictitious points is obtained in a similar way as that used for the Taylor Quickest scheme.We compute the fictitious points using the equalities (7.29)–(7.32) and use these values when needed in the general formula of the Polynomial Quickest scheme determined according to the direction of the flow velocity, at the points next to the physical boundaries.

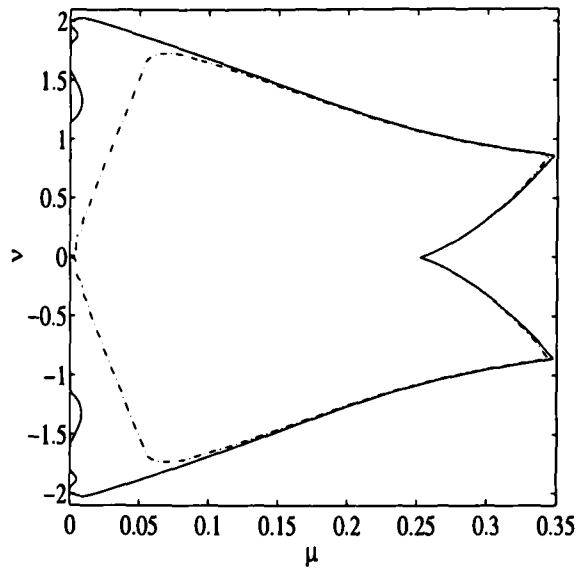


Figure 7.15: Matrix size $15^2 \times 15^2$: Polynomial Quickest with downwind numerical boundary condition: $\rho(B) \leq 1(-)$, $||B|| \leq 1(- \cdot -)$.

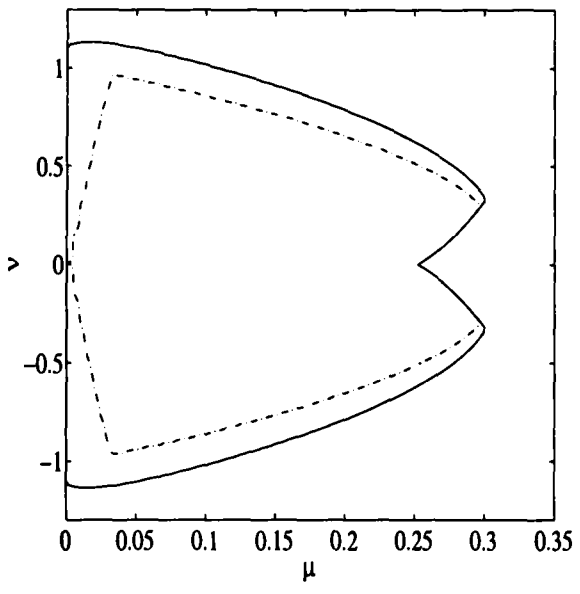


Figure 7.16: Matrix size $15^2 \times 15^2$: Polynomial Quickest with Lax-Wendroff numerical boundary condition: $\rho(B) \leq 1(-)$, $||B|| \leq 1(- \cdot -)$.

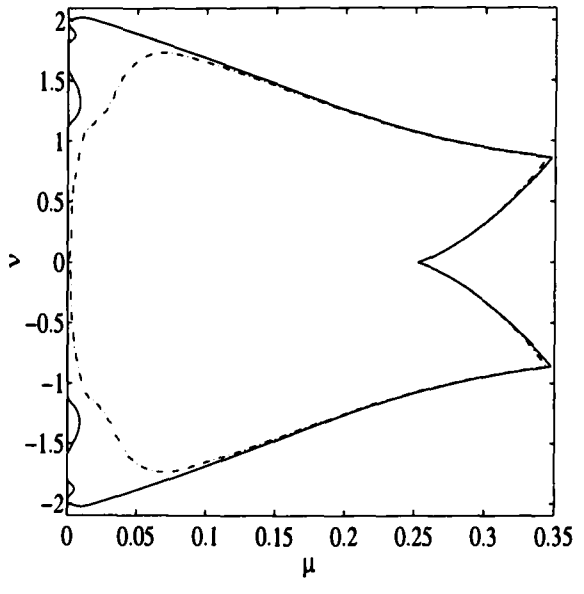


Figure 7.17: Matrix size $15^2 \times 15^2$: Polynomial Quickest with fictitious point numerical boundary condition: $\rho(B) \leq 1(-)$, $\|B\| \leq 1(- \cdot -)$.

The stability region for the Polynomial Quickest scheme with the numerical boundary condition involving the fictitious points is displayed in figure 7.17.

7.3.3 Summary

We have observed that, for this problem with non-uniform velocity, the largest stability region for both Quickest schemes is given by numerical boundary conditions with the fictitious points and also by using downwind numerical boundary conditions. In the case of the two Lax-Wendroff schemes, there was only a slight difference between their stability regions, the larger being that of the Polynomial Lax-Wendroff scheme.

In the next chapter we turn to a particular Navier-Stokes problem to which we apply only the Taylor Lax-Wendroff scheme and the Taylor Quickest scheme. In the latter case we also take into account the different numerical boundary conditions.

Chapter 8

The Navier-Stokes equations

So far we have considered the discretisation and associated stability questions for a convection-diffusion equation but equally important is that a convection-diffusion equation lies at the centre of fluid mechanics and, in particular, the Navier-Stokes equations. These equations have been regarded with scientific fascination for the wide variety of physical phenomena that come within their governance.

Unfortunately, from the point of view of computational fluid dynamics, most flow situations occurring in nature and in technology enter into a particular form of instability, called turbulence. This occurs when the velocity, or more precisely, the Reynolds number defined as the product of representative scales of velocity and length divided by the kinematic viscosity is sufficient large. Although there has been considerable progress in the development of computational techniques, the accurate numerical simulation of high Reynolds number flows remains a difficult and costly exercise. However, there are also a significant number of applications where the Reynolds number is sufficiently small for the flow to be laminar.

Many of the difficulties encountered in early Navier-Stokes calculations were inherent not only in the choice of the difference equations (accuracy), but also in the method of solution or choice of algorithm (convergence and stability), in the manner in which the dependent variables or discretised equations were related (coupling), in the manner that boundary conditions were applied, in the manner that the coordinate mesh was specified (grid generation), and finally, in recognising that for many high Reynolds number flows not all contributions to the Navier-Stokes equations were necessarily of equal importance. Many of the computational techniques and difficulties are widely reported, for instance, in

Hirsch [30], Fletcher [15], Peiret and Taylor [57] and Roache [65].

The finite difference method is a popular technique in fluid mechanics. Its application to the Navier-Stokes equations has been studied by many authors and there is a substantial literature on the subject.

In this chapter a numerical simulation of unsteady incompressible flow in two dimensions in a unit cavity is undertaken. We consider the formulation of the unsteady Navier-Stokes equations in terms of the vorticity and the streamline function and approximate the vorticity equation using the Taylor Lax-Wendroff scheme and the Taylor Quickest scheme. For the third-order Quickest scheme near the boundary we consider the numerical boundary conditions discussed in the previous chapter. We run experiments to illustrate the practical stability of the scheme for a Navier-Stokes problem and show how results from the simpler case presented in the previous chapter carry over to the more complex case.

We close this chapter by considering a global iteration matrix for the stream-function vorticity formulation for the Navier-Stokes equations, showing that the true time marching iteration matrix is more complicated than the matrix iteration for the convection-diffusion equation that is part of the Navier-Stokes equations. Indeed, examination of a one-dimensional test problem shows that the full system has much tighter stability constraints than would be predicted from the convection-diffusion equation alone.

8.1 Introduction

The equations of motion of an incompressible fluid are:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = \frac{1}{\rho} \nabla \sigma, \tag{8.1}$$

$$\nabla \cdot \mathbf{u} = 0. \tag{8.2}$$

Here σ is the stress tensor and $\mathbf{u} = (u, v)$. If the fluid is Newtonian, the stress tensor σ is given by

$$\sigma = -p\delta + \mu\gamma. \tag{8.3}$$

where δ is the Kronecker delta; p is the hydrodynamic pressure and μ is the dynamic viscosity (which will be assumed to be constant). In two dimensions the rate of strain tensor is defined by

$$\gamma = \begin{pmatrix} 2u_x & u_y + v_x \\ v_y + u_x & 2v_y \end{pmatrix}.$$

If (8.3) is substituted into (8.1) and if the continuity equation (8.2) is invoked, the Navier-Stokes equations are obtained:

$$\begin{cases} \dfrac{\partial u}{\partial t} + \mathbf{u}\cdot\nabla u = & -\dfrac{1}{\rho}p_x + \nu\left(\dfrac{\partial^2 u}{\partial x^2} + \dfrac{\partial^2 u}{\partial y^2}\right), \\ \dfrac{\partial v}{\partial t} + \mathbf{u}\cdot\nabla v = & -\dfrac{1}{\rho}p_y + \nu\left(\dfrac{\partial^2 v}{\partial x^2} + \dfrac{\partial^2 v}{\partial y^2}\right). \end{cases} \tag{8.4}$$

Here ν is the kinematic viscosity ($\nu = \mu/\rho$).

Stream-function vorticity

An alternative form of the equations (8.4) that has been applied extensively in two-dimensions is obtained using a vorticity, ω, and a stream function ψ formulation.

Let $\omega = v_x - u_y$ and consider ψ such that

$$u = \psi_y, \qquad v = -\psi_x.$$

This definition leads to the equation

$$\omega = -\nabla^2\psi.$$

To obtain the vorticity transport equation the momentum equations (8.4) are cross differentiated:

$$\frac{\partial \omega}{\partial t} + \mathbf{u}\cdot\nabla\omega = \nu\nabla^2\omega. \tag{8.5}$$

Significantly, the pressure no longer appears explicitly in the (ψ,ω)–system and the vorticity is determined by a convection-diffusion equation (but with non-constant velocity).

Non-Dimensionalisation

Let T be the time scale, L a length scale set by the system size and U the velocity scale set by the boundary conditions. Consider

$$\omega' = \frac{U\omega}{L}, \quad t' = \frac{t}{T}, \quad x' = \frac{x}{L}, \quad u' = \frac{u}{U},$$

and let $St = L/UT$ be the Strouhal number. So the equation (8.5) becomes in dimensionless variables (dropping the prime for variables):

$$St\frac{\partial \omega}{\partial t} + \mathbf{u}\cdot\nabla\omega = \frac{1}{\mathrm{Re}}\nabla^2\omega. \tag{8.6}$$

where $\mathrm{Re} = UL/\nu$ is a Reynolds number. If we consider the time scale $T = L/U$ we will have a Strouhal number equal to one.

8.2 Model problem : Driven cavity

Our aim is to study the development of (ψ, ω) calculations through the example of the flow in a driven cavity. This problem has served as the prototype for the incompressible Navier-Stokes equations; see, Benjamim and Denny [6], Schreiber and Keller [69], Hou and Wetton [32] and Shen [68], only to mention a few. We will, however, find it useful to consider a new variant of the driven cavity problem.

The stability results for the new variant of the driven cavity problem are then compared with the results obtained by computing the traditional driven cavity problem.

8.2.1 Problem formulation

In two space dimensions our Navier-Stokes problem can be written in terms of the vorticity ω and the stream function ψ, as follows:

$$\omega_t + u\omega_x + v\omega_y = \frac{1}{\mathrm{Re}}\nabla^2\omega \tag{8.7}$$

$$\nabla^2\psi = -\omega \tag{8.8}$$

$$u = \psi_y \tag{8.9}$$

$$v = -\psi_x. \tag{8.10}$$

This set of equations is usually completed by specifying the velocity on the boundaries of the flow domain, either explicitly such as $\mathbf{u} = 0$ on a non-moving wall, or implicitly, for instance in a periodic channel of length l in the x direction, $\mathbf{u}(x + l, y) = \mathbf{u}(x, y)$.

We consider the equations (8.7), (8.8), (8.9) and (8.10) in a cavity which is driven by having the wall vorticity specified, rather than a non-constant wall vorticity, see figure 8.1. The cavity problem we consider is different from the traditional problem and we are not aware that it has been used before.

The traditional problem assumes $u = 0$ and $v = 0$ on all the fixed walls and on the moving wall at $y = 1$ it assumes $u = 1$ and $v = 0$. These boundary conditions can be written in terms of the stream function as

$$\psi = 0$$

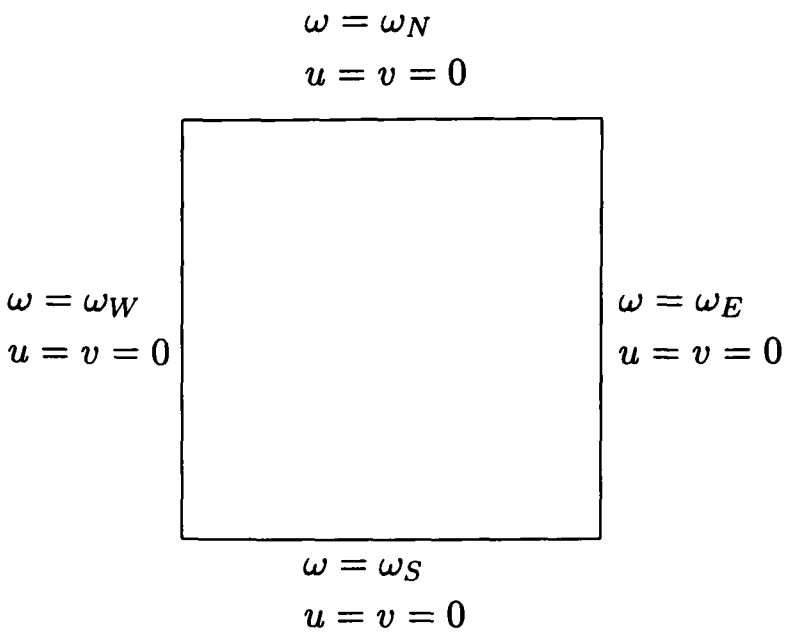


Figure 8.1: Driven Cavity with vorticity specified on boundaries

on all boundaries and

$$\frac{\partial \psi}{\partial n} = \epsilon$$

where $\epsilon = 0$ on fixed walls and $\epsilon = -1$ on the moving wall at $y = 1$. The coordinate n is normal to the surface. At first it seems it that we have too many boundary conditions for (8.8) and none for (8.7). However if (8.8) is substituted into (8.7), the resulting equation for the stream function has appropriate boundary conditions. Finding boundary conditions for the vorticity is a matter of converting one of the boundary conditions for the stream function to a boundary condition for the vorticity.

The problem that we consider is not the traditional driven cavity problem because we define the vorticity on the walls. We choose these unusual boundary conditions to allow us to concentrate on the problems associated with the convection-diffusion vorticity equation (8.7), isolated from the Poisson equation (8.8). In choosing this model problem we also disregard the numerical complications that can be associated with the way we choose the vorticity boundary conditions. A substancial literature is dedicated to this difficulty and a survey of the different vorticity conditions that can be used is given by Napolitano *et al* [52].

Although this problem is not physical, in the sense that there is no physical mechanism to generate a specific vorticity on the wall, it is expected that it preserves qualitatively the dynamical properties of the driven cavity flow. However since the velocity distribution along the walls is different than that of the driven cavity flow, it is clear that the effective Reynolds number will be different for

the two flows.

An $N \times N$ grid, with spacing

$$h = \Delta x = \Delta y = \frac{1}{N},$$

is laid on the driven cavity, and we then consider continuous time approximations $\tilde{\psi}_{ij}(t)$ to $\psi(ih, jh, t)$. The approximations $\tilde{\omega}_{ij}$, $\tilde{u}_{ij}$ and $\tilde{v}_{ij}$ are defined similarly.

In terms of the operators defined earlier we approximate the Navier-Stokes equations by:

$$\tilde{\omega}^{n+1} = \tilde{\omega}^n + F(\tilde{\omega}^n) \tag{8.11}$$

$$\nabla_h^2 \tilde{\psi}^{n+1} = -\tilde{\omega}^{n+1} \tag{8.12}$$

$$\tilde{u}^{n+1} = \Delta_{0y} \tilde{\psi}^{n+1} \tag{8.13}$$

$$\tilde{v}^{n+1} = -\Delta_{0x} \tilde{\psi}^{n+1} \tag{8.14}$$

where $F(\tilde{\omega}^n)$ is a finite difference approximation such as, a Lax-Wendroff scheme or a Quickest scheme to approximate the convection-diffusion operator in (8.7).

The above approximations can be implemented as follows: The solution starts with the establishment of initial values of ψ, ω, u and v everywhere at time $t = 0$. These initial conditions may correspond to some real initial situation for a transient problem of interest. Then the computational cycle begins as some finite difference equation (8.11) for the vorticity equation (8.7) is used to calculate an approximation to ω_t at all interior points in the computational field. The new values of ω are calculated at a new time level, increased by an increment Δt, by marching the vorticity transport equation forward in time. For example, (newω) = (oldω) + $\Delta t \omega_t$. The next step in the computational cycle is to solve the Poisson equation (8.8) using a multigrid method to get new values of the stream function ψ, using the new interior values of ω. At this point, new velocity components can be evaluated by finite difference equations (8.13) and (8.14) for the equations (8.9) and (8.10). Note that the ω values are given at the walls. Then the computational cycle is repeated until the desired time is reached.

8.2.2 The vorticity equation

The incompressible flow equations are more complicated than the vorticity equation alone, but the relation is close enough so that a study of the simpler vorticity equation is beneficial to the study of the incompressible equations. The vorticity transport equation, in either its non-conservation form or its conservation form,

is parabolic in time, contains two independent space coordinates, and is coupled to the elliptic Poisson stream-function equation through the nonlinear advective terms. But many aspects of the behaviour of this equation can be studied, and the essential features of many finite difference schemes can be illustrated.

We consider the model problem described in the previous section. The vorticity equation (8.7) is approximated by (8.11). For this specific problem, where the vorticity is explicitly given on the walls, the vector $\tilde{\omega}^n$ of interior solution values

$$\tilde{\omega}^n = (\tilde{\omega}^n_{1,1}, \ldots, \tilde{\omega}^n_{1,N-1}, \ldots, \tilde{\omega}^n_{N-1,1}, \ldots, \tilde{\omega}^n_{N-1,N-1})^T,$$

at the $(n+1)$th time level is related to the vector of solution values at the nth time level by the equation,

$$\tilde{\omega}^{n+1} = A(\tilde{u}^n, \tilde{v}^n)\tilde{\omega}^n + \mathbf{b}^n,$$

where $\mathbf{b}^n$ is a column vector of known boundary values and zeros and A is an $(N-1)^2 \times (N-1)^2$ matrix.

The finite difference schemes we consider in this chapter to approximate the vorticity equation are the Taylor Lax-Wendroff scheme and the Taylor Quickest scheme. The latter is associated with the numerical boundary conditions described in the previous chapter for a flow with non-uniform velocity. In the next section we present the numerical results when both methods are applied.

8.2.3 Numerical results and discussions

The numerical computations for the variant of the traditional driven cavity were performed for a mesh size 16×16. The initial conditions were taken to be zero for all the computations, unless otherwise specified. The vorticity values at the walls are

$$\omega_N = 5 \quad \text{and} \quad \omega_E = \omega_W = \omega_S = 0.5.$$

These vorticity values were chosen with arbitrary magnitude but so as to obtain a single vortex flow.

In the usual form of a driven cavity problem, energy is provided to the system through forces acting on the moving wall and this energy is dissipated by viscous action, becoming heat which will be lost through the cavity walls. In the modified cavity problem, energy is input through application of a torque to the fluid near the boundaries and that energy too is dissipated by viscous action in the interior of the cavity.

We consider the following finite difference approximations to approximate the vorticity equation: the Taylor Lax-Wendroff scheme and the Taylor Quickest scheme with three different numerical boundary conditions, the downwind third difference numerical boundary conditions, the Lax-Wendroff numerical boundary conditions and the numerical boundary conditions involving fictitious points. All these numerical boundary conditions were given in detail in section 7.3.

The main measures we use to determine whether we have convergence are the change in vorticity over one time step, referred to as the vorticity error, and the total kinetic energy.

Energy is a conserved quantity, that we compute in the inner points $(x_j, y_k) \in [0,1] \times [0,1]$:

$$||E|| = \left\{ \sum_{j=1}^{N-1} \sum_{k=1}^{N-1} (\tilde{u}^2(jh, kh) + \tilde{v}^2(jh, kh)) \right\}^{1/2}, \tag{8.15}$$

and the vorticity error is measured by:

$$||e\omega|| = \left\{ \sum_{j=1}^{N-1} \sum_{k=1}^{N-1} (\tilde{\omega}^{n+1}(jh, kh) - \tilde{\omega}^{n}(jh, kh))^2 \right\}^{1/2}. \tag{8.16}$$

In figure 8.2 we plot the vorticity error $||e\omega||$ as the step size diminishes and in figure 8.3 we plot the values of $||E||$ as time changes, for a Reynolds number $\mathrm{Re} = 10^3$. Observing figure 8.3 we can also see that the Taylor Lax-Wendroff scheme seems to be more dissipative.

To analyse the stability regions for the Navier-Stokes equations and to compare them with the stability regions that we computed for a convection-diffusion problem with non-uniform velocity in the previous chapter, we write the stream-function ψ as a Fourier series:

$$\psi(x, y) = \sum_r \sum_s \hat{\psi}_{rs} \sin(\pi r x) \sin(\pi s y), \tag{8.17}$$

where the Fourier modes $\hat{\psi}_{rs}$ are given by

$$\hat{\psi}_{rs} = 4 \int_0^1 \int_0^1 \psi(x, y) \sin(\pi r x) \sin(\pi s y) dx dy.$$

Since we are primarily investigating single vortex flows (see figure 8.9a) where the dominant mode is associated with $\hat{\psi}_{11}$, we approximate ψ by

$$\psi \approx \hat{\psi}_{11} \sin(\pi x) \sin(\pi y).$$

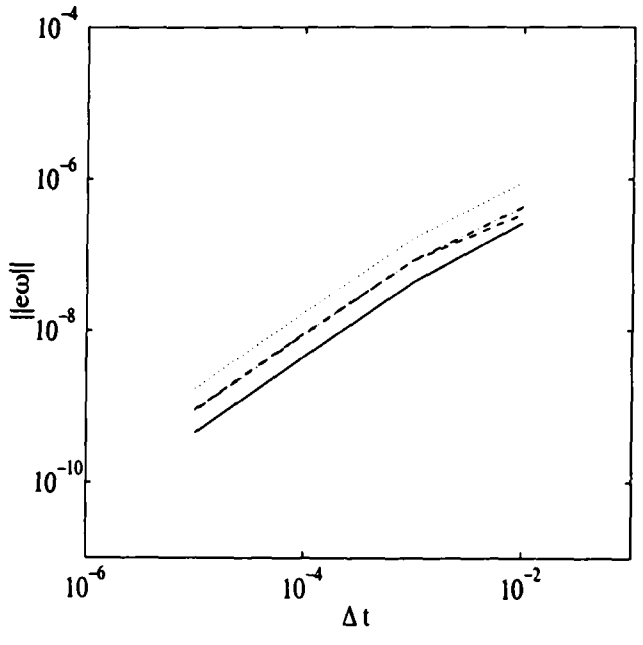


Figure 8.2: Vorticity error at the time $t = 400$ and Reynolds number $\mathrm{Re} = 10^3$ for Taylor Lax-Wendroff $(\cdots)$ and Taylor Quickest scheme with: Downwind numerical boundary $(- \cdot -)$; Lax-Wendroff numerical boundary $(--)$; Fictitious points $(-)$

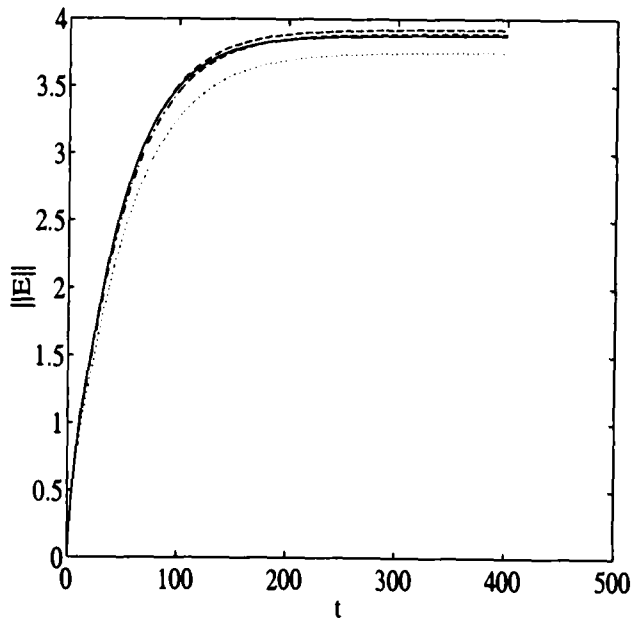


Figure 8.3: Energy until the time $t = 400$, Reynolds number $\mathrm{Re} = 10^3$ and time-step $\Delta t = 0.01$ for Taylor Lax-Wendroff $(\cdots)$ and Taylor Quickest scheme with: Downwind numerical boundary $(- \cdot -)$; Lax-Wendroff numerical boundary $(--)$; Fictitious points $(-)$

We denote

$$\nu = \hat{\psi}_{11}\pi\frac{\Delta t}{h} \quad \text{and} \quad \mu = \frac{1}{\text{Re}}\frac{\Delta t}{h^2}.$$

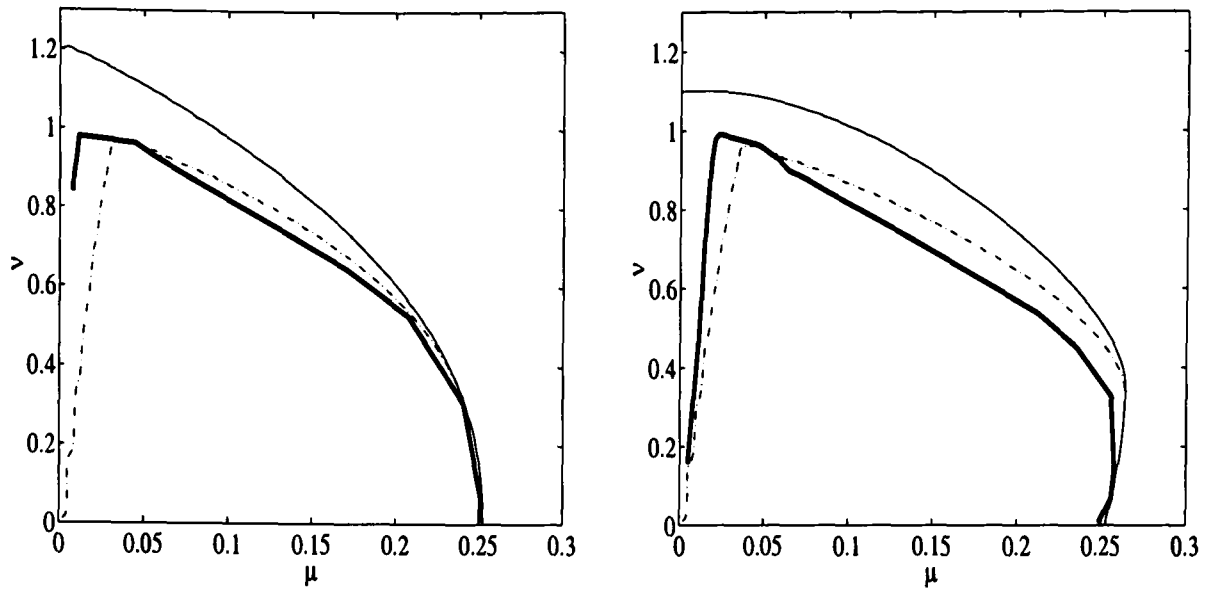
Note the similarity of the above definition of ν defined in the previous chapter were the equality (7.26) for $p_2 = 1$ gives,

$$\nu = \psi_0\pi\frac{\Delta t}{h}.$$

We plot the stability regions in the coordinates (μ, ν) in the figures 8.4 and 8.5. In these figures the curve in bold limits the practical stable region for the Navier-Stokes equations. The other curves are a repetition of the figures 7.10b, 7.13, 7.12 and 7.14 respectively. Those curves limit regions that have been determined by whether the the value of the norm and spectrum of the iterative matrix, of the respective scheme, are less than one. The practical stability regions for the Navier-Stokes equations are determined based on the behaviour of the vorticity error and the energy. If the energy is conserved and the vorticity error is converging to zero we assume we have practical stability.

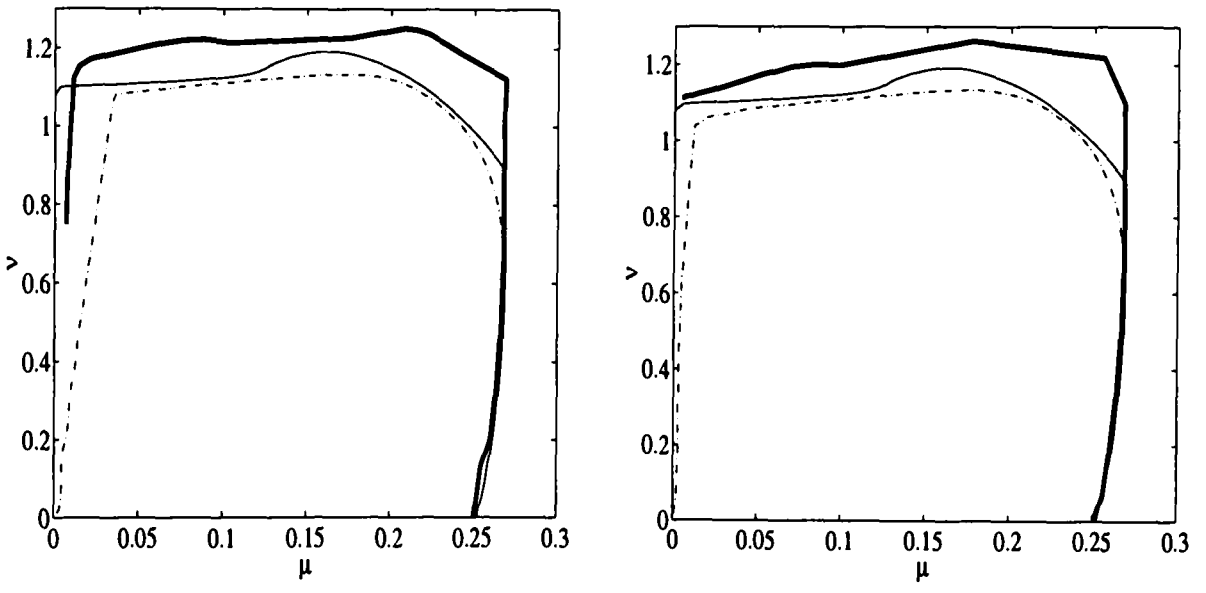
The figures 8.4a and 8.4b show that the Navier-Stokes equations stability regions are similar to the regions that were determined in section 7.3, for the Taylor Lax-Wendroff scheme and the Taylor Quickest scheme with Lax-Wendroff numerical boundary conditions, by the fact that the the norms of the iterative matrices are less than one respectively. On the other hand figures 8.5a and 8.5b show that the stability regions for the Navier-Stokes equations is close mostly to the regions determined by the value of the spectrum, for the Taylor Quickest scheme with the downwind numerical boundary condition and the numerical boundary condition involving fictitious points.

The computations were performed for a given Reynolds number Re and a fixed value of the mesh size h. We searched for the maximum value the time-step Δt could take, in order to have stability. In a linear problem, if Δt is chosen too large to satisfy the stability condition, the numerical results exhibit oscillations that grow very rapidly, and after few time steps their amplitude is infinite so that an overflow is registered by the computer. The phenomenon is characteristic of instability. In this case we have a non-linear equation, the vorticity equation, and the demarcation between stable and unstable calculations is not as sharp as for linear problems. The instability is more difficult to discover because sometimes no overflow appears and the amplitude of oscillations can remain bounded. None the less, we plot in figures 8.4 and 8.5, the results where it seems to be clear that we have stable solutions (not necessarily accurate).



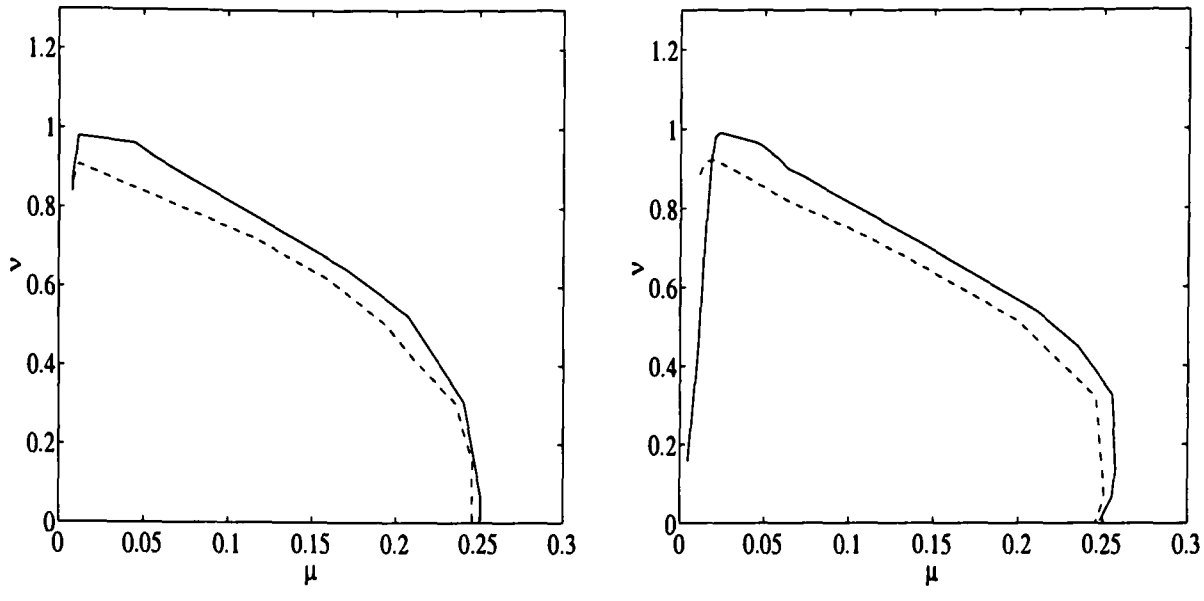
(a) (b)

Figure 8.4: (a) Taylor Lax-Wendroff scheme; (b) Taylor Quickest with Lax-Wendroff numerical boundary conditions. Line in bold limits the region where the Navier-Stokes problem is stable. The other regions are a repetition of figures 7.10b and 7.13 respectively.



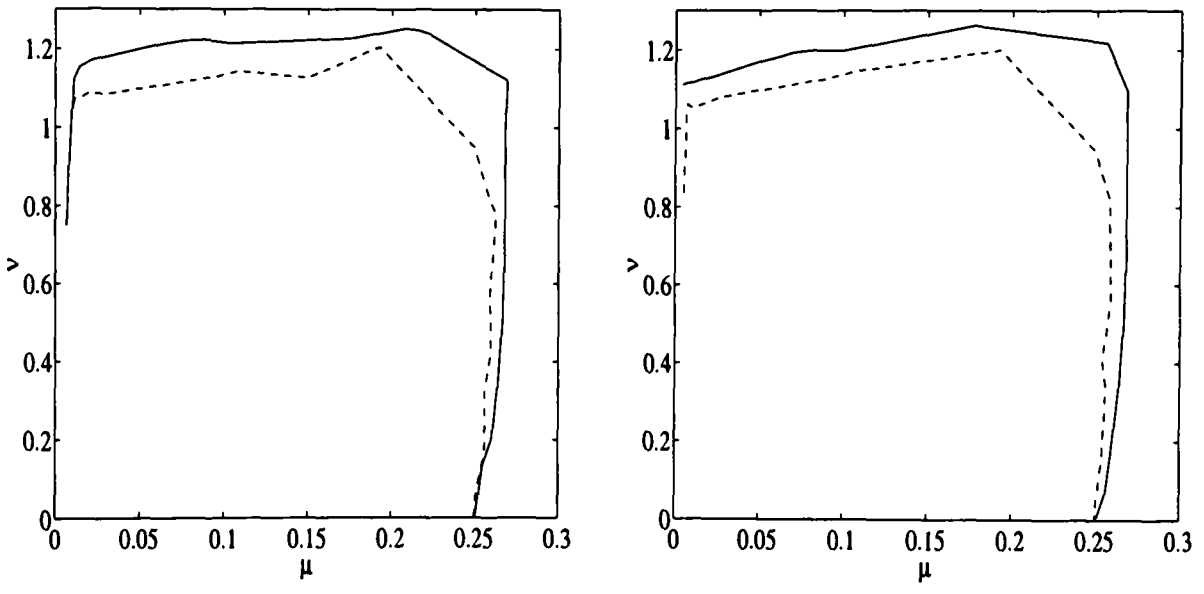
(a) (b)

Figure 8.5: (a) Taylor Quickest with downwind numerical boundary conditions; (b) Taylor Quickest with fictitious points as the numerical boundary condition. Line in bold limits the region where the Navier-Stokes problem is stable. The other regions are a repetition of figures 7.12b and 7.14 respectively.



(a) (b)

Figure 8.6: (a) Taylor Lax-Wendroff scheme; (b) Taylor Quickest with Lax-Wendroff numerical boundary conditions. Line (−−) denotes the practical stability region for the Navier-Stokes problem for a mesh size 16×16; Line (−) the same for a mesh size 32×32.

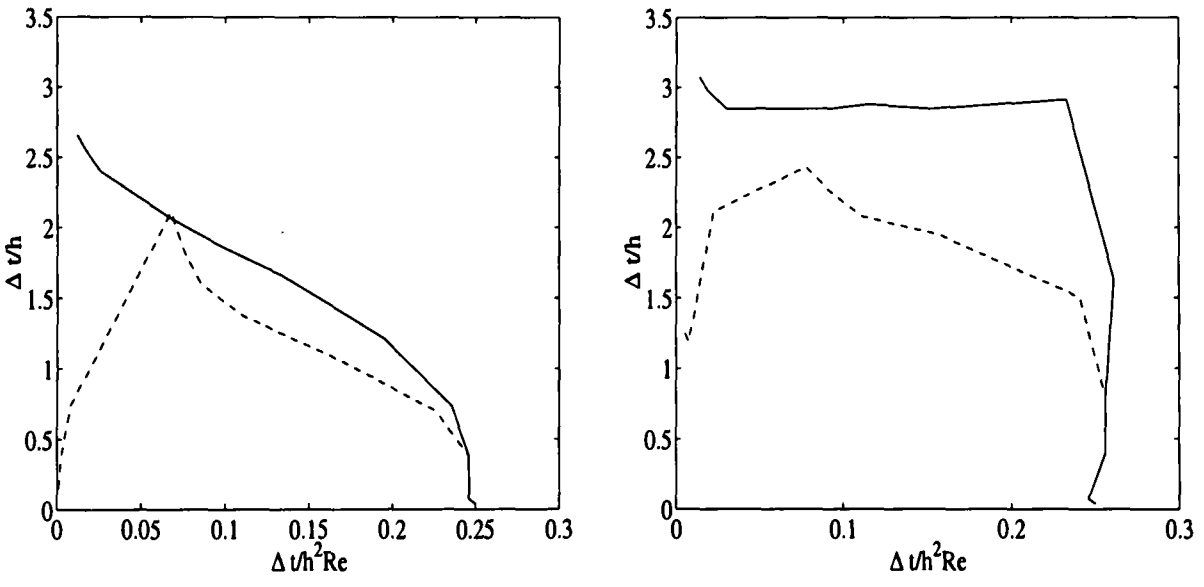


(a) (b)

Figure 8.7: (a) Taylor Quickest with downwind numerical boundary conditions; (b) Taylor Quickest with fictitious points as the numerical boundary condition. Line (−−) denotes the practical stability region for the Navier-Stokes problem for a mesh size 16×16; Line (−) the same for a mesh size 32×32.

When performing the computations the region that limits the maximum value of μ on the right side of figure 8.4 and figure 8.5 seems to almost coincide with the region given in chapter seven. Also the transition between the stable and the unstable region is quite clear, in the sense that the vorticity error or either converges to zero or blows up very fast. On the other hand considering in these figures the top line that determines the maximum value that ν can take in the stable regions, the transition from stable to unstable is not so clearly detected. We do not have a clear divergence immediately but we have a region were the vorticity error oscillates between some fixed values or is constant with a non-small value. The energy sometimes looks conserved other times also present oscillations between fixed values. As the solutions neither tend to a steady state nor to a suitable physical oscillation, we categorise this region as numerically unstable.

For the same schemes as above, in figures 8.6 and 8.7 we compare the stability regions obtained when considering a grid 32×32 with the grid 16×16. We obtain a smaller region for the former, but it does still not seem to be very significant difference from the stability regions plotted in the previous chapter and repeated in figures 8.4 and 8.5.



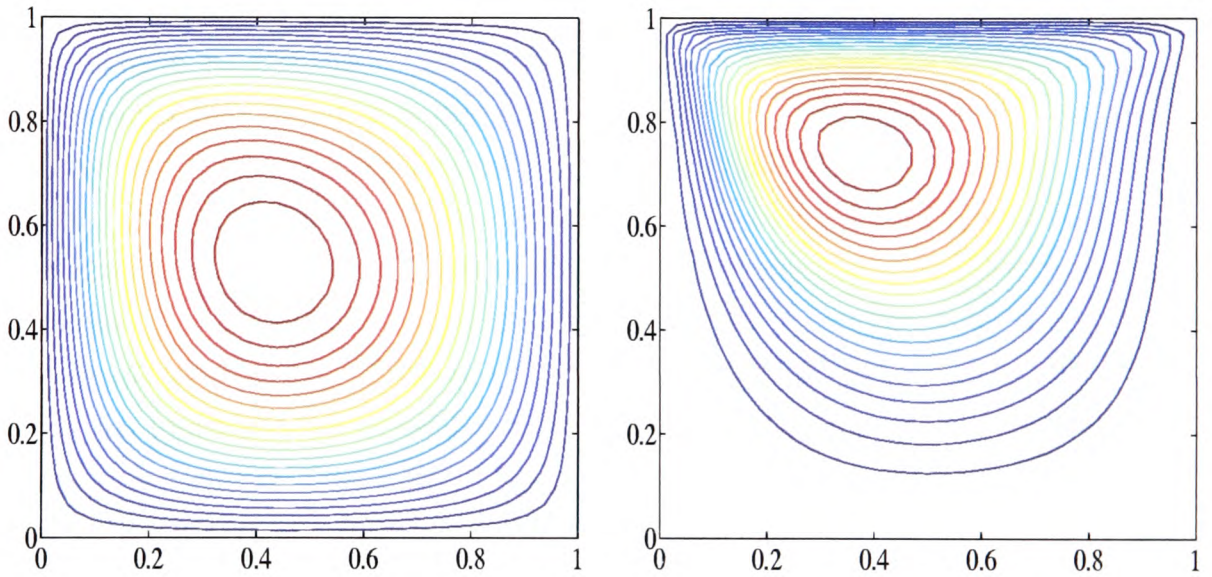
(a) (b)

Figure 8.8: Mesh size 32×32. Line $(--)$ denotes the practical stability region for the Navier-Stokes problem for the traditional driven cavity problem; Line $(-)$ denotes the practical stability region for the Navier-Stokes problem for the modified driven cavity problem. (a) Taylor Lax-Wendroff; (b) Taylor Quickest with downwind numerical boundary conditions.

Now we turn to the traditional cavity problem described earlier where the boundary vorticity is not given but determined by discretising equation (8.8) using Thom's vorticity boundary conditions [80]. We compare the practical stability regions of the scheme for the traditional driven cavity problem with the stability regions obtained for the modified cavity problem where the vorticity is given at the boundaries.

The computations were done for only two of the schemes: the Taylor Lax-Wendroff scheme and the Taylor Quickest scheme with downwind boundary conditions and for a grid 32×32, since for Reynolds number larger than 1000 we did not get satisfactory results when a grid 16×16 was considered. The regions obtained are plotted in figure 8.8a and 8.8b. The maximum value for $\Delta t/h$, considering the traditional driven cavity problem, in the figures 8.8a and 8.8b is around $\mathrm{Re} = 1000$. It is then around this value of the Reynolds number that we can take the biggest time-steps inside the stable region.

By observing the results in the figures 8.8a and 8.8b we can say that the modified cavity flow provides considerable insight into the stability regions for the traditional cavity flow.



(a) (b)

Figure 8.9: Streamline contours using a mesh size 32×32 for Reynolds number $\mathrm{Re} = 100$ (a) Modified Driven Cavity (b) Traditional Driven Cavity.

The computations whose results are shown in figures 8.8a and 8.8b were done for Reynolds number $\mathrm{Re} \leq 7000$, whereas the results of figures 8.4–8.7 were obtained for Reynolds number $\mathrm{Re} \leq 10000$. The reason is that the driven

cavity does not show stability for a Reynolds number larger than 7500, instead we get bounded oscillations even for very small time-steps. We did not analyse the nature of this phenomenon, although the reasons may be associated with the dynamical features of the physical problem as reported in the literature for the driven cavity in papers such that Bruneau and Jouron [7], Goodrich *et al* [25] and Shen [68], when other numerical schemes were used.

We plot the streamline contours for the modified cavity flow in figure 8.9a and for the traditional driven cavity in figure 8.9b.

In the next section we consider global iteration matrices for the stream-function vorticity formulation for the Navier-Stokes equations, showing that the true time marching iteration matrix is more complicated than the iterative matrix for the convection-diffusion equation that is part of the Navier-Stokes equations. We examine a one-dimensional test problem, which suggests that the full system should have tighter stability constraints than would be predicted from the convection-diffusion equation alone.

8.3 Global iteration matrices for the stream-function vorticity

We continue to consider the stream-function vorticity formulation for the two-dimensional Navier-Stokes equations, given by (8.7)–(8.10). To these equations one has to add boundary conditions on the velocity and possibly the vorticity.

Our main concern in this section is not so much the fine detail of how the equations are discretised but rather in formulating the overall iterative procedure in terms of a global iteration matrix, something which does not seem to have been done previously. This work is reported in Sousa and Sobey [77].

As an example we consider a relatively simple one-dimensional model problem which mimics the major features of the stream-function vorticity formulation. Although stability regions for multidimensional schemes can be more stringent than stability regions for their one-dimensional counterparts, our experience in examining the stability constraints of a convection-diffusion equation alone is that general guidelines that come from one-dimensional model problems are nevertheless useful when we turn to two-dimensional equations. So we think the results of this simple model problem are sufficiently interesting to be examined now although we intend in a further work to consider more complicated two dimensional problems using the same general formulation for the stability of the global iteration matrix.

8.3.1 Formulation

We suppose that (as is true for most two-dimensional stream-function vorticity calculations) the computational domain can be divided into two sets: points in the interior of the region where the vorticity is updated through the convection-diffusion equation, (8.7), and points on the boundary where the vorticity update occurs after the stream function has been updated to the new time level. The set of interior points is denoted by a subscript I while the boundary points are denoted by a subscript B, so that $\mathbf{W}_I^n$ and $\mathbf{\Psi}_I^n$ are vorticity and stream function values in the interior (which may of course, include inflow or outflow boundary points where the vorticity is still determined by convection-diffusion) at time $t = t_n$. The boundary sets are then $\mathbf{W}_B^n$ and $\mathbf{\Psi}_B^n$. Underlying the discussion here is an assumption of a uniform mesh of size h in both directions but it is not an essential assumption and the theory could be developed analogously for more general meshes. We shall also assume that the boundary values of the stream function vanish although that restriction can be lifted to add a little further algebraic complexity, as we intend to do in a future work.

Most time marching discretisations for the vorticity equation can be written

$$\mathbf{Q}\mathbf{W}_I^{n+1} = \mathbf{A}\mathbf{W}_I^n + \mathbf{B}\mathbf{W}_B^n, \tag{8.18}$$

for suitable matrices $\mathbf{Q}$, $\mathbf{A}$ and $\mathbf{B}$. This of course hides the non-linearity of the Navier-Stokes equations since the matrices $\mathbf{A}$ and $\mathbf{B}$ are both functions of the stream function but for the moment this can be kept in the background. The equation (8.18), also covers most implicit or explicit time marching schemes.

Next the stream function in the interior has to be determined from the updated vorticity in the interior. This is a linear equation, a discrete form of (8.8), which can be written

$$\frac{1}{h^2}\mathbf{L}\mathbf{\Psi}_I^{n+1} = -\mathbf{W}_I^{n+1}, \tag{8.19}$$

where $\mathbf{L}$ is a suitable matrix of discretisation of the Laplace operator. The values of the stream function on the boundaries will have been specified by the flux though the flow region.

The boundary vorticity values are then found from applying a discrete form of (8.8) at the boundaries using the updated values of the stream function (and possibly the interior vorticity),

$$\mathbf{W}_B^{n+1} = \frac{1}{h^2}\mathbf{M}\mathbf{\Psi}_I^{n+1} + \mathbf{P}\mathbf{W}_I^{n+1}, \tag{8.20}$$

where $\mathbf{M}$ and $\mathbf{P}$ are suitable matrices. The matrix $\mathbf{P}$ is zero in many formulations but for instance in Woods' method [95] the interior vorticity is used in updating the wall vorticity with second order accuracy.

This enables the stream function to be eliminated,

$$\mathbf{W}_B^{n+1} = (\mathbf{P} - \mathbf{M}\mathbf{L}^{-1})\mathbf{W}_I^{n+1}, \tag{8.21}$$

and then the update of the vorticity in the interior can be replaced so that

$$\mathbf{W}_B^{n+1} = (\mathbf{P} - \mathbf{M}\mathbf{L}^{-1})\mathbf{Q}^{-1}(\mathbf{A}\mathbf{W}_I^n + \mathbf{B}\mathbf{W}_B^n). \tag{8.22}$$

This essentially completes the derivation of the iteration matrix for this version of the Navier-Stokes equations, since we now have

$$\begin{bmatrix} \mathbf{W}_I^{n+1} \\ \mathbf{W}_B^{n+1} \end{bmatrix} = \begin{bmatrix} \mathbf{Q}^{-1}\mathbf{A} & \mathbf{Q}^{-1}\mathbf{B} \\ (\mathbf{P} - \mathbf{M}\mathbf{L}^{-1})\mathbf{Q}^{-1}\mathbf{A} & (\mathbf{P} - \mathbf{M}\mathbf{L}^{-1})\mathbf{Q}^{-1}\mathbf{B} \end{bmatrix} \begin{bmatrix} \mathbf{W}_I^n \\ \mathbf{W}_B^n \end{bmatrix}, \tag{8.23}$$

or

$$\mathbf{W}^{n+1} = \mathbf{K}\mathbf{W}^n, \tag{8.24}$$

where we use $\mathbf{K}$ to denote the overall iteration matrix.

The usual method to consider stability of a convection-diffusion equation is to examine the eigenvalues of the matrix $\mathbf{A}$. In the case of the Navier-Stokes equations that is not sufficient and it is the eigenvalues of a global iteration matrix such as $\mathbf{K}$ which must be considered.

8.3.2 One-dimensional model problem

The iteration matrix $\mathbf{K}$ is difficult to calculate for an arbitrary domain and associated mesh but for some easier examples all the matrix algebra, including matrix inversion is within the capability of MATLAB. It is also true that the separation of the domain into interior and exterior points introduces an ordering which is a little involved from an implementational viewpoint. In the case of one space dimension the ordering is fairly simple and quite fine discretisation is computationally feasible. Consequently we have examined the following model problem.

The vorticity transport is described by

$$\frac{\partial \omega}{\partial t} + u\frac{\partial \omega}{\partial x} = \frac{1}{\mathrm{Re}}\frac{\partial^2 \omega}{\partial x^2}, \quad 0 \le x \le 1, \tag{8.25}$$

where u may be either a positive constant or a variable function of x, and the vorticity is related to the function $\psi(x,t)$ by

$$\frac{\partial^2 \psi}{\partial x^2} = -\omega. \tag{8.26}$$

The boundary conditions on the function ψ are

$$\psi(0,t) = \psi(1,t) = 0. \tag{8.27}$$

Of course the long time solution to this problem will be $\psi = \omega = 0$ but that does not alter the utility of the model problem for studying the stability of the time marching iteration process.

Assume then that this system is discretised in space at points $x_j = jh$, $j = 0,\ldots,M$ with $h = 1/M$ and discretised in time with a time step Δt so that $t_n = n\Delta t$. The interior of the domain will be the points $j = 1,\ldots,M-1$ while the boundary points are at $j = 0$ and $j = M$. A simple explicit forward difference in time is used so that

$$\frac{\partial \omega}{\partial t} \approx \frac{1}{\Delta t}(W_j^{n+1} - W_j^n). \tag{8.28}$$

In this model problem the interior and boundary vectors are

$$\mathbf{W}_I^n = [W_1^n, W_2^n, \ldots, W_{M-1}^n]^{\mathrm{T}}, \quad \text{and} \quad \mathbf{W}_B^n = [W_0^n, W_M^n]^{\mathrm{T}}. \tag{8.29}$$

We consider a number of different space discretisation schemes, both for the convection-diffusion equation and for specifying the boundary vorticity. The first set use the simplest discretisation for the boundary vorticity, due to Thom [80],

$$W_0^{n+1} = -2\frac{\Psi_1^{n+1}}{h^2}, \quad W_M^{n+1} = -2\frac{\Psi_{M-1}^{n+1}}{h^2}, \tag{8.30}$$

so that the matrix $\mathbf{P}$ vanishes and the matrix $\mathbf{M}$ is simply

$$\mathbf{M} = \begin{bmatrix} -2 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & -2 \end{bmatrix}, \tag{8.31}$$

with $M-1$ columns. If the 'Poisson' equation, (8.26) is discretised using second order central differences, the square matrix $\mathbf{L}$ of size $M-1$ is also simple to write down and will not vary in any of the following,

$$\mathbf{L} = \begin{bmatrix} -2 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 1 & -2 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & -2 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 1 & -2 \end{bmatrix}. \tag{8.32}$$

The remaining matrices **A** and **B** depend on the discretisation method but once specified they can be calculated by MATLAB and the matrix **K** assembled and analysed. In the next few sections we use two numerical parameters, a Courant number,

$$\nu = \frac{u\Delta t}{h}, \tag{8.33}$$

and a parameter representing the effect of viscosity,

$$\mu = \frac{\Delta t}{\mathrm{Re}h^2}. \tag{8.34}$$

The calculations we present are, unless otherwise stated, for the case $M = 30$. Other experiments with larger values of M only change the results a little and not in any significant way.

Lax-Wendroff

If the convection-diffusion equation is discretised using the compact three point Lax-Wendroff scheme then the $(M-1)\times(M-1)$ matrix **A** and the $(M-1)\times 2$ matrix **B** are given by

$$\mathbf{A} = \begin{bmatrix} 1-2\alpha & \alpha-\nu & 0 & \cdots & 0 & 0 & 0 \\ \alpha+\nu & 1-2\alpha & \alpha-\nu & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \alpha+\nu & 1-2\alpha & \alpha-\nu \\ 0 & 0 & 0 & \cdots & 0 & \alpha+\nu & 1-2\alpha \end{bmatrix}, \tag{8.35}$$

where $\alpha = \mu + \frac{1}{2}\nu^2$, and

$$\mathbf{B} = \begin{bmatrix} \alpha+\nu & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & \alpha-\nu \end{bmatrix}. \tag{8.36}$$

The matrices **A** and **B** (and hence **K**) depend on the two numerical parameters μ and ν and in figure 8.10 we show the contour in the (μ,ν)-space where the spectral radius $\rho(\mathbf{A}) = 1$ and $\rho(\mathbf{K}) = \mathbf{1}$. It can be seen that the region of stability is reduced somewhat for the full system compared to the convection-diffusion equation alone.

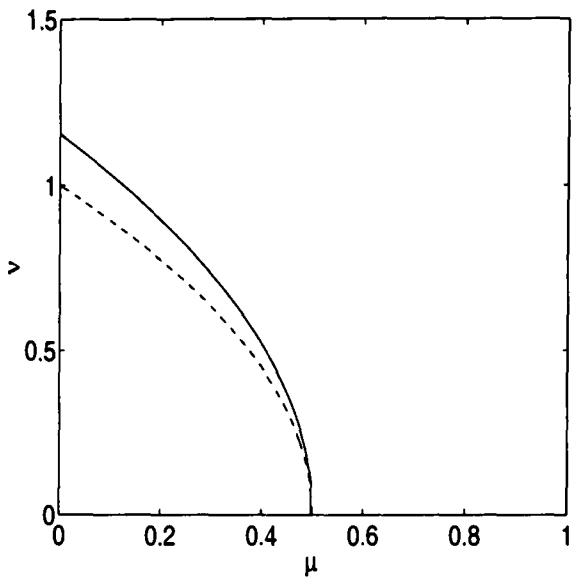


Figure 8.10: Contours of unit spectral radius for Lax-Wendroff method. (—) convection-diffusion operator, (- -) Navier-Stokes operator. The region of stability is the area between the curves and the axes.

Quickest with Lax-Wendroff at the first point

The application of a higher order method such as third-order Quickest requires two upstream points and of course this is not straightforward at the first interior point where there is only one upstream point. In the third chapter we have considered a number of options to get round this problem and the first scheme we examine here is just to use the three point Lax-Wendroff scheme at the first point and Quickest scheme at subsequent points. This modifies the **A** and **B** matrices a little. Quickest introduces a third-order difference with coefficient $\beta = \nu(1 - \nu^2 - 6\mu)/6$. If we keep the Lax-Wendroff scheme at the first point, and let

$$s_{ll} = -\beta,\ s_l = \alpha + \nu + 3\beta,\ s_c = 1 - 2\alpha - 3\beta,\ s_r = \alpha - \nu + \beta, \tag{8.37}$$

then

$$\mathbf{A} = \begin{bmatrix} 1-2\alpha & \alpha-\nu & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ s_l & s_c & s_r & 0 & \cdots & 0 & 0 & 0 & 0 \\ s_{ll} & s_l & s_c & s_r & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & & \ddots & & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & s_{ll} & s_l & s_c & s_r \\ 0 & 0 & 0 & 0 & \cdots & 0 & s_{ll} & s_l & s_c \end{bmatrix}, \tag{8.38}$$

and

$$\mathbf{B} = \begin{bmatrix} \alpha + \nu & 0 \\ s_{ll} & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & s_r \end{bmatrix}. \tag{8.39}$$

The stability boundaries for this scheme are shown in figure 8.11 and the region of stability for the Navier-Stokes type problem is considerably smaller than that for the convection-diffusion operator.

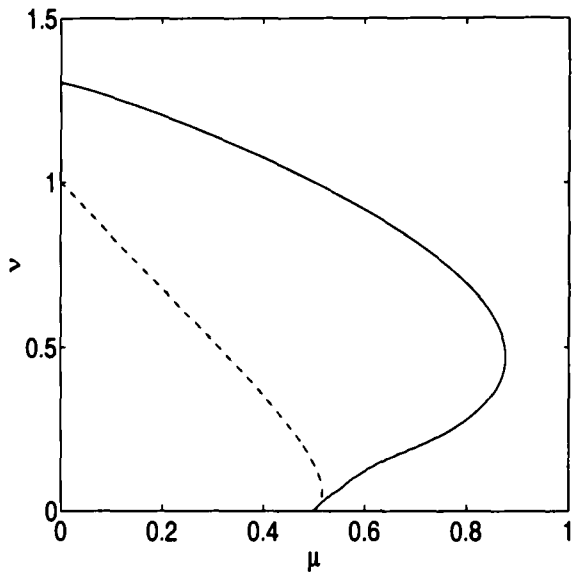


Figure 8.11: Contours of unit spectral radius for Quickest with Lax–Wendroff at first interior point. (—) convection-diffusion operator, (- -) Navier-Stokes operator. The region of stability is the area between the curves and the axes.

Quickest with downwinded at the first point

A second solution for the problem of what method to apply at the first point is to use a downwinded third difference. In principle this does not affect the order of accuracy of the solution. In this case the **A** and **B** matrices are given by

$$\mathbf{A} = \begin{bmatrix} 1-2\mu+3\beta & \alpha-\nu-3\beta & \beta & 0 & \cdots & 0 & 0 & 0 & 0 \\ s_l & s_c & s_r & 0 & \cdots & 0 & 0 & 0 & 0 \\ s_{ll} & s_l & s_c & s_r & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & & \ddots & & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & s_{ll} & s_l & s_c & s_r \\ 0 & 0 & 0 & 0 & \cdots & 0 & s_{ll} & s_l & s_c \end{bmatrix}, \tag{8.40}$$

and

$$\mathbf{B} = \begin{bmatrix} \alpha+\nu-\beta & 0 \\ s_{ll} & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & s_r \end{bmatrix} \tag{8.41}$$

The computed stability boundaries are shown in figure 8.12 and again the region of stability of the model problem is somewhat smaller than the stability region for the convection-diffusion operator alone.

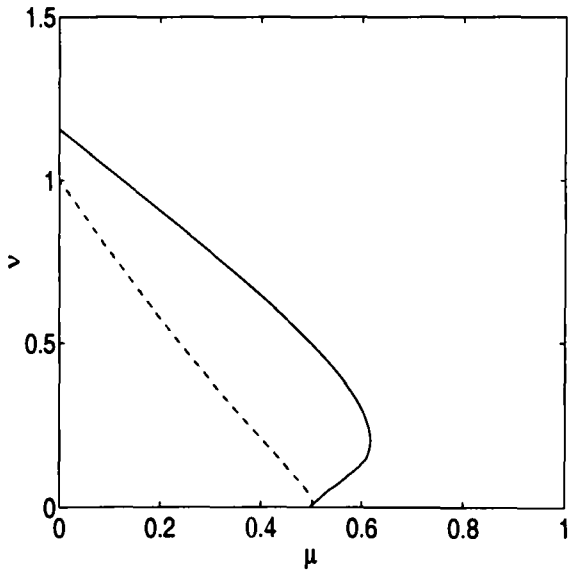


Figure 8.12: Contours of unit spectral radius for Quickest with downwinded third difference at first interior point. (—) convection-diffusion operator, (- -) Navier-Stokes operator. The region of stability is the area between the curves and the axes.

Quickest modified by Lax-Wendroff at inlet: fictitious point

The final Quickest type method we consider is the one whereby the Lax-Wendroff scheme is applied at $x = 0$ using the boundary variation of the advected quantity at inlet to obtain an estimate for a fictitious point at $x = -h$ so that Quickest may be applied consistently at the first interior point. A word of warning: it is not clear how this scheme might be implemented in a practical stream-function vorticity solver; we are able to do so here because the Poisson equation for the stream function is solved exactly. If the solution were to proceed through an iterative solver it is unclear how the updated boundary vorticity could be incorporated in practice. This example is included mostly to show how improving the treatment of the first interior point may lead to significant stability improvement.

The premise of this model is to assume the existence of an upstream point at $x = -h$ and apply Lax-Wendroff at $x = 0$ giving

$$W_0^{n+1} = a_l W_{-1}^n + a_c W_0^n + a_r W_1^n, \tag{8.42}$$

where the coefficients are

$$a_l = \mu + \frac{1}{2}\nu^2 + \nu,\ a_c = 1 - 2\mu + \nu^2, a_r = \mu + \frac{1}{2}\nu^2 - \nu. \tag{8.43}$$

What is assumed is that the boundary values W_0^{n+1} and W_0^n are known, so that this equation can be rearranged to give

$$W_{-1}^n = \frac{1}{a_l} W_0^{n+1} - \frac{a_c}{a_l} W_0^n - \frac{a_r}{a_l} W_1^n. \tag{8.44}$$

Of course in the case of convection-diffusion of a material quantity such as concentration, the boundary values at the inlet are known as an explicit boundary condition, it is only in the case of this form of the Navier-Stokes equations that the boundary values are coupled to the solution in the interior of the domain at the new time level.

This approximation enables Quickest to be applied in unmodified form at the first interior point, $x = h$, so that

$$W_1^{n+1} - \frac{s_{ll}}{a_l} W_0^{n+1} = (s_l - \frac{s_{ll} a_c}{a_l}) W_0^n + (s_c - \frac{s_{ll} a_r}{a_l}) W_1^0 + s_r W_2^n. \tag{8.45}$$

This gives the matrices **A** and **B** as

$$\mathbf{A} = \begin{bmatrix} s_c - \frac{s_{ll}a_r}{a_l} & s_r & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ s_l & s_c & s_r & 0 & \cdots & 0 & 0 & 0 & 0 \\ s_{ll} & s_l & s_c & s_r & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & & \ddots & & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & s_{ll} & s_l & s_c & s_r \\ 0 & 0 & 0 & 0 & \cdots & 0 & s_{ll} & s_l & s_c \end{bmatrix}, \tag{8.46}$$

and

$$\mathbf{B} = \begin{bmatrix} s_l - \frac{s_{ll}a_c}{a_l} & 0 \\ s_{ll} & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & s_r \end{bmatrix}. \tag{8.47}$$

The complication that this modification introduces to the scheme is now clear because the iteration (8.24) has to be modified to

$$\mathbf{V}\mathbf{W}^{n+1} = \mathbf{K}\mathbf{W}^n. \tag{8.48}$$

Fortunately in this case the matrix **V** is particularly simple,

$$\mathbf{V} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & -s_{ll}/a_l & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 & 0 \\ \vdots & & & \ddots & & & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 1 \end{bmatrix}. \tag{8.49}$$

Hence it is straightforward to apply Gauss elimination to the matrix **K** to effectively apply $\mathbf{V}^{-1}$ by adding a multiple s_{ll}/a_l of the second last row of **K** to the first row and reduce (8.48) to the same form as (8.24).

The results of computing the stability boundary are shown in figure 8.13. Both the convection-diffusion operator and the Navier-Stokes type operators show much improved stability region over the other methods we have considered although again, the Navier-Stokes type equations has a more restricted stability region compared to the convection-diffusion operator.

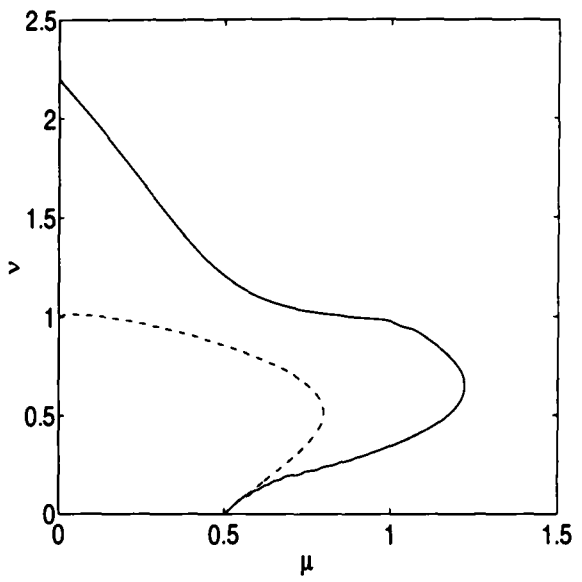


Figure 8.13: Contours of unit spectral radius for Quickest with Lax-Wendroff at inlet. (—) convection-diffusion operator, (- -) Navier-Stokes operator. The region of stability is the area between the curves and the axes.

HODS method

As an example of the application of this theory to an alternative discretisations scheme, consider the HODS scheme described in Castro and Jones [10]. In their case the advection term was discretised by

$$u\frac{\partial \omega}{\partial x} \approx \frac{u}{h}(\frac{3}{2}W_j - 2W_{j-1} + \frac{1}{2}W_{j-2}). \tag{8.50}$$

We have combined this with a central difference for the diffusion term and applied first order upwinding at the first interior mesh point. The results for stability are shown in figure 8.14.

Second order vorticity on walls

There are different formulas for the boundary vorticity that can be found in the literature. For a survey and discussion of boundary vorticity formulas, see for instance Napolitano *et al* [52].

An approximation for the wall vorticity that uses additional points for the stream function in the domain to increase the accuracy of the vorticity approximation, which is a second order scheme, is given by

$$h^2W_0 = -4\Psi_1 + \frac{1}{2}\Psi_2, \quad h^2W_M = -4\Psi_{M-1} + \frac{1}{2}\Psi_{M-2}. \tag{8.51}$$

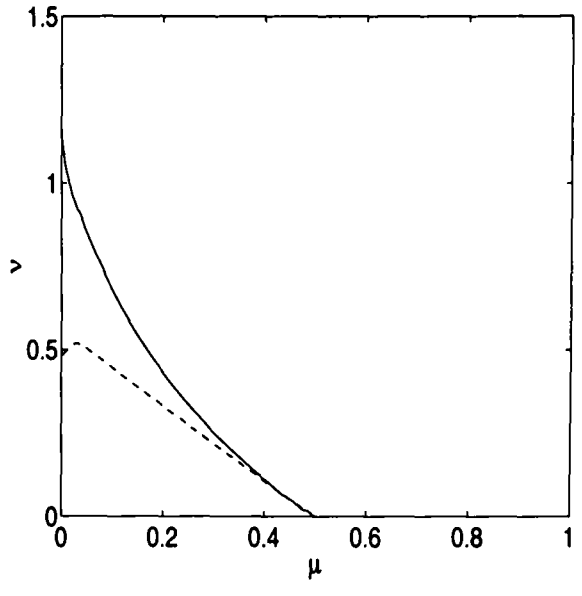


Figure 8.14: Contours of unit spectral radius for the HODS scheme. (—) convection-diffusion operator, (- -) Navier-Stokes operator. The region of stability is the area between the curves and the axes.

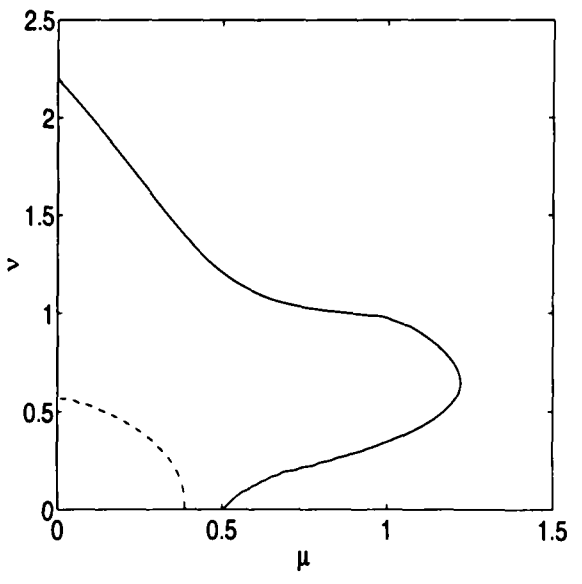


Figure 8.15: Contours of unit spectral radius for Quickest (using lax Wendroff at inlet) and second order boundary vorticity. (—) convection-diffusion operator, (- -) Navier-Stokes operator. The region of stability is the area between the curves and the axes.

This variation can be easily implemented by varying the matrix M. If for instance this is done for Quickest using Lax-Wendroff at the inlet then the stability curve shown in figure 8.15 is obtained. It is evident that attempting to improve accuracy in this way has a substantial stability penalty.

8.4 Summary

In this chapter we have shown that numerical solutions of the two-dimensional Navier-Stokes equations lead to stability problems which are both similar and different to the convection-diffusion equation alone. They are similar in that results obtained for convection-diffusion problems can be used to give general guidelines for the stream-function vorticity solutions but they are different in that the fine detail of stability regions is altered, generally they are more restrictive.

We have also shown that the Navier-Stokes solver can, at least symbolically, be put into the context of a global iteration matrix and we have explored some of the consequences for a simple one dimensional model, again with the conclusion that stability regions are more restrictive when the global solution is considered.

Chapter 9

Conclusion

9.1 Concluding remarks

This dissertation centred around the fact that we have an exact evolution operator for a constant coefficient convection-diffusion problem. We have used this operator to derive new finite difference schemes in one dimension and two dimensions as well as to derive numerical boundary conditions. The influence of the numerical boundary conditions on the stability of a general scheme was one of the main themes. In most of the cases the stability analysis was performed by using the von Neumann method and the matrix method.

The Godunov-Ryabenkii theory was applied to the one-dimensional case. Additionally, in connection with this theory, we gave some new theoretical results and developed a suitable algorithm to identify the stability region for our schemes.

We also highlighted the importance of the four-point scheme provided by Leonard's Quickest scheme by showing its superiority in terms of stability over the Lax-Wendroff method in both one dimension and two dimensions. In two dimensions we derived different forms of Lax-Wendroff schemes and Quickest schemes. Then, we applied some of these schemes to a Navier-Stokes problem and computed the practical stability regions by carrying out numerical experiments, showing how results from a simpler case presented in earlier chapters carry over to the more complex case.

We ended the last chapter by giving a global iteration matrix for the stream-function vorticity formulation for the Navier-Stokes equations, showing that the true time marching iteration matrix is more complicated than the iteration matrix for the convection-diffusion equation that is part of the Navier-Stokes equa-

tions. We examined a one-dimensional test problem which has shown the full system to have tighter stability constraints than would have been predicted from the convection-diffusion equation alone. More work can be done from this point on and we describe some of it in the next section.

9.2 Further work

To conclude this dissertation we would like to describe some open problems encountered during the research. We have tried to cover as comprehensive as possible the stability of one-dimensional finite difference schemes derived from an evolutionary operator. Consequently most of the remaining significant problems are for two and three dimensions.

(a) One challenging problem left unsolved for a one dimensional problem is the application of Godunov-Ryabenkii theory to further examples, such as the global iteration matrix derived in chapter 8. Such a problem is more complex than the ones considered in this dissertation, since we have two physical boundary conditions: an inflow and an outflow boundary condition. Nevertheless, the theory shows that each boundary can be analysed separately. The difficulty of the problem might also be associated with the analytical form of the boundary conditions since usually simple problems already lead to complicated algebraic forms. One way to avoid the algebraic complexity is to adapt the algorithm in chapter 4 to this problem.

(b) The evolutionary operator has been applied to derive finite element schemes in one dimension in Morton and Sobey [49] but there has been no such work for two-dimensional problems.

(c) The practical extension of the one-dimensional model in chapter 8 for the global matrix to a two dimensional case would be extremely interesting although it would involve more challenging programming than in the one-dimensional case. An additional complication would be the need to include non-uniform velocity fields in the matrix A. However this effect is almost certainly worthwhile as it should lead to very good stability bounds for the Navier-Stokes equations. Additionally the model could be extended to study non-linear stability as in Bagget *et al* [2] and Bagget and Trefethen [3]. There the linear iteration was augmented by a small non-linear term with the result that bounded oscillations

could vary for sufficiently large thresholds disturbances. The main interest was in the width of the basin of attraction of the stable solutions as the Reynolds number varied. So far this has only been considered for simple models (two or three equations) or experimentally (e.g. Darbyshire and Mullin [12]) but the global iteration matrix should provide a more direct Navier-Stokes solver for such studies.

(d) The schemes we have derived in two dimensions need to be tested in primitive variable Navier-Stokes solvers to see whether they offer any advantages over existing discretisations.

(e) The new schemes we have derived need to be tested in a number of two-dimensional problems, particularly channel flows. Previous work (see e.g. Uchibori and Sobey [87]) studying the onset of a simple bifurcation has shown extreme sensitivity to discretisation schemes using first order upwind or central differences. Now we have a range of other two-dimensional schemes and it would be interesting to study their accuracy for this simple bifurcation problem.

Appendix A

Error analysis for the new schemes

Following Morton and Sobey [49] suggestion, the truncation error is given by

$$T^n = \frac{1}{\Delta t} RE(\Delta t)(u^n - I_p R u^n),$$

where $E(\Delta t)$ is the evolution operator $u(\cdot, t+\Delta t) = E(\Delta t)u(\cdot, t)$, for our problem in the semi line $x \geq 0$, R is the restriction operator onto the nodes and I_p is the local approximation based on nodal values.

Let us define the interpolation error $Lu^n = u^n - I_p u^n$ and define for $a \geq 0$, the integrals of the form

$$E_m(a;\mu) = \int_0^\infty \xi^m \mathrm{e}^{-(\xi+a)^2/4\mu} \frac{d\xi}{2\sqrt{\pi\mu}}. \tag{A.1}$$

In what follows we denote $s = (x - x_j)/\Delta x$ and we omit the subscript n, referring only to $u(x)$ and its evolution over one time step.

Quadratic interpolation

We calculate the error at the point $x_j = j\Delta x$. We have, for $j \geq 1$

$$Lu = u(x) - \frac{1}{2}[-s + s^2]u(x_{j-1}) - [1 - s^2]u(x_j) - \frac{1}{2}[s^2 + s]u(x_{j+1}).$$

Then using the Peano kernel theorem we can write

$$Lu = \int_0^\infty K(x,p)u^{(3)}(p)dp,$$

where $K(x,p) = (1/2)L_x[(x-p)_+^2]$, L_x refers to L acting in x, and $(x-p)_+^2 = (x-p)^2$ if $x-p \geq 0$, and zero otherwise. We calculate the Peano kernel function $K(s\Delta x + j\Delta x, \xi\Delta x)$,

$$K = \frac{1}{2}\Delta x^2 \begin{cases} (s+j-\xi)_+^2 - (j-1-\xi)^2(-s/2+s^2/2) & \\ -(j-\xi)^2(1-s^2) - (j+1-\xi)^2(s^2/2+s/2), & 0<\xi<j-1, \\ (s+j-\xi)_+^2 - (j-\xi)^2(1-s^2) & \\ -(j+1-\xi)^2(s^2/2+s/2), & j-1<\xi<j, \\ (s+j-\xi)_+^2 - (j+1-\xi)^2(s^2/2+s/2), & j<\xi<j+1, \\ (s+j-\xi)_+^2, & \xi>j+1. \end{cases}$$

The local error is given by

$$\Delta tT = RE_x \int_0^\infty K(x,p)u^{(3)}(p)dp = \int_0^\infty RE_x K(x,p)u^{(3)}(p)dp$$

where we use the notation E_x to describe $E(\Delta t)$ acting on x. After some manipulation to calculate $RE_xK(x,p)u^{(3)}(p)$ we have,

$$RE_xK = \frac{1}{2}\Delta x^2 \begin{cases} E_2(\xi+\nu-j;\mu) - \mathrm{e}^{\nu j/\mu}E_2(\xi+\nu+j;\mu) & \\ -(j-1-\xi)^2(-f_1+f_2)/2 & \\ -(j-\xi)^2(f_3-f_2) - (j+1-\xi)^2(f_2+f_1)/2, & 0<\xi<j-1, \\ E_2(\xi+\nu-j;\mu) - \mathrm{e}^{\nu j/\mu}E_2(\xi+\nu+j;\mu) & \\ -(j-\xi)^2(f_3-f_2) - (j+1-\xi)^2(f_2+f_1)/2, & j-1<\xi<j, \\ E_2(\xi+\nu-j;\mu) - \mathrm{e}^{\nu j/\mu}E_2(\xi+\nu+j;\mu) & \\ -(j+1-\xi)^2(f_2+f_1)/2, & j<\xi<j+1, \\ E_2(\xi+\nu-j;\mu) - \mathrm{e}^{\nu j/\mu}E_2(\xi+\nu+j;\mu), & \xi>j+1, \end{cases}$$

where,

$$\begin{aligned} f_1 &= -\nu a(j) + b(j) \\ f_2 &= (\nu^2+2\mu)a(j) + 2c(j) \\ f_3 &= a(j) \\ f_4 &= -(\nu^3+6\mu\nu)a(j) + e(j). \end{aligned}$$

The exact error at x_j is given by

$$\Delta tT_{|x_j} = \Delta x^3[\int_0^{+\infty} K_1(\xi,\nu,\mu)u^{(3)}(\xi\Delta x)d\xi + \int_0^{j+1} K_2(\xi,\nu,\mu)u^{(3)}(\xi\Delta x)d\xi].$$

where we have introduced two functions

$$K_1(\xi,\nu,\mu) = \frac{1}{2}[E_2(-j+\nu+\xi;\mu) - e^{\nu j/\mu}E_2(j+\nu+\xi;\mu)];$$

$$K_2(\xi,\nu,\mu) = \begin{cases} -(f_2-f_1)(j-1-\xi)^2/4 & \\ +(f_2-f_3)(j-\xi)^2/2 & \\ -(f_2+f_1)(j+1-\xi)^2/4 & 0 \le \xi < j-1, \\ (f_2-f_3)(j-\xi)^2/2 & \\ -(f_2+f_1)(j+1-\xi)^2/4 & j-1 \le \xi < j, \\ -(f_2+f_1)(j+1-\xi-j)^2/4 & j \le \xi \le j+1. \end{cases}$$

Although the exact error expression appears complicated, it can be used to examine the detailed structure of the error and to obtain overall bounds for the local error.

Considering first the structure of the error, if we assume $u \in C^\infty(\mathbb{R})$ and use a Taylor series expansion for $u^{(3)}$ around x_j, then after some algebra,

$$\Delta tT|_{x_j} = \frac{1}{6}\Delta x^3 u_{xxx} g_j^2(\nu,\mu) + O(\Delta x^4 u_{x^4}),$$

where

$$g_j^2(\nu,\mu) = \nu(1-\nu^2-6\mu)a(j) - b(j) + e(j).$$

Secondly, we can obtain an overall bound for the local error since

$$|\Delta tT|_{x_j}| \le \Delta x^3 |u|_{3,\infty}[\int_0^{+\infty} |K_1(\xi,\nu,\mu)|d\xi + \int_0^{j+1} |K_2(\xi,\nu,\mu)|d\xi].$$

Cubic interpolation

We have

$$\begin{aligned} Lu &= u(x) - \frac{1}{6}[s-s^3]u(x_{j-2}) - \frac{1}{2}[-2s+s^2+s^3]u(x_{j-1}) \\ &\quad -\frac{1}{2}[2+s-2s^2-s^3]u(x_j) - \frac{1}{6}[2s+3s^2+s^3]u(x_{j+1}). \end{aligned}$$

The Peano kernel function $K(x,p)$ for $x = s\Delta x + j\Delta x$ and $p = \Delta x\xi$ is given by

$$\begin{aligned} K &= \frac{1}{6}\Delta x^3 \Big[(s+j-\xi)_+^3 - \frac{1}{6}(j-2-\xi)_+^3(s-s^3) \\ &\quad -\frac{1}{2}(j-1-\xi)_+^3(-2s+s^2+s^3) - \frac{1}{2}(j-\xi)_+^3(2+s-2s^2-s^3) \\ &\quad -\frac{1}{6}(j+1-\xi)_+^3(2s+3s^2+s^3)\Big]. \end{aligned}$$

Summarising the calculations in this case, the exact error for $x_j, j \ge 2$ is given by

$$\Delta tT|_{x_j} = \frac{\Delta x^4}{6}[\int_0^{+\infty} W_1^2(\xi,\nu,\mu)u^{(4)}(\xi\Delta x)d\xi + \int_0^{j+1} W_2^2(\xi,\nu,\mu)u^{(4)}(\xi\Delta x)d\xi].$$

where the functions

$$W_1^2(\xi,\nu,\mu) = [E_3(-j+\nu+\xi;\mu) - e^{\nu j/\mu}E_3(j+\nu+\xi;\mu)];$$

$$W_2^2(\xi,\nu,\mu) = \begin{cases} q_1(\xi-j+2)^3/6 & \\ +q_2(\xi-j+1)^3/2 & \\ +q_3(\xi-j)^3/2 + q_4(j)(\xi-j-1)^3/6, & 0 \le \xi < j-2, \\ q_2(\xi-j+1)^3/2 + q_3(j)(\xi-j)^3/2 & \\ +q_4(\xi-j-1)^3/6, & j-2 \le \xi < j-1, \\ q_3(\xi-j)^3/2 + q_4(j)(\xi-j-1)^3/6, & j-1 \le \xi \le j, \\ q_4(\xi-j-1)^3/6, & j \le \xi \le j+1. \end{cases}$$

where

$$\begin{aligned} q_1 &= f_1 - f_4, \\ q_2 &= f_4 + f_2 - 2f_1, \\ q_3 &= 2f_3 - f_4 + f_1 - 2f_2, \\ q_4 &= 2f_1 + 3f_2 + f_4. \end{aligned}$$

If we assume $u \in C^\infty(\mathbb{R})$ and use a Taylor expansion for $u^{(4)}$ then the truncation error is given by the cumbersome expression

$$\Delta t T|_{x_j} = \frac{1}{24}\Delta x^4 u_{xxxx} g_j^3(\nu,\mu) + \cdots,$$

with

$$\begin{aligned} g_j^3(\nu,\mu) &= (12\mu^2 - 2\mu - 12\mu\nu(1-\nu) + \nu(1-\nu^2)(2-\nu))a(j) \\ &\quad -2b(j)(12\mu\nu + 2\nu^3 + 2j\nu + 1 + 2j^3) \\ &\quad +2c(j)(1+6j^2) + 2(1-2j)e(j) + 2Z(j)(3\nu^2 + j^2 + 10\mu). \end{aligned}$$

Bibliography

[1] Allen, D. and R. Southwell (1955). Relaxation methods applied to determining the motion, in two dimensions, of a viscous fluid past a cylinder. *Quart. J. Mech. Appl. Math.* **8**, 129-145.

[2] Baggett, J.S., T.A. Driscoll and L.N. Trefethen (1995). A mostly linear model of transition to turbulence. *Physics of Fluids* **7**, 833-838.

[3] Baggett, J.S. and L.N. Trefethen (1997). Low dimensional models of subcritical transiction to turbulence. *Physics of Fluids* **9**, 1043-1053.

[4] Baum, H.R., M. Ciment, R.W. Davis and E.F. Moore (1981). Numerical solutions for a moving shear layer in a swirling axisymmetric flow, in: Numerical Methods in Fluid Dynamics, Eds. W.C. Reynolds and R.W. MacCormack. *Lect. Notes in Physics* **141**, 74-79.

[5] Beckers, J.M. (1992). Analytical linear numerical stability conditions for an anisotropic three-dimensional advection-diffusion equation. *SIAM Journal of Numerical Analysis* **29**, 701-713.

[6] Benjamin, A.S. and V.E. Denny (1979). On the convergence of numerical solutions for 2-D flows in a cavity at large Re. *Journal of Computational Physics* **33**, 340-358.

[7] Bruneau, C.H. and C. Jouron (1988). A new upwind scheme for the driven cavity flow. *C. R. des Academie des Sciences* **307**, 359-362.

[8] Bruneau, C.H., P. Fabrie and P. Rasetarinera (1997). An accurate finite difference scheme for solving convection-dominated diffusion equations *International Journal for Numerical Methods in Fluids* **24**, 169-183.

[9] Carpenter, M.H., D. Gottlieb and S. Abarbanel (1993). The stability of numerical boundary treatments for compact high-order finite-difference schemes. *Journal of Computational Physics* **108**, 272-295.

[10] Castro, I.P. and J.M. Jones (1987). Studies in numerical computations of recirculating flows. *International Journal for Numerical Methods in Fluids* **7**, 793-823.

[11] Chan, T.F. (1984). Stability analysis of finite difference schemes for the advection-diffusion equation. *SIAM Journal of Numerical Analysis* **21**, 272-283.

[12] Darbyshire, A. G. and T. Mullin (1995). Transition to turbulence in constant-mass-flux pipe flow. *Journal of Fluid Mechanics* **289**, 83-114.

[13] Davis, R.W. and E.F. Moore (1982). A numerical study of vortex shedding from rectangles. *Journal of Fluid Mechanics* **116**, 475-506.

[14] Douglas, J. and T.F. Russel (1982). Numerical methods for convection dominated diffusion problems based on combining the method of characteristics with finite element of finite difference procedures. *SIAM Journal of Numerical Analysis* **19/5**, 871-885.

[15] Fletcher, C.A.J. (1988). *Computational Techniques for Fluid Dynamics*, Vol I and II, Springer-Verlag: Berlin.

[16] Fromm, J. (1964). The time dependent flow of an incompressible viscous fluid. *Methods in Computational Physics* **3**, 345-382.

[17] Gautschi, W. (1962). On inverses of Vandermonde and confluent Vandermonde matrices. *Numerische Mathematik* **4**, 117-123.

[18] Gautschi, W. (1963). On inverses of Vandermonde and confluent Vandermonde matrices II. *Numerische Mathematik* **5**, 425-430.

[19] Gautschi, W. (1978). On inverses of Vandermonde and confluent Vandermonde matrices III. *Numerische Mathematik* **29**, 445-450.

[20] Gautschi, W. (1997). *Numerical Analysis*, Birkhauser: Boston.

[21] Godunov, S.K. and V.S. Ryabenkii (1963). Spectral criteria for the stability of boundary problems for non-self-adjoint difference equations. *Uspekhi Mat. Nauk.* 18 (In Russian).

[22] Godunov, S.K. and V.S. Ryabenkii (1964). *The theory of difference schemes*, Amsterdam: North-Holland.

[23] Gresho, P.M. and R.E. Lee (1981). Don't supress the wiggles, they are telling you something! *Computers and Fluids* **9**, 223-253.

[24] Goldberg, M. and E. Tadmor (1978). Scheme-independent stability criteria for difference approximations of hyperbolic initial-boundary value problems. *Mathematics of Computation* **32**, 1097-1107.

[25] Goodrich, J.W., K. Gustafson and K. Halasi (1990). Hopf bifurcation in the driven cavity. *Journal of Computational Physics* **90**, 219-261.

[26] Griffiths, D.F., I. Christie and A.R. Mitchell (1980). Analysis of error growth for explicit difference schemes in conduction-convection problems. *International Journal for Numerical Methods in Engineering* **15**, 1075-1081.

[27] Gustafsson, B., H.O. Kreiss and J. Oliger (1995). Time-dependent problems and difference methods. Wiley-Interscience.

[28] Gustafsson, B., H.O. Kreiss and A. Sundstrom (1972). Stability theory of difference approximations for mixed initial boundary value problems, II. *Mathematics of Computation* 26, 649-686.

[29] Hindmarsh, A.C., P.M. Gresho and D.F. Griffiths (1984). The stability of explicit Euler time-integration for certain finite difference approximations of the multi-dimensional advection-diffusion equation. *International Journal for Numerical Methods in Fluids* **4**, 853-897.

[30] Hirsch, C. (1990). *Numerical Computaion of Internal and external flows*, Vol I and II, Wiley Interscience: Chichester.

[31] Hirt, C.W. (1968). Heuristic stability theory for finite difference equations. *Journal of Computational Physics* **2**, 339-355.

[32] Hou, T.Y. and B.T.R. Wetton (1992). Convergence of a finite difference scheme for the Navier-Stokes equations using vorticity boundary conditions. *SIAM Journal of Numerical Analysis* **29**, 615-639.

[33] Il'in, A.M. (1969). Accurate monotone cubic interpolation. *SIAM J. Numer. Anal.* **30**, 57-100.

[34] Johnson, R.W. and R.J. MacKinnon (1992). Equivalent versions of the Quick scheme for finite-difference and finite-volume numerical methods. *Communications in applied numerical methods* **8**, 841-847.

[35] Kreiss, H.O. (1968). Stability theory for difference approximations of mixed initial boundary value problems, I. *Mathematics of Computation* **22**, 703-714.

[36] Kurganov, A. and E. Tadmor (2000). New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations. *Journal of Computational Physics* **160**, 241-282.

[37] Kwok, Y-K and K-K Tam (1993). Stability analysis of three-level difference schemes for initial-boundary problems for multidimensional convective-diffusion equations. *Communications in Numerical Methods in Engineering* **9**, 595-605.

[38] Lax, P.D. and B. Wendroff (1960). Systems of conservations laws. *Communications on Pure and Applied Mathematics* **13**, 217-237.

[39] Lax, P.D. and B. Wendroff (1964). Difference schemes for hyperbolic equations with high order of accuracy. *Communications on Pure and Applied Mathematics* **17**, 381-398.

[40] Lenferink, H.W.J. and M.N. Spijker (1991). On the use of stability regions in the numerical analysis of initial value problems. *Mathematics of Computation* **57**, 221-237.

[41] Leonard, B.P. (1979). A stable and accurate convective modelling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering* **19**, 59-98.

[42] Leonard, B.P. (1980). Note on the von Newmann stability of the explicit FTCS convection diffusion equation. *Applied Mathematical Modeling* **4**, 401-402.

[43] Leonard, B.P. (1987). Locally-modifed QUICK scheme for highly convective 2D and 3D flows. *Proc. 5th Conf. Num. Meth. in Laminar and Turbulent Flow.* Montreal.

[44] Leonard, B.P. and S. Mokhtari (1990). Beyond first-order upwinding the ultra-sharp alternative for non-oscillatory steady-state simulation of convection. *International Journal for Numerical Methods in Engineering* **30**, 729-766.

[45] Leonard, B.P. (1994). Note on the von Neumann stability of explicit one-dimensional advection schemes. *Computational Methods in Applied Mechanical Engineering* **118**, 29-46.

[46] Leonard, B.P. (1994). Comparison of truncation error of finite-difference and finite-volume formulations of convection terms. *Appl. Math. Modelling* **18**, 46-50.

[47] Michelson, D. (1983). Stability theory of difference approximations for multidimensional initial-boundary value problems. *Mathematics of Computation* 40, 1-45.

[48] Morton, K.W. (1980). Stability of finite difference approximations to a diffusion-convection equation. *International Journal for Numerical Methods in Engineering* **15**, 677-683.

[49] Morton, K.W and I.J Sobey (1993). Discretization of a convection-diffusion equation. *IMA Journal of Numerical Analysis* **13**, 141-160.

[50] Morton, K.W. (1996). *Numerical Solution of Convection-Diffusion Problems*, Chapman and Hall: London.

[51] Morton, K.W. and Mayers (1994). *Numerical solution of partial differential equations*, Cambridge University Press: Cambridge.

[52] Napolitano, M., G. Pascazio and L. Quartapelle (1999). A review of vorticity conditions in the numerical solution of the $\omega - \psi$ equations. *Computer and Fluids* **28**, 139-185.

[53] Oliger, J. (1974). Fourth order difference methods for the initial boundary-value problem for hyperbolic equations. *Mathematics of Computations* **28**, 15-25.

[54] Oliger, J. (1976). Hybrid difference methods for the initial boundary-value problem for hyperbolic equations. *Mathematics of Computations* **30**, 724-738.

[55] Osher, S. (1969). Stability of difference approximations of dissipative type for mixed initial-boundary value problems. *Mathematics of Computation* 23, 335-340.

[56] Otto, K. and M. Thune (1989). Stability of a Runge-Kutta method for the Euler equations on a substructured domain. *SIAM Journal of Scien Stat. Comput.* **10**, 154-174.

[57] Peyret, R. and T.D. Taylor (1986). *Computational methods for fluid flow*, Springer-Verlag: Berlin.

[58] Powell, M.J.D. (1981). *Approximation theory and method*, Cambridge.

[59] Reddy, S.C. and L.N. Trefethen (1990). Lax-stability of fully discrete spectral methods via stability regions and pseudo-eigenvalues. *Computer Methods in Applied Mechanics and Engineering* **80**, 147-164.

[60] Reddy, S.C. and L.N. Trefethen (1992). Stability of the method of lines. *Numerische Mathematik* **62**, 235-267.

[61] Reddy, S.C. and L.N. Trefethen (1994). Pseudospectra of the convection-diffusion operator. *SIAM Journal of Applied Mathematics* **54**, 1634-1649.

[62] Richtmyer, R.D. and K.W. Morton (1967). *Difference methods for initial-value problems.* Wiley-Interscience: New York.

[63] Rigal, A. and G. Aleix (1978). Stability analysis of some finite difference schemes for the Navier-Stokes equations. *International Journal for Numerical Methods in Engineering* **12**, 1399-1405.

[64] Rigal, A. (1979). Stability analysis of explicit finite difference schemes for the Navier-Stokes equations. *International Journal for Numerical Methods in Engineering* **14**, 617-620.

[65] Roache, P.J. (1972). *Computational Fluid Dynamics*, Albuquerque, New Mexico: Hermosa.

[66] Roos, H.G. (1994). Ten ways to generate the Il'in and related schemes. *Journal of Computational and Applied Mathematics* **53**, 43-59.

[67] Roos, H.G., M. Stynes and L. Tobiska (1996). *Numerical methods for singularly perturbed differential equations – convection diffusion and flow problems*, Springer: Berlin.

[68] Shen, J. (1991). Hopf bifurcation of the unsteady regularized driven cavity flow. *Journal of Computational Physics* **95**, 228-245.

[69] Schreiber, R. and H.B. Keller (1983). Driven cavity flows by efficient numerical techniques. *Journal of Computational Physics* **49**, 310-333.

[70] Siemieniuch, J. and I. Gladwell (1978). Analysis of explicit difference methods for the diffusion-convection equation. *International Journal for Numerical Methods in Engineering* **12**, 899-916.

[71] Sloan, D.M.(1983). Boundary conditions for a fourth order hyperbolic difference scheme. *Mathematics of Computation* **41**, 1-11.

[72] Smith, G.D. (1985). *Numerical solution of partial differential equations: finite difference methods*, Oxford University Press: Oxford.

[73] Sod, G.A. (1988). *Numerical methods in fluid dynamics: initial and initial boundary-value problems*, Cambridge University Press: Cambridge.

[74] Sousa, E. and I.J. Sobey (1998). Finite Difference Approximation of a Convection Diffusion Equation Near a Boundary, OUCL Numerical Analysis Group Technical Report 98/16.

[75] Sousa, E. (2001). A Godunov-Ryabenkii instability for a Quickest scheme, in: Numerical Analysis and Applications, Eds. J. Wasniewski, L. Vulkov and P. Yalamov, *Lecture Notes in Computer Science* **1988**, Springer Verlag.

[76] Sousa, E. and I.J. Sobey (2001). On the influence of boundary conditions. Submitted for publication.

[77] Sousa, E. and I.J. Sobey (2001). Stability of unsteady stream-function vorticity calculations. In preparation.

[78] Strikwerda, J. (1980). Initial boundary value problems for the method of lines. *Journal of Computational Physics* **34**, 94-107.

[79] Strikwerda, J. (1989). *Finite difference schemes and partial differential equations*, Wadsworth & Brooks: California.

[80] Thom, A. (1933). The flow past circular cylinders at low speeds, *Proceedings of the Royal Society of London* **A141**, 651-666.

[81] Thompson, H.D., B.W. Webb and J.D. Hoffmann (1985). The cell Reynolds number myth. *International Journal for Numerical Methods in Fluids* **5**, 305-310.

[82] Thuné, M. (1986). Automatic GKS stability analysis. *SIAM Journal of Sci. Stat. Comp.* **7**, 959-977.

[83] Thuné, M. (1990). A numerical algorithm for stability analysis of difference methods for hyperbolic systems. *SIAM Journal of Sci. Stat. Comp.* **11**, 63-81.

[84] Trefethen, L.N. (1983). Group velocity interpretation of the stability theory of Gustafsson, Kreiss and Sundstrom. *Journal of Computational Physics* **49**, 199-217.

[85] Trefethen, L.N. (1984). Instability of difference models for hyperbolic initial boundary value problems. *Comm. Pure and Applied Mathematics* **37**, 329-367.

[86] Turkel, E. (1977). Symmetric Hyperbolic difference schemes and Matrix Prolems. *Linear Algebra and its Applications* **16**, 109-129.

[87] Uchibori, Y. and I.J.Sobey (1992). Dependence on numerical algorithm of bifurcation structure for flow through a symmetric expansion. OUCL Numerical Analysis Group Technical Report 92/14.

[88] Verwer, J.G. and Sommeijer (1997). Stability analysis of an odd-even-line hopscotch method for three-dimensional advection-diffusion problems. *SIAM Journal of Numerical Analysis* **34**, 376-388.

[89] van Dorsselaer, J.L.M., J.F.B.M. Kraaijevanger and M.N. Spijker (1993). Linear stability analysis in the numerical solution of initial value problems. *Acta Numerica* 199-237.

[90] Varah, J.M. (1971). Stability of difference approximations to the mixed initial boundary value problems for parabolic systems. *SIAM Journal Of Numerical Analysis* **8**, 598-615.

[91] Warming, F.F. and B.J. Hyett (1974). The modified equation approach to the stability and accuracy analysis of finite difference methods. *Journal of Computational Physics* **14**, 159-179.

[92] Wesseling, P.(1996). Von Newmann stability conditions for the convection-diffusion equation. *IMA Journal of Numerical Analysis* **16**, 583-598.

[93] Wetton, B.T.R. (1992). Finite difference vorticity methods. *Lecture Notes in Mathematics* **1530**, 210-225.

[94] Xu, H.Y., M.D. Matovic and A. Pollard (1997). Finite difference schemes for three-dimensional time-dependent convection-diffusion equation using full global discretization. *Journal of Computational Physics* **130**, 109-122.

[95] Woods, L.C. (1954). A note on the numerical solution of fourth order differential equations. *Aeronautical Quarterly* **5**, 176.