Fundamentals of Atmospheres and Oceans on Computers

Lars Petter Røed†,⋆,‡

†Research and Development Department, Norwegian Meteorological Institute, Postboks 43 Blindern, 0313 Oslo, Norge.
★Department of Geosciences, Section Meteorology and Oceanography, University of Oslo, P. O. Box 1022 Blindern, 0315 Oslo, Norway
‡E-mail: larspetter.roed@met.no
PREFACE

To me atmospheres and oceans on computers are one of the most fascinating tools in contemporary meteorology and oceanography. I therefore strongly believe that everyone who ventures into these fields and aspire to become a meteorologist or an oceanographer should have an insight into the fundamental methods used to develop numerical oceanographic and meteorological models.

These lecture notes are directed towards Master students in Meteorology and Oceanography to give them insight into the basic concepts and methods we use to put atmospheres and oceans on computers and some of the caveats met. In this respect these notes are also useful to scientists unfamiliar with numerical methods used in geophysical fluid dynamics. The content is based on the notes I have used to teach numerical methods to solve oceanographic problems over the past 20 years or so. I should emphasize that the material covers the fundamental methods used in numerical weather prediction models as well as in similar oceanographic models. In this respect I would like to point out that many of the numerical methods presented were originally developed within the meteorological community. Most of them, if not all, are now adopted and widely used also in the oceanographic community. It is therefore timely to compose a set of lecture notes that combine the now more or less common set of fundamental numerical methods.

A further rationale is the fact that the atmosphere and the ocean is inherently a coupled system. To fully appreciate and understand the coupled system and their modeling one therefore needs to have knowledge of meteorology as well as oceanography. This fact is perhaps most evident within climate modeling. I therefore believe, even though there are a few differences in the methods employed, that it is important that everyone, whether they want to become meteorologists or oceanographers, obtain a basic knowledge of the numerical methods used within both communities. Although most of the the basic numerical methods used to a large extent are the same, there are more advanced methods and techniques that are unique to each field. Additional courses must therefore be offered to those students where the numerical model is or will become an essential tool in their work.

Finally, I would like to extend my appreciation and sincere thanks to Dr. James J. O’Brien, former director of the Center for Atmospher-Ocean Predictions (COAPS), and now distinguished professor emiritus at the Florida State University, USA for introducing me to numerical methods to solve oceanographic and atmospheric problems. Some of the material covered are in fact based on notes taken when I followed his lectures at Florida State University back in 1980/81. I would also like to thank Prof. Thor Erik Nordeng, Norwegian Meteorological Institute for helping compile the necessary material from his lecture notes on atmospheres on computers. In this respect we both feel gratitude toward’s Dr. Arne Bratseth, former professor at the University of Oslo now deseased. Some of the material covered in his lecture notes on “Numerical Atmosphere Models” is explicitly used. I would also like to extend my thanks to the many students who has pointed out misprints and errors in this and earlier versions of hese notes over the years.

Blindern, July 11, 2008
Lars Petter Røed (sign.)
# Contents

PREFACE ................................................................. i

1 INTRODUCTION ..................................................... 1
   1.1 The governing equations .................................... 1
   1.2 Boundary conditions ........................................ 3
   1.3 The hydrostatic approximation .............................. 4
   1.4 The Boussinesq approximation .............................. 5
   1.5 The shallow water equations ............................... 6
   1.6 The quasi-geostrophic equations ......................... 7

2 PRELIMINARIES .................................................... 11
   2.1 General PDEs ................................................ 11
   2.2 Elliptic equations ......................................... 11
   2.3 Parabolic equations ....................................... 12
   2.4 Hyperbolic equations ..................................... 12
   2.5 Boundary conditions ....................................... 14
   2.6 Taylor expansions ......................................... 15
   2.7 Truncation errors .......................................... 17
   2.8 Notations .................................................. 18
   2.9 Orthogonal functions ..................................... 20
   2.10 Fourier series ............................................. 22
   2.11 Fourier transforms ....................................... 22

3 TIME MARCHING PROBLEMS ..................................... 24
   3.1 Advection-diffusion ....................................... 25
   3.2 Diffusion .................................................. 26
   3.3 Advection .................................................. 27

4 THE DIFFUSION PROBLEM ....................................... 29
   4.1 Finite difference form .................................... 30
   4.2 Numerical stability ....................................... 33
   4.3 Stability analysis ......................................... 34
   4.4 Necessary stability ...................................... 37
   4.5 Explicit and implicit schemes ......................... 38
List of Figures

1.1 The equation of state for the ocean. Dotted curves show isolines of density as a function of salinity (horizontal axis) and potential temperature (vertical axis) for a fixed pressure (= 0 dbars). Numbers on curves indicates density in \( \sigma_t \) units where \( \sigma_t = \rho - 1000 kg/m^3 \). Dashed line denotes the freezing point of sea water. Note that for low temperatures (temperatures close to the freezing point of sea water) the density is close to being a function of salinity alone, while the importance of temperature increases with increasing temperature. Due to the non-linear nature of the equation of state for sea water two parcels of water having different temperatures and salinities may still have the same density as for instance the two square points marked A and B along \( \sigma_t = 20.6 kg/m^3 \).

2.1 Displayed is a commonly used grid when employing numerical methods to solve PDEs. The points in the \( x, y \) directions are incremented by \( \Delta x, \Delta y \), respectively, so that there are a total of \( J \) points along the \( x \)-axis and \( K \) points along the \( y \)-axis. The points are counted by using the dummy counters \( j, k \).

4.1 Displayed is the employed grid we use to solve (4.1) by numerical means. The grid points in the \( x, t \) directions are incremented by \( \Delta x, \Delta t \), respectively. There is a total of \( J + 1 \) points along the \( x \)-axis and \( N + 1 \) points along the \( t \)-axis, counted by using the dummy indices \( j, n \). The coordinates of the grid points are \( x_j = (j - 1) \Delta x \) and \( t^n = n \Delta t \), respectively.

4.2 Displayed are solutions of the diffusion equation using the scheme (4.4) for respectively \( K = \kappa \Delta t / \Delta x^2 = 0.45 \) (left panel) and \( K = 0.55 \) (right panel). The solutions are shown for the time levels \( n = 0, n = 50 \) and \( n = 90 \). Note the sawtooth like pattern in the right panel for \( n = 90 \) not present in the left panel. This indicates that the stability condition (4.32) is violated for \( K = 0.55 \), but not for \( K = 0.45 \).

5.1 Sketch of the characteristics in the \( x, t \) plane. For \( u = u_0 = \text{constant} > 0 \) the characteristics are the straight lines sloping to the the right in \( x, t \) space as given by (5.45). If \( x = L \) marks the end of the computational domain, then all information about the initial condition is lost for times \( t > t_c \).
5.2 Sketch of the method of characteristics. The distance between the grid points are $\Delta t$ in the vertical and $\Delta x$ in the horizontal direction. The sloping solid line is the characteristic through the point $j, n + 1$. It is derived from (5.45) and the slope is given by $1/u$ ($u > 0$). The point labeled $Q$ is therefore a distance $u\Delta t$ to the left of $x_j$. As long as $u\Delta t < \Delta x$ then $Q$ is located between $x_{j-1}$ and $x_j$. If however $u\Delta t > \Delta x$ then the point $Q$ is located to the left of $x_{j-1}$.

5.3 Displayed is an example of the diffusion inherent in the upwind scheme. The solid curve shows the initial distribution at time level $n = 0$, while the dashed curve (red) shows the distribution at time level $n = 200$. The Courant number is $C = 0.5$. Cyclic boundary conditions are used at the boundaries of the computational domain.

5.4 Solutions to the advection equation using the MPDATA scheme suggested by Smolarkiewicz (1983). Left panel corresponds to a scaling factor of 1.0 (no scaling), while the right-hand panel employs a scaling factor of 1.3. The Courant number is 0.5 in both cases. Solid, black lines show the initial value (time step $n = 0$), while the red dotted lines show the solution after 200 time steps. The green dashed lines are after 400 time steps while the blue, dash-dot lines are after 800 time steps. Cyclic boundary conditions are employed. (cf. Computer problem No. 6 in the Computer Problem notes).

5.5 Numerical dispersion for the leapfrog scheme. The curves depicts the numerical phase speed as a function of the wavenumber based on (5.76) for various values of the Courant number $C = u\Delta t/\Delta x$. The vertical axis indicates the phase speed $c$ normalized by the advection speed $u$. The horizontal axis indicates the wavenumber normalized by $\pi/\Delta x$ where $\Delta x$ is the space increment or the grid size. The analytic dispersion curve is just a straight line corresponding to the phase speed $c = u$, that is $c/u = 1$. Note that as the wavenumber increases (that is the wavelength decreases) the numerical phase speed deviates more and more from the correct analytic phase speed for all values of the Courant number. For wave numbers which gives $u\Delta x/\pi > 0.5$, that is for waves of wavelengths $\lambda < 4\Delta x$ the slope of the curves indicates that the group velocity is negative. Thus for waves with wavelengths shorter than $4\Delta x$ the energy is propagating in the opposite direction of the waves.

6.1 Displayed is the cell structure of a staggered grid in one dimension. The circles are associated with $\phi$-points, while the horizontal bar is associated with a $u$-points within the same cells. The sketched staggering is such that the distance between adjacent $\phi$-points and $u$-points are one half grid distance apart.

8.1 Upper panel shows the Earth’s surface covered by a 2 degree mesh. Lower panel shows a similar mesh of 30 degrees mesh size. The figure conveniently illustrates how a 2 degree mesh in the ocean would look like in the atmosphere scaled by the Rossby radius of deformation.
8.2 Sketch of the mesh in the $t, x$ plane close to the right-hand open boundary. The computational domain is then to the left of $x = L$. The letters $J$, $J - 1$, and $J - 2$ denote grid points respectively at the open boundary, the first and second points inside the computational domain, while $n$, $n - 1$, and $n + 1$ denote the time levels.

8.3 Sketch of the FRS zone, the interior domain and the computational domain. Also shown are the appropriate indices.

9.1 Displayed are the two waves of wavelength $4\Delta x$ (solid curve) and $\frac{4}{3}\Delta x$ (dashed curve), in a grid of grid size $\Delta x$. Note that our grid cannot distinguish between the unresolved wave of wavelength $\frac{4}{3}\Delta x$ and the resolved wave of wavelength $4\Delta x$. Thus the energy contained in the unresolved wave will be folded into the low wavenumber space represented by the $4\Delta x$ wave.

9.2 The diagram illustrates the region of stability for the combined advection-diffusion equation approximated in (9.35). This corresponds to the area inside of the parabola (hatched area). The area inside the rectangular is where both the advection and the diffusion are stable individually. We notice that we obtain a more stringent stability condition to the advection equation when we are adding diffusion.
Chapter 1

INTRODUCTION

We concern ourselves with the fundamentals tools needed to understand how we put oceans and atmospheres on computers. In this we mostly concern ourselves with developing methods whereby some important balance equations in oceanography and meteorology, namely the advection-diffusion equation and a simplified form of the shallow water equations on a rotating earth, can be solved by numerical means. To this end we particularly make use of so called finite difference methods.

We assume that the reader has little or no prior knowledge of or experience in solving differential equations numerically. We therefore explain the methods in some detail. The advection-diffusion equation and the shallow water equations belongs to a class of equations known as partial differential equations. Consequently we include in the next chapter (Chapter 2) a rather detailed account on how various types of partial differential equations relates to the advection-diffusion equations and the shallow water equations.

To motivate the reader, and for later reference purposes, we first show how the advection-diffusion equation and the shallow water equations relates to the full equations governing the motion of the atmosphere and ocean. This necessitates a recapitulation of the governing equations, the boundary conditions and the basic approximations commonly made in meteorology and oceanography. We therefore end this introductory chapter by deriving the shallow water equations from the full governing equations, highlighting the necessary assumptions and approximations needed to arrive at these equations.

1.1 The governing equations

In the atmosphere and ocean the most prominent dependent variables are the three components $u, v,$ and $w$ of the three-dimensional velocity $\mathbf{v}$, pressure $p$, density $\rho$, (potential) and temperature $\theta$. For the atmosphere also humidity $q$ and cloud liquid water content $q_L$ must be included, while the salinity $s$ must be included among the prominent variables in the ocean. To determine

\footnote{Velocity is normally referred to as wind in the atmosphere and current in the ocean.}

\footnote{In the following bold upright fonts, e.g., $\mathbf{v}$, are used to denote a vector, while bold special italic fonts, e.g., $\mathbf{F}$, are used to denote tensors}
these unknowns we need an equal number of equations. These equations are normally referred to as the governing equations since they govern the motion of the two spheres atmosphere and ocean.

Of the variables above only the velocity is a vector. The remaining variables, commonly referred to as the state variables, are all scalars. The state variables, except for density and pressure, are all examples of what is referred to as tracers. Other examples of tracers are any dissolved chemical component or substance. Since the salinity, temperature and humidity influence the motion via the pressure forcing they are commonly referred to as active tracers. Tracers that do not influence the motion, like for instance dissolved chemical components, are referred to as passive tracers.

As is common when making a mathematical formulation of a physical problem, the governing equations are developed based on conservation principles, in our case the conservation of mass, momentum, internal energy and tracer content. For the atmosphere and ocean the governing equations in their non-Boussinesq form are\(^3\)

\[
\begin{align*}
\frac{\partial t}{\partial t} \rho + \nabla \cdot (\rho \mathbf{v}) &= 0, \\
\frac{\partial t}{\partial t} (\rho \mathbf{v}) + \nabla \cdot (\rho \mathbf{v} \mathbf{v}) &= -2\rho \Omega \times \mathbf{v} - \nabla p + \rho \mathbf{g} - \nabla \cdot (\rho \mathbf{F}_M), \\
\frac{\partial t}{\partial t} (\rho C_i) + \nabla \cdot (\rho C_i \mathbf{v}) &= -\nabla \cdot (\rho \mathbf{F}_i) + \rho S_i \quad i = 1, 2, \ldots, \\
\rho &= \rho(p, C_1, C_2, \ldots). 
\end{align*}
\]

where \(\nabla\) is the three-dimensional del-operator defined by

\[
\nabla = \mathbf{i} \partial_x + \mathbf{j} \partial_y + \mathbf{k} \partial_z,
\]

\(C_i\) represents the concentration of any tracer, the tensor \(\mathbf{F}_M\) and vector \(\mathbf{F}_i\) represents fluxes due to turbulent mixing of momentum and tracers, respectively, \(\Omega\) is the Earth’s rotation rate, \(\mathbf{g}\) is the gravitational acceleration and \(S_i\) is the tracer source, if any.

Equation (1.1) is the conservation of mass, while (1.2) constitutes the conservation of momentum. Equation (1.3) is the tracer conservation equation. The fourth and last (1.4) is the equation of state, relating density and pressure to the active tracers.

It should be noted that in the atmosphere the equation of state is linear and follows the ideal gas law, that is,

\[
p = \rho R \theta
\]

where \(R\) is the gas constant\(^4\). In contrast the equation of state for the ocean is highly non-linear and cannot be expressed in a formal, closed form. We may visualize the equation of state for the ocean in a so called temperature-salinity (\(T - S\)) diagram where the salinity \(s\) is drawn along the horizontal axis and the (potential) temperature \(\theta\) is drawn along the vertical axis. An example for \(p = 0\) is displayed in Figure 1.1.

---

\(^3\)See for example Gill (1982) or Griffies (2004)

\(^4\)\(R = 287.04\) Jkg\(^{-1}\)K\(^{-1}\)
1.2 Boundary conditions

We observe that to solve (1.1) - (1.4) we need to 1) specify conditions at the spatial boundaries of the domain and 2) specify the state of the ocean and/or atmosphere at some given time. The spatial boundary conditions consist of the dynamic and kinematic boundary conditions, while the state of the two spheres at a given time is called the initial condition.

Normally the bounding surfaces of the volume containing the ocean or the atmosphere consist of material surfaces. We recall that a material surface is a surface that consist of the same particles at all times. Thus the dynamic boundary condition at material surfaces requires that there is no acceleration at the surface, i.e., that the pressure and the fluxes must be continuous at such a surface. The kinematic boundary conditions at a material surface follows directly from the its definition, that is, a particle must stay at the surface.

As an example let us consider a system consisting of the atmosphere on top of the ocean.
Furthermore, let \( \eta = \eta(x, y, t) \) denote the deviation of the atmosphere/ocean interface away from its equilibrium level at \( z = 0 \), and let \( H = H(x, y) \) be the equilibrium depth of the ocean. Then the kinematic boundary condition at \( z = \eta \) is

\[
w = \partial_t \eta + \mathbf{u} \cdot \nabla_H \eta \quad \text{at} \quad z = \eta
\]

where \( \mathbf{u}, w \) are, respectively, the horizontal and vertical component of the three-dimensional velocity \( \mathbf{v} \), and where \( \nabla_H = i \partial_x + j \partial_y \) is the horizontal component of the three-dimensional del-operator (1.5). The kinematic boundary condition at \( z = -H \) is likewise

\[
w = -\mathbf{u} \cdot \nabla_H \cdot H \quad \text{at} \quad z = -H.
\]

Note that we assume that the “bottom” \( z = -H \) is a material surface, and hence (1.8) says that the bottom is impermeable or that there is no trough-flow across the bottom, that is, \( \mathbf{n} \cdot \mathbf{v} = 0 \), where \( \mathbf{n} \) is the unit vector perpendicular to the bottom.

### 1.3 The hydrostatic approximation

In the atmosphere and ocean the horizontal scales of the dominant motions are large compared to the vertical scale. As a consequence the vertical acceleration, \( Dw/dt \), is small when compared to, e.g., the gravitational acceleration \( \rho g \). Consequently we replace the vertical momentum equation by the hydrostatic equation in which the gravitational acceleration is balanced by the vertical pressure gradient. When one solves this reduced system the model is said to be hydrostatic, and the motion is said to satisfies the hydrostatic approximation.

To illustrate the approximation we first write the vertical component of the momentum equation using (1.2) in full, that is,

\[
\partial_t (\rho w) + \nabla \cdot (\rho \mathbf{v} w) = -\partial_z p - \rho g - \nabla \cdot (\rho \mathbf{F}^V_M),
\]

where \( \mathbf{F}^V_M \) is the vertical vector component of the mixing tensor \( \mathbf{F}_M \). In most cases, but not all, the vertical velocity and its associated acceleration terms \( \partial_t (\rho w) \) and \( \nabla \cdot (\rho w \mathbf{v}) \) are small compared to the gravitational acceleration and can safely be neglected. Furthermore, since the vertical motion is small compared to the horizontal motion also the friction term may be neglected. Under these circumstances the vertical momentum equation reduces to

\[
\partial_z p = -\rho g,
\]

which is the hydrostatic equation\(^5\). We are then left with the two horizontal components of the momentum equation, or

\[
\partial_t (\rho \mathbf{u}) + \nabla_H \cdot (\rho \mathbf{u} \mathbf{u}) + \partial_z (\rho w \mathbf{u}) + \rho f \mathbf{k} \times \mathbf{u} = -\nabla_H p + \partial_z \tau - \nabla_H \cdot (\rho \mathbf{F}^H_M),
\]

\(^5\)The name is used since a fluid at rest in the gravitational field satisfies exactly this equation. This is often referred to as a static fluid, hence the name hydrostatic.
where \( f = 2\Omega \sin \phi \) is the Coriolis parameter where \( \phi \) is the latitude and \( \Omega \) is the Earth’s rotation rate, \( \mathbf{F}_M^H \) is the horizontal tensor component of the three-dimensional flux tensor \( \mathbf{F}_M \) due to turbulent mixing, and \( \tau \) is the vertical shear stress.

The tracer equation is left unchanged, but as in the momentum equation we may single out the vertical acceleration term and thus write

\[
\partial_t (\rho \mathrm{C}_i) + \nabla_H \cdot (\rho \mathrm{C}_i \mathrm{u}) + \partial_z (\rho \mathrm{C}_i \mathrm{w}) = -\partial_z (\rho \mathrm{F}^V) - \nabla_H \cdot (\rho \mathbf{F}_i^H) + \rho \mathbf{S}_i \quad i = 1, 2, \cdots, (1.12)
\]

where \( \mathrm{F}^V \) and \( \mathbf{F}_i^H \) are respectively the vertical and horizontal components of the turbulent mixing.

### 1.4 The Boussinesq approximation

One common approximation employed in most ocean models is the so called Boussinesq approximation. We note that the fundamental basis for this approximation is the fact that the ocean in contrast to the atmosphere is nearly incompressible. This implies that any parcel of fluid conserves its volume. Thus for any parcel of fluid the change in density is small with respect to the density itself, that is,

\[
\frac{1}{\rho} \frac{D \rho}{dt} = \frac{D \ln \rho}{dt} \approx 0, \quad (1.13)
\]

where the operator \( \frac{D}{dt} \) is the material derivative\(^6\), defined by

\[
\frac{D}{dt} = \partial_t + \mathbf{v} \cdot \nabla. \quad (1.14)
\]

Under the Boussinesq approximation this approximation is taken as an equality. The mass conservation equation (1.1) then reduces to

\[
\nabla \cdot \mathbf{v} = 0. \quad (1.15)
\]

Use of an ocean model employing the Boussinesq approximation, a Boussinesq ocean, has one major disadvantage. One particularly pertinent example is the expected change in sea level, or volume, when the global ocean becomes warmer. When uniformly heating the ocean the equation of state implies that the density decreases. For a non-Boussinesq ocean, which is mass conserving, the response to the decrease in density is to expand its volume. Hence the sea level rises. In contrast for a Boussinesq ocean, which conserves volume, the response to the decrease in density is to loose mass. Obviously the latter is highly unrealistic.

The reason why the Boussinesq approximation is still widely used in the ocean modeling community, despite the Boussinesq fluid’s inability to expand due to heating, is the fact that we allow the hydrostatic pressure to change when the density changes. Thus under the Boussinesq approximation the density is treated as a constant except when it appears together with the gravitational acceleration. Another important aspect is that under the Boussinesq approximation

---

\(^6\)Also referred to as the individual derivative
1.5 The shallow water equations

A very common reduced set of equations in meteorology and oceanography is the so called shallow water equations. We may derive these equations from the full governing equation (1.1) - (1.4) by first making the hydrostatic and Boussinesq approximation. Hence the starting point is mass conservation in the form (1.15), the momentum equations in the form (1.10) and (1.16), and the tracer equation (1.17). In addition we assume that the density is uniform in time and space, i.e., $\rho = \rho_0$ where $\rho_0$ is a constant, which makes (1.4) obsolete. The resulting governing equations are then

$$\nabla_H \cdot \mathbf{u} + \partial_z w = 0,$$

(1.22)

$$\partial_t \mathbf{u} + \nabla_H \cdot (\mathbf{u} \mathbf{u}) + \partial_z (\mathbf{w} \mathbf{u}) = -f \mathbf{k} \times \mathbf{u} - \rho_0^{-1} \nabla_H p + \rho_0^{-1} \partial_z \tau - \nabla_H \cdot (\mathbf{F}_M^H), \quad (1.23)$$

$$\partial_z p = -\rho_0 g,$$

(1.24)

$$\partial_t C_i + \nabla_H \cdot (C_i \mathbf{u}) + \partial_z (C_i w) = -\partial_z F^V - \nabla_H \cdot (\mathbf{F}_i^H) + S_i \quad ; \quad i = 1, 2, \cdots \quad (1.25)$$
Note that the momentum equation is split in two; the horizontal component and the vertical component where the latter is the hydrostatic equation. We also observe that there are no dynamic active tracers present when we assume the density to be constant. Hence also the tracer equation (1.25) is in a sense obsolete. It is only there to enable transport and spreading of passive tracers, if any, but does not have any influence on the dynamics.

Integrating (1.22) and (1.23) from the bottom \( z = -H(x, y) \) to the top \( z = \eta(x, y, t) \), and using the kinematic boundary conditions (1.7) and (1.8) and the dynamic boundary condition \( p = 0 \) at \( z = \eta \) we get,

\[
\begin{align*}
\partial_t \mathbf{U} + \nabla_H \cdot \left( \frac{\mathbf{U} \mathbf{U}}{h} \right) + f \mathbf{k} \times \mathbf{U} &= -gh \nabla_H h + \rho_0^{-1}(\tau_s - \tau_b) + \mathbf{X}, \\
\partial_t h + \nabla_H \cdot \mathbf{U} &= 0,
\end{align*}
\]

where \( \mathbf{U} = \int_{-H}^{\eta} \mathbf{u} \, dz \) is the volume flux of fluid through the fluid column of height/depth \( h = \eta + H \), \( \tau_s \) and \( \tau_b \) are, respectively, the turbulent vertical momentum fluxes at the top and bottom of the fluid column, and \( \mathbf{X} \) is what is left of the horizontal momentum fluxes when integrated vertically from bottom to top. To arrive at (1.26) we have also integrated (1.24) from some arbitrary height/depth \( z \) to the top \( z = \eta \). Furthermore we have used the fact that \( H \) is independent of time to replace, e.g., \( \partial_t \eta \) by \( \partial_t h \). Finally we have absorbed the term arising from the approximation

\[
\int_{-H}^{\eta} \nabla_H \cdot (\mathbf{u} \mathbf{u}) \, dz \approx \nabla_H \cdot \left( \frac{\mathbf{U} \mathbf{U}}{h} \right)
\]

into the last term \( \mathbf{X} \) on the right-hand side of (1.26). We commonly refer to (1.26) and (1.27) as the shallow water equations. Written in the form (1.26) and (1.27) the shallow water equations are said to be written in flux form. We note that \( \mathbf{U} \) is the total volume flux of fluid through the fluid column of height/depth \( h \).

We note that if we define the mean depth average velocity by \( \hat{\mathbf{u}} = \mathbf{U}/h \) then the shallow water equations become

\[
\begin{align*}
\partial_t (h \hat{\mathbf{u}}) + \nabla_H \cdot (h \hat{\mathbf{u}} \hat{\mathbf{u}}) + f \mathbf{k} \times \hat{\mathbf{u}} &= -gh \nabla_H h + \rho_0^{-1}(\tau_s - \tau_b) + \mathbf{X}, \\
\partial_t h + \nabla_H \cdot (h \hat{\mathbf{u}}) &= 0,
\end{align*}
\]

For later reference purposes we note that the acceleration terms \( \partial_t (h \hat{\mathbf{u}}) + \nabla_H \cdot (h \hat{\mathbf{u}} \hat{\mathbf{u}}) \) in (1.29) can be written

\[
\begin{align*}
\partial_t (h \hat{\mathbf{u}}) + \nabla_H \cdot (h \hat{\mathbf{u}} \hat{\mathbf{u}}) &= h \left( \partial_t \hat{\mathbf{u}} + \hat{\mathbf{u}} \cdot \nabla_H \hat{\mathbf{u}} \right) + \hat{\mathbf{u}} \left[ \partial_t h + \nabla_H \cdot (h \hat{\mathbf{u}}) \right] \\
&= h \left( \partial_t \hat{\mathbf{u}} + \mathbf{u} \cdot \nabla_H \hat{\mathbf{u}} \right),
\end{align*}
\]

where we have used (1.30) to arrive at the last equal sign.

### 1.6 The quasi-geostrophic equations

Another common set of reduced equations are based on quasi-geostrophic theory as for instance detailed in Pedlosky (1979) or Stern (1975). Here we essentially follow Stern (1975).
We first note that the starting point for the quasi-geostrophic equations are the governing equations employing the hydrostatic and Boussinesq approximations. Without loss of generality we may therefore start with the shallow water equations (1.26) and (1.27). If we neglect the forcing terms on the right-hand side of (1.26) and make use of (1.31) we may write them in the following form

\[
\frac{D_H u}{dt} + fk \times u = -g \nabla_H h, \tag{1.32}
\]

\[
\frac{1}{h} \frac{D_H h}{dt} + \nabla_H \cdot u = 0, \tag{1.33}
\]

where \( \frac{D_H}{dt} \) is the two-dimensional version of the operator (1.14), that is,

\[
\frac{D_H}{dt} = \partial_t + u \cdot \nabla_H. \tag{1.34}
\]

If we in addition assume that the acceleration term \( \frac{D_H u}{dt} \) is small compared to the Coriolis term, we get

\[
fk \times u = -g \nabla_H h, \tag{1.35}
\]

which describes a balance between the Coriolis term and the pressure term. Equation (1.35) is the geostrophic equation and the balance is called the geostrophic balance. Under these circumstances we may solve for the relative velocity to get

\[
u = \frac{g}{f} k \times \nabla_H h. \tag{1.36}
\]

In those problems where the acceleration terms in (1.35) are small (but finite) compared to the Coriolis term we have

\[
R = \frac{|D_H u/dt|}{|fk \times u|} \ll 1, \tag{1.37}
\]

where \( R \) is the Rossby number. Thus the geostrophic equation is a first approximation to the relation between the velocity and the pressure field where terms of \( O(R) \) are neglected. We note that (1.35) obviously provides no information about the space-time variations in either the velocity field or the pressure field. Those dynamics must be obtained from the relevant asymptotic form of the vorticity equation.

To derive the vorticity equation we start by defining the relative vorticity

\[
\zeta = k \cdot \nabla_H \times u. \tag{1.38}
\]

Then operating \( k \cdot \nabla_H \times \) on (1.32) and then substituting for \( \nabla_H \cdot u \) from (1.33) we get

\[
\frac{D_H}{dt} \left( \frac{\zeta + f}{h} \right) = 0. \tag{1.39}
\]
Here $\zeta + f$ is the absolute vorticity, while $(\zeta + f)/h$ is the potential vorticity for a barotropic fluid. If $L$ is a typical lateral (horizontal) scale of $u$, so that $|u \cdot \nabla_H u| \sim |u|^2/L$, then the necessary condition for quasi geostrophy (cf. eq. 1.37) is

$$R = \frac{|u|}{fL} \ll 1. \quad (1.40)$$

The condition is however not sufficient since $\partial_t u$ might be comparable to the Coriolis term $fk \times u$. Consequently we must also require that the initial condition satisfies (1.35). The small $\partial_t u$ will then depend on the small $u \cdot \nabla_H u$, and the temporal evolution of the geostrophic field is computed from the asymptotic vorticity equation derived below.

Equation (1.37) and its equivalent (1.40) requires that the relative vorticity (1.38) is small compared to $f$ by a factor of $R$. We also note that the variation in layer thickness $h$, obtained from the integration of (1.35) are

$$h - H_m \sim \frac{fL}{g} |u| \quad \text{or} \quad \frac{h - H_m}{H_m} \sim RF^2, \quad (1.41)$$

where $H_m$ is the mean layer thickness and

$$F \equiv \left( \frac{L^2}{L_R} \right)^{\frac{1}{2}}, \quad (1.42)$$

where $L_R = gH_m/f^2$ is the Rossby radius of deformation. If we now assume $F \sim O(1)$ or less, which is tantamount to assuming that $L$ is not large compared to Rossby’s deformation radius, then the layer thickness variation in (1.41) is small to the same order as the ratio of the relative vorticity $\zeta$ to the planetary vorticity $f$, that is, $\zeta/f$.

Under these circumstances we first rewrite the potential vorticity equation (1.39) to get

$$\frac{D}{dt} \left( \frac{\zeta + f}{h} \right) = \frac{f}{H_m} (\partial_t + u \cdot \nabla_H) \left( \frac{1 + \frac{\zeta}{f}}{1 + \frac{h - H_m}{H_m}} \right) = 0. \quad (1.43)$$

We are now in a position to expand this expression in terms of $R$, and thus we get

$$(\partial_t + u \cdot \nabla_H) \left[ 1 + \frac{\zeta}{f} - \frac{h - H_m}{H_m} + O(R^2) \right] = (\partial_t + u \cdot \nabla_H) \left( \frac{\zeta}{f} - \frac{h}{H_m} \right) + O(R^3) = 0. \quad (1.44)$$

The leading terms in (1.44) are $O(R^2)$ since the (non-dimensional) magnitude of the acceleration terms $\partial_t u$ and $u \cdot \nabla_H u$ are $O(R)$. Thus the fractional error in the asymptotic vorticity equation

$$\left( \frac{\zeta}{f} - \frac{h}{H_m} \right) = 0 \quad (1.45)$$

Recall that we have assumed that the density is constant. The fluid is therefore barotropic. The potential vorticity may also be derived for a baroclinic fluid in a similar fashion, but has then a different mathematical expression.
and in the asymptotic momentum equation (1.35) are both of $\mathcal{O}(R)$. Hence if (1.36) be substituted for $u$ wherever the latter appears in (1.45), then the resulting differential equation for the layer thickness (or pressure) $h$ is also asymptotic when $R \ll 1$. It is thus permissible to evaluate the velocity and the relative vorticity in (1.45) using the geostrophic equation (1.35) or (1.36). In fact if we substitute the expression (1.36) for $u$ into (1.38) and then into (1.45) we first get

$$\zeta = \frac{g}{f} \nabla^2 h,$$  

(1.46)

and then

$$\left[ \partial_t + \frac{g}{f}(k \times \nabla_H h) \cdot \nabla_H \right] \left( \nabla_H^2 h - L_{-1}^{-1} h \right) = 0.$$  

(1.47)

Equation (1.46) and (1.47) together with the geostrophic equation (1.35) are commonly referred to as the quasi-geostrophic equations (QG equations). Thus we may use (1.47) to compute the pressure or layer thickness $h$ at an arbitrary time $t > 0$ from any initial distribution at time $t = 0$. The resulting solution is then almost geostrophic but not quite, hence the name quasi-geostrophic. We emphasize that it is only under very stringent conditions, as explained, that these equations are valid.

We finally remark that, although each step in the hierarchy of the approximations, that is the Boussinesq approximation, the hydrostatic approximation, the shallow water equations, and finally the quasi-geostrophic approximation, removes or filters out a certain class of phenomena, the advantage of such procedures is that they allow us to isolate effects having different space-time scales. In a numerical contexts they are also very useful in establishing solutions against which our numerical solutions may be tested or verified.
Chapter 2

PRELIMINARIES

The equations that governs the motion of the atmosphere and the ocean, as well as the hierarchy of equations that follows employing the various approximations as outlined in the introductory chapter (Chapter 1), belongs to a class of equations called partial differential equations (henceforth PDEs). They differ from ordinary differential equations in that there are more than one independent variable and sometimes also more than one dependent variable.

In this chapter we learn more about PDEs and reveal that they have different characters depending on the physics they describe. We also introduce some basic mathematics underlying two of the most important numerical methods used to solve atmospheric and oceanic problems, namely finite difference methods and spectral methods. These mathematics include knowledge about Taylor series expansions, orthogonal functions, Fourier series and Fourier transforms. Finally, we include some notations that conveniently helps us to solve PDEs using numerical methods.

2.1 General PDEs

In general a PDE is written

$$a\partial^2_{x'}\theta + 2b\partial_{x'}\partial_{y'}\theta + c\partial^2_{y'}\theta + 2d\partial_{x'}\theta + 2e\partial_{y'}\theta + f\theta = g.$$  \hspace{1cm} (2.1)

Here $\partial_{x'}, \partial_{y'}$ denotes differentiation with respect to the independent variables $x', y'$, while $\theta = \theta(x', y')$ denotes the dependent variable. The coefficients $a, b, \ldots, g$ are in general functions of the independent variables, that is, $a = a(x', y')$, etc. Note that $x'$ and $y'$ represents any independent variable, for instance time or one of the spatial variables, while $\theta$ represent any dependent variable, e.g., velocity, pressure, density, salinity, or humidity.

2.2 Elliptic equations

If $b^2 - ac < 0$ then the roots of (2.1) are imaginary, distinct, and complex conjugated. The corresponding PDE is then elliptic. The classic example is Poisson’s equation,

$$\nabla_H^2 \phi = \partial^2_x \phi + \partial^2_y \phi = g(x, y),$$  \hspace{1cm} (2.2)
2.3. PARABOLIC EQUATIONS

2.3. PARABOLIC EQUATIONS

where again \( \nabla_H \) is the two-dimensional part of the three-dimensional del operator. We arrive at this equation by letting \( x' = x \) and \( y' = y \) in (2.1), and by letting \( a = c = 1 \) and \( b = d = e = f = 0 \). Other examples are the Helmholtz equation

\[
\nabla^2_H \phi + f(x, y) \phi = g(x, y),
\]

(2.3)

and the Laplace equation

\[
\nabla^2_H \phi = 0.
\]

(2.4)

2.3 Parabolic equations

If \( b^2 - ac = 0 \) then the corresponding PDE is parabolic. The classic example is the diffusion equation or the heat conduction equation,

\[
\partial_t \theta = \kappa \partial^2_x \theta,
\]

(2.5)

where \( \kappa \) is the diffusion coefficient (heat capacity). To arrive at (2.5) from (2.1) we let \( u = \theta \), \( x' = t \), \( y' = x \), \( a = 1 \), \( b = 0 \), \( c = d = f = g = 0 \), and \( e = 1/2 \). Equation (2.5) is in fact a simplification of the full three-dimensional tracer equation (1.3) to one dimension where the advection term as well as the source and sink terms are neglected. In fact erasing source and sink terms and the advection term allow us to write the tracer equation for a Boussinesq fluid (1.17) as

\[
\partial_t C_i = \nabla \cdot (K \cdot \nabla C_i),
\]

(2.6)

where the diffusive tracer flux \( F_i \) is parameterized as \( F_i = -K \cdot \nabla C_i \) where in turn \( K \) is a matrix (dyade) describing the conductive efficiency of the medium with regard to the tracer \( C_i \) (cf. Section 3.1). Thus \( K = \kappa_{mn} i_m j_n \), \( m, n = 1, 2, 3 \). To retrieve (2.5) we simply let \( \kappa_{11} = \kappa \) and \( \kappa_{mn} = 0 \) for \( m \neq 1 \) and \( n \neq 1 \) and assume that \( \kappa \) is constant.

Let us for a moment assume that the atmosphere/ocean is at rest \( (v = 0) \) and that there are no sources for the tracer \( C_i \) \( (S_i = 0) \). Then (1.3) reduces to (2.6), which implies that also the diffusion balance is of fundamental importance when solving atmospheric and oceanographic problems.

2.4 Hyperbolic equations

If \( b^2 - ac > 0 \) then the roots of (2.1) are real and distinct. The corresponding PDE is then hyperbolic. The classic example is the wave equation,

\[
\partial^2_t \phi - c_0^2 \partial^2_x \phi = 0.
\]

(2.7)

To derive (2.7) from (2.1) we let \( u = \phi \), \( x' = t \), \( y' = x \), \( a = 1 \), \( b = 0 \), \( c = -c_0^2 \), and \( d = e = f = g = 0 \). Then \( b^2 - ac = 0 - (-c_0^2) = c_0^2 \) which indeed is positive.

We note that by defining

\[
\Phi = \partial_t \phi - c_0 \partial_x \phi
\]

(2.8)
we may rewrite the wave equation (2.7) to get
\[ \partial_t \Phi + c_0 \partial_x \Phi = 0. \] (2.9)
Since \( c_0 \) is a constant (2.9) may be written
\[ \partial_t \Phi + \partial_x (\Phi c_0) = 0. \] (2.10)

We observe that (2.10), commonly referred to as the *advection equation*, is a one-dimensional version of (1.1) with \( \rho \) replaced by \( \Phi \) and \( v \) replaced by \( c_0 i \). It is also a one-dimensional version of (1.3) with suitable replacements. Thus the advection equation is of fundamental importance in the modeling of atmospheric and oceanographic motions. It also indicates that the equations governing atmospheric and oceanographic motions viewed as time marching problems are inherently hyperbolic.

We also notice that the one-dimensional version of the shallow water equations (1.26) and (1.27) is inherently a hyperbolic problem. To illustrate this we first linearize (1.26) and (1.27). To this end we assume that the deviation of height of a fluid column is small compared to its equilibrium depth \( H \), that is, \( (h - H)/H \ll 1 \). We then get
\[
\begin{align*}
\partial_t u &= fv - g \partial_x h, \\
\partial_t v &= -fu - g \partial_y h, \\
\partial_t h &= -H \partial_x u - H \partial_y v.
\end{align*}
\] (2.11) \( \text{to} \) (2.13)

Here \( u, v \) are the components of the horizontal velocity \( u \) along the axes \( x, y \) respectively. Note that we for clarity neglect the vertical shear stress term as well the diffusive horizontal momentum flux term. We now manipulate (2.11) and (2.12) to find \( u, v \) as functions of \( h \). Next we substitute the results into (2.13) to get
\[ (\partial_t^2 + f^2 - gH \nabla^2_H) \partial_t h = 0. \] (2.14)

Let \( h = H + h' \) and let \( h' = 0 \) at time \( t = 0 \). Integration in time \( t \) then yields
\[ (\partial_t^2 + f^2 - gH \nabla^2_H) h' = 0. \] (2.15)

If we in addition assume that the motion is independent of one of the dependent variables, say \( y \), we get
\[ (\partial_t^2 + f^2 - gH \partial_y^2) h' = 0. \] (2.16)

We note that (2.15) is hyperbolic in \( t, x \) (and in \( t, y \)), but elliptic in \( x, y \). Thus, we note that although the steady state solution to (2.16) is elliptic, the time marching problem is inherently hyperbolic.

The governing equations describing the time evolution of atmospheric and oceanographic motion are fundamentally hyperbolic. It is important to keep this in mind when developing numerical methods to solve atmosphere-ocean problems.

We will return to the shallow water equations in Section 6.1 on page 75. There we use them as an example problem to show how multiple variable problems are solved using numerical methods.

---

\(^1\)To this end we first differentiate (2.11) with respect to time, and then add (2.12) multiplied by \( f \). This results in an equation containing \( h \) and \( u \) only. Similarly by first differentiate (2.12) with respect to time and then adding (2.11) multiplied by \( -f \) gives an expression relating \( h \) and \( v \).
2.5 Boundary conditions

As is well known the solution of any PDE contains integration constants. The number of integration constants is determined by the order of the PDE. For instance integrating the linearized shallow water equations (2.11) - (2.13) (or eq. 2.14) in time gives three integration constants, while integration in space gives another four integration constants (two in \( x \) and two in \( y \)) for a total of seven. Thus we need seven conditions to determine these constants. These conditions are commonly referred to as boundary conditions.

We emphasize that the number of boundary conditions needed are exactly the same as the number of integration constants, no more no less. If we specify too many boundary conditions the system is overspecified, and if we specify too few results we end up with an underspecified system. It is imperative that this is followed when we make use of numerical methods to solve our problems. The computer always produce numbers. If we over- or underspecify our system, the computer will still produce numbers that may even look realistic, but they are nevertheless incorrect. The reason is that the solution to any problem is equally dependent on the boundary conditions as on any other forcing.

To determine for instance the solution to the elliptic Poisson equation (2.2) we need four boundary conditions, two in \( x \) and two in \( y \). To determine the solution to the diffusion equation (2.5) we need three boundary conditions, two in \( x \) and one in time \( t \). Finally, to determine the solutions to the wave equation we need a total of four conditions to determine the four integration constants, namely two in \( t \) and two in \( x \). As we increase the dimensions of the equation we note that the number of integration constants increases and thus also the number of boundary conditions needed.

There are essentially two types of boundary conditions belonging to the class of natural boundary conditions, namely

- Dirichlet conditions,

in which case the variable is known at the boundary, and

- Neuman conditions,

in which case the derivative normal to the boundary is specified. Most other boundary conditions are just combinations of these. A natural boundary condition is one which is dictated by the physics of the problem.

As an example we note that the wind in the atmosphere or the current in the ocean cannot flow through an impermeable wall. Formulated mathematically this condition implies that

\[ \mathbf{n} \cdot \mathbf{v} = 0 \]  

(2.17)

at the wall, where \( \mathbf{n} \) denotes the unit vector perpendicular to the wall. This is also a classic example of a Dirichlet type boundary condition, which is tantamount to specifying the variable itself at the boundary (in this case no flow through the boundary).

Another example of a natural boundary is an insulated boundary. The natural condition dictated by the physics is that for the boundary to be insulated there can be no heat exchange
across the boundary, that is, the diffusive flux of heat through the boundary must be zero. In mathematical terms this is written

\[ \mathbf{n} \cdot \mathbf{F}_\theta = 0, \quad (2.18) \]

where \( \mathbf{F}_\theta = -\kappa \nabla \theta \) is the diffusive heat flux vector. Thus (2.18) is the same as specifying the gradient (in this case a zero gradient) at the boundary, which is the classic example of a Neuman condition.

As alluded to the two conditions may be combined to give other natural boundary conditions. One is the so called Cauchy condition or “slip” condition. For instance consider a flat bottom or surface at \( z = -H \) (or \( z = 0 \)) at which we give the following condition

\[ \nu \partial_z \mathbf{u} = C_D \mathbf{u} \quad ; \quad z = -H, \quad (2.19) \]

where \( \nu \) is the vertical eddy viscosity, \( \mathbf{u} \) is the horizontal component of the current (or wind), and \( C_D \) is a drag coefficient (more often than not the latter is a constant).

Other common boundary conditions are cyclic or periodic boundary conditions. A periodic boundary condition is one in which the solution is specified to be periodic in space, that is, that the solution repeats itself beyond a certain distance. Thus a periodic boundary condition in \( x \) for a given tracer concentration \( C'(x) \) would be

\[ C(x, t) = C(x + X, t), \quad (2.20) \]

where \( X \) is the distance over which the solution repeats itself. Such conditions are commonly in use when solving problems where the atmosphere or ocean is considered to be contained in a zonal channel bounded to the south an north by a zonal wall. In the longitudinal direction the solution is then dictated by physics to naturally repeat itself every 360 degrees.

### 2.6 Taylor expansions

The basis for all numerical finite difference methods is that all “good” functions can be expanded in terms of a Taylor series. A good function is simply one for which the function itself and all its derivatives exist and are continuous\(^2\). One characteristic of a good function is that it can always be expanded in a so called Taylor series. Another is that it can be represented by an infinite sum of orthogonal functions such as for instance trigonometric function (Sections 2.9 and 2.10).

Consider the function \( \theta(x, t) \) to be a good function. Then we may use a Taylor series expansion to find the values of \( \theta \) at \( x + \Delta x \) and \( x - \Delta x \), that is,

\[ \theta(x + \Delta x, t) = \theta(x, t) + \partial_x \theta(x, t) \Delta x + \frac{1}{2} \partial_x^2 \theta(x, t) \Delta x^2 + \frac{1}{6} \partial_x^3 \theta(x, t) \Delta x^3 + \mathcal{O}(\Delta x^4) \quad (2.21) \]

\[ \theta(x - \Delta x, t) = \theta(x, t) - \partial_x \theta(x, t) \Delta x + \frac{1}{2} \partial_x^2 \theta(x, t) \Delta x^2 - \frac{1}{6} \partial_x^3 \theta(x, t) \Delta x^3 + \mathcal{O}(\Delta x^4) \quad (2.22) \]

\(^2\)This definition is somewhat different from the one offered in the little known but enlightening book by M. J Lighthill entitled “Good functions”
By subtracting (2.22) from (2.21), and then do some suitable manipulation the first derivative of θ at the point \((x, t)\) in time and space may be written

\[
\partial_x \theta(x, t) = \frac{\theta(x + \Delta x, t) - \theta(x - \Delta x, t)}{2\Delta x} + O(\Delta x^2) \tag{2.23}
\]

We may also choose to solve (2.21) directly with respect to the first derivative. Then we obtain

\[
\partial_x \theta(x, t) = \frac{\theta(x + \Delta x, t) - \theta(x, t)}{\Delta x} + O(\Delta x) \tag{2.24}
\]

Expression (2.23) and expression (2.24) above may also be used to formulate possible finite difference approximations of the first derivative of \(\theta\) with respect to \(x\), that is,

\[
[\partial_x \theta]_{x,t} \approx \frac{\theta(x + \Delta x, t) - \theta(x - \Delta x, t)}{2\Delta x} \tag{2.25}
\]

\[
[\partial_x \theta]_{x,t} \approx \frac{\theta(x + \Delta x, t) - \theta(x, t)}{\Delta x} \tag{2.26}
\]

Instead of (2.21) we may also use (2.22) to formulate an approximation, that is,

\[
[\partial_x \theta]_{x,t} \approx \frac{\theta(x, t) - \theta(x - \Delta x, t)}{\Delta x} \tag{2.27}
\]

We note that while (2.25) is centered on the spatial point \(x\) (2.26) and (2.27) are one-sided. The approximation (2.25) is therefore denoted a centered approximation, while (2.26) and (2.27) are denoted a forward, one-sided approximation and a backward, one-sided approximation respectively, or simply forward and backward approximations.

We may perform exactly the same calculations based on Taylor series expansion to arrive at finite difference approximation for the derivatives in time \(t\). For instance by expanding \(\theta\) in time we get

\[
\theta(x, t + \Delta t) = \theta(x, t) + \partial_t \theta(x, t) \Delta t + \frac{1}{2} \partial_t^2 \theta(x, t) \Delta t^2 + \frac{1}{6} \partial_t^3 \theta(x, t) \Delta t^3 + O(\Delta t^4), \tag{2.28}
\]

\[
\theta(x, t - \Delta t) = \theta(x, t) - \partial_t \theta(x, t) \Delta t + \frac{1}{2} \partial_t^2 \theta(x, t) \Delta t^2 - \frac{1}{6} \partial_t^3 \theta(x, t) \Delta t^3 + O(\Delta t^4). \tag{2.29}
\]

To construct a centered finite difference approximation to the time rate of change of \(\theta\) we simply subtract (2.29) from (2.28) to obtain

\[
[\partial_t \theta]_{x,t} \approx \frac{\theta(x, t + \Delta t) - \theta(x, t - \Delta t)}{2\Delta t} \tag{2.30}
\]

Similarly we may construct approximations to higher order derivatives. For instance to find a centered finite difference approximation to the second order derivative of \(\theta\) with respect to \(x\) we first simply add the two Taylor expansion (2.21) and (2.22) to give

\[
\partial_x^2 \theta(x, t) = \frac{\theta(x + \Delta x, t) - 2\theta(x, t) + \theta(x - \Delta x, t)}{\Delta x^2} + O(\Delta x^2). \tag{2.31}
\]
Then by neglecting terms of higher order in (2.31) a finite difference approximation to the second order derivative is

\[
[\partial_x^2 \theta]_{x,t} \approx \frac{\theta(x + \Delta x, t) - 2\theta(x, t) + \theta(x - \Delta x, t)}{\Delta x^2}.
\] (2.32)

Since this expression gives equal weight to the points \(x + \Delta x\) and \(x - \Delta x\), that is, to the points on either side of \(x\), the approximation is centered. Like in (2.25) we note that the neglected terms are of \(O(\Delta x^2)\). This is in contrast to the forward and backward approximations in which the neglected terms where of \(O(\Delta x)\). Thus the centered approximations appear to share the fact that the neglected terms are of higher order than the one-sided approximations.

As exemplified in (2.30) we may formulate finite difference approximations to any higher order derivative with respect to \(t\), \(x\) and other spatial independent variables. It is for instance commonplace in contemporary model codes to formulate higher order approximations where the terms neglected in the Taylor series expansion are \(O(\Delta x^n)\) where \(n \geq 3\).

### 2.7 Truncation errors

As alluded to the main difference between the one-sided and centered difference approximations is the order of the terms neglected when making the approximation form the Taylor series expansion. While we neglected terms of \(O(\Delta x^2)\) when using the centered finite difference approximation, the terms we neglected when using the one-sided approximation was \(O(\Delta x)\). Thus the centered finite difference approximation is more accurate than the one-sided finite difference approximation. While the centered finite difference approximation has an error of second order, the one-sided finite difference approximation has an error of first order. Since the error is a direct consequence of neglecting higher order terms in a Taylor series expansion, we often refer to this error as the truncation error in that the series is truncated when making the finite difference approximation.

As shown in Exercise 3 we may also use the Taylor series expansion to construct finite difference approximations that are truncated to a higher order and thus are even more accurate. Such approximations are usually called higher order schemes or higher order finite difference approximations. We note from (2.63) that when constructing such approximations we have to include points that are a distance \(2\Delta x\) away from the point \(x\). Although we desire our approximations to be as accurate as possible we emphasize that higher order schemes have other potential complications associated with troubles at boundaries, higher order computational modes in space, and a more stringent instability criteria.

Finally, we emphasize that when the problem is multi-dimensional it is important that the finite difference approximation in all spatial directions are truncated to the same order. As an example consider a line wave propagating in space. The only way to ensure that the numerical solution then has the same accuracy regardless of the propagation direction of the wave is to use finite difference approximations that have the same accuracy along all axes.
2.8 Notations

When solving a PDE using numerical methods, and in particular finite difference methods, it is common to define a grid or mesh which covers the domain over which the solution is to be found. As an example let us consider a two-dimensional spatial problem for which we seek a solution within a quadratic domain where \( x, y \) both starts at 0 and ends at \( L \), e.g., the Laplace problem (2.4). We start by covering the domain by a quadratic mesh as displayed in Figure 2.1. We keep track of the grid points in the mesh by counting along the \( x \)-axis and the \( y \)-axis, respectively. Let us furthermore assume that there are \( J \) points along the \( x \)-axis and \( K \) points along the \( y \)-axis. To count the points we use dummy indices for instance \( j \) along the \( x \)-axis and \( k \) along the \( y \)-axis.

The point \( x = 0 \) along the \( x \)-axis is then associated with \( j = 1 \), while the point \( x = L \) along the \( x \)-axis is associated with \( j = J \). Similarly we associate the point \( y = 0 \) with \( k = 1 \) and the point \( y = L \) with \( k = K \). The \( j \)th point along the \( x \)-axis is then \( x = x_j \) where the subscript refers to the value for \( x \) at the \( j \)th point along the \( x \)-axis. Similarly we let \( y = y_k \) be associated with the \( k \)th point along the \( y \)-axis. The coordinates of the grid is then given by \( x_j, y_k \).

Let us denote the distance between two adjacent points along the \( x \)-axis by \( \Delta x \) and the distance between two adjacent points along the \( y \)-axis by \( \Delta y \). Then the \( j \)th point along the \( x \)-axis is denoted

\[
x_j = (j - 1)\Delta x,
\]

while the \( k \)th point along the \( y \)-axis is denoted

\[
y_k = (k - 1)\Delta y.
\]

We note in particular that \( x_1 = y_1 = 0 \) and that \( x_J = y_K = L \). We also notice for later convenient use that the latter gives

\[
\Delta x = L/(J - 1), \quad \Delta y = L/(K - 1),
\]

respectively\(^3\). It is also common to use the notation \( \theta_{jk} \) to denote the value of the variable \( \theta(x,y) \) at the grid point \( x_j, y_k \). Thus

\[
\theta_{jk} = \theta(x_j, y_k) = \theta[(j - 1)\Delta x, (k - 1)\Delta y].
\]

Furthermore follows that

\[
\theta_{jk} = \theta(x_j, y_k) = \theta[(j - 1)\Delta x, (k - 1)\Delta y]
\]

\[
\theta_{j-1k} = \theta(x_{j-1}, y_k) = \theta[(j - 2)\Delta x, (k - 1)\Delta y]
\]

\[
\theta_{jk+1} = \theta(x_j, y_{k+1}) = \theta[(j - 1)\Delta x, k\Delta y]
\]

To discriminate between spatial and temporal variables we will, as is common, hereon use a superscript when counting time. Thus

\[
t^n = n\Delta t, \quad n = 0, 1, 2, \cdots
\]

\(^3\)FORTRAN 90/95 allows us to use \( j = 0 \) and \( k = 0 \) as dummy counters. Under these circumstances \( x_j = j\Delta x \) and \( y_k = k\Delta y \). Thus \( x_0 = y_0 = 0 \) while \( x_J = y_K = L \) as before. Under this circumstance \( \Delta x = L/J \), and \( \Delta y = L/K \).
when counting time, where $\Delta t$ is the time step and $n$ is the time counter. Thus the variable $\theta(x, t)$ at the point $x_j, t^n$ in space and time is written

$$\theta^n_j = \theta(x_j, t^n) = \theta[(j - 1)\Delta x, n\Delta t] \quad (2.41)$$

We note that when the variable is four-dimensional the notation we use is

$$\theta^n_{jkl} = \theta(x_j, y_k, z_l, t^n) \quad (2.42)$$

where $z_l = (l - 1)\Delta z$. 

Figure 2.1: Displayed is a commonly used grid when employing numerical methods to solve PDEs. The points in the $x, y$ directions are incremented by $\Delta x, \Delta y$, respectively, so that there are a total of $J$ points along the $x$-axis and $K$ points along the $y$-axis. The points are counted by using the dummy counters $j, k$. 

We note that when the variable is four-dimensional the notation we use is
As an example let us consider the Taylor series expansions (2.21) and (2.22). Using the preceding notation we find

\[
\theta^n_{j+1} = \theta^n_j + [\partial_x \theta]_j^n \Delta x + \frac{1}{2} [\partial_x^2 \theta]_j^n \Delta x^2 + \frac{1}{6} [\partial_x^3 \theta]_j^n \Delta x^3 + O(\Delta x^4)
\]

(2.43)

\[
\theta^n_{j-1} = \theta^n_j - [\partial_x \theta]_j^n \Delta x + \frac{1}{2} [\partial_x^2 \theta]_j^n \Delta x^2 - \frac{1}{6} [\partial_x^3 \theta]_j^n \Delta x^3 + O(\Delta x^4)
\]

(2.44)

and hence that the forward in space, finite difference approximations to the first derivative is written

\[
[\partial_x \theta]^n_j \approx \frac{\theta^n_{j+1} - \theta^n_j}{\Delta x},
\]

(2.45)

while the second order, centered approximation is written

\[
[\partial_x \theta]^n_j \approx \frac{\theta^n_{j+1} - \theta^n_{j-1}}{2\Delta x}.
\]

(2.46)

Similarly follows that the second order, centered finite difference approximation to the second derivative with the above notation is written

\[
[\partial_x^2 \theta]^n_j \approx \frac{\theta^n_{j+1} - 2\theta^n_j + \theta^n_{j-1}}{\Delta x^2}.
\]

(2.47)

Finally we remark that the increments \(\Delta x, \Delta y, \Delta z\) and \(\Delta t\) do not have to be constant, but may be allowed to vary in space and even in time. If the increments vary in space only we refer to the grid as an unstructured mesh. If the increments vary in both time and space we refer to the grid as an adaptive unstructured mesh.

### 2.9 Orthogonal functions

Note that when using finite difference techniques for time dependent or evolutionary problems, we only consider grid-point values of the dependent variables; no assumption is made about how the variables behave between grid points. An alternative approach is to expand the dependent variables in terms of a finite series of smooth orthogonal functions. The problem is then reduced to solving a set of ordinary differential equations which determine the behavior in time of the expansion coefficients.

As an example consider the general linear one-dimensional time dependent problem

\[
\partial_t \phi = \mathcal{H}[\phi] \quad \text{for} \quad x \in \langle -L, L \rangle \quad \text{and} \quad t > 0
\]

(2.48)

where \(\phi = \phi(x, t)\) is a good function as defined in Section 2.6 and \(\mathcal{H}\) is a linear differential operator in \(x\). Note that to solve (2.48) we have to specify suitable boundary conditions at \(x = \pm L\) and initial condition at \(t = 0\). Here we will simply assume that the condition at \(x = \pm L\)
is that $\phi$ is cyclic and that the initial value is known. Since $\phi$ is a good function it may be expanded in terms of an infinite set of orthogonal functions $e_n(x)^4$, where $n = 1, 2, 3, \ldots$. Thus

$$\phi = \sum_{n=-\infty}^{\infty} \varphi_n(t)e_n(x),$$

(2.49)

where $\varphi_n(t)$ are the expansion coefficients$^5$. Without loss of generality we may assume that the expansion functions $e_n(x)$ are orthonormal so that

$$\int_{-L}^{L} e_n(x)e^*_m(x)dx = \begin{cases} 1 & n = m \\ 0 & n \neq m \end{cases},$$

(2.50)

where $e^*_n(x)$ is the complex conjugate of $e_n(x)$. Considering that we know the expansion functions $e_n(x)$, it is the expansion coefficients $\varphi_n(t)$ whose behavior we want to determine. To this end we first multiply (2.48) by $e^*_m$, and then integrate over all possible $x$-values, to give

$$\int_{-L}^{L} \partial_t \phi(x,t)e^*_m(x)dx = \int_{-L}^{L} \mathcal{H}[\phi]e^*_m(x)dx.$$  

(2.51)

The left-hand side is further developed by use of (2.49) and (2.50) to give

$$\int_{-L}^{L} \partial_t \phi e^*_m dx = \int_{-L}^{L} \left( \sum_n \partial_t \varphi_n e_n \right) e^*_m dx = \sum_n \partial_t \varphi_n \int_{-L}^{L} e_n e^*_m dx = \sum_n \partial_t \varphi_n.$$  

(2.52)

Since the operator $\mathcal{H}$ only operates on $x$ follows in addition that

$$\mathcal{H}[\phi] = \sum_m \varphi_m \mathcal{H}[e_m].$$

(2.53)

Using these results we get

$$\partial_t \varphi_n = \sum_m \varphi_m \int_{-L}^{L} \mathcal{H}[e_m]e^*_n dx ; \quad \forall m.$$  

(2.54)

We thus have a set of coupled, ordinary differential equations for the time rate of change for the expansion coefficients $\varphi_n$.

It is now interesting to consider how our choice of expansion functions can greatly simplify the problem

1. If the expansion functions are eigenfunctions of $\mathcal{H}$, we have $\mathcal{H}[e_m] = \lambda_m e_m$ where $\lambda_m$ are the eigenvalues. Equation (2.54) then becomes

$$\partial_t \varphi_m = \lambda_m \varphi_m ; \quad \forall m.$$  

(2.55)

and the equations becomes decoupled.

$^4$Note that the expansion functions $e_n(x)$ are in general complex functions, e.g., $e_n(x) = e^{i\alpha_n x}$ where $\alpha_n$ are the wavenumbers.

$^5$Generally this method is used to separate variables when solving differential equations involving more than one independent variable.
2. If the original equation is
\[ G[\partial_t \phi] = H[\phi] \] (2.56)
where \( G \) is a linear operator, then our problem is simplified by using expansion functions that are eigenfunctions of \( G \) with eigenvalues \( \lambda_n \). We then have,
\[ \lambda_n \partial_t \varphi_n = \sum_n \varphi_m \int_{-L}^L H[e_m] e_n^* dx. \] (2.57)

### 2.10 Fourier series

A much used orthogonal set of expansion functions are the trigonometric functions \( e^{i\alpha_n x} \) where \( \alpha_n \) are an infinite number of discrete wavenumbers\(^6\). Thus any good function \( \phi(x, t) \) may be written
\[ \phi(x, t) = \frac{1}{2\pi} \sum_{n=-\infty}^{\infty} \varphi_n(t; \alpha_n) e^{i\alpha_n x} \] (2.58)
where the factor \( 1/2\pi \) is present for later convenience. The series (2.58) is called a Fourier series and the expression
\[ \varphi_n(t)e^{i\alpha_n x} \] (2.59)
is called a Fourier component. We note that the complex conjugate to the expansion functions are \( e^{-i\alpha_n x} \), and hence the Fourier series may be written
\[ \phi(x, t) = \phi_0 + \frac{1}{2\pi} \sum_{n=1}^{\infty} \varphi_n(t; \alpha_n) e^{i\alpha_n x}. \] (2.60)

It is important to realize that the subscript \( n \) attached to the expansion coefficients implies that they are different for each wavenumber, and hence depends on the wavenumber \( \alpha_n \) as well as time.

### 2.11 Fourier transforms

Finally, let us assume that the function \( \phi \) depends on \( x \) only. Under these circumstances we may define a function \( \hat{\phi} \) such that
\[ \phi(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{\phi}(\alpha) e^{-i\alpha x} d\alpha. \] (2.61)

We observe that the “expansion coefficients” \( \hat{\phi} \) now are continuous functions of the wavenumber \( \alpha \in [-\infty, +\infty] \) and that the summation is replaced by an integral. Commonly the function \( \hat{\phi} \)

\(^6\)In the above problem with cyclic boundary conditions at \( x = \pm L \) the wavenumbers are \( \alpha_n = n\pi/L \).
is referred to as the Fourier transform of the original function $\phi$. In fact we can show that the Fourier transform is given by the simple formula

$$\hat{\phi}(\alpha) = \int_{-\infty}^{\infty} \phi(x) e^{i\alpha x} dx.$$  

(2.62)

Hence we may plot $\hat{\phi}$ as a function of $\alpha$. The space spanned by $\hat{\phi}$ and $\alpha$ is called the Fourier space.

As revealed by (2.61) and (2.62) the Fourier transform is simply the amplitude associated with each particular wavenumber (or wavelength) $\alpha$. In a sense it reveals how much “energy” that is associated with the each wavelength. Plotting the Fourier transform in Fourier space therefore reveals information on how much energy each wavelength contains. The waves with wavelengths having the highest amplitudes are also the wavelengths that contain the highest energy content. Knowing the Fourier transform thus reveals information about the wavelengths that dominates the motion. This information is important regarding the construction of the grid, particularly the size of the spatial increments to choose (cf. Figure 2.1). If we intend to resolve the dominant portion of the motion we must choose the increments so that we have enough points per wavelength to resolve it. Ideally we should have 10 points per wavelength. As a minimum we must require that the size of the increments are such that we have 4 points per wavelength.

Finally we remark that our functions $\phi$ above are in general complex functions, while the functions we are concerned with are real functions. We therefore emphasize that our real function is recovered by extracting the real part of, e.g., $\phi$.

**Exercises**

1. Show that both the Helmholtz and the Laplace equations are elliptic in $x$ and $y$.

2. Show that the diffusion equation is parabolic in $t, x$ and $t, y$, but elliptic in $x, y$.

3. Show by use of Taylor series expansions that a possible finite difference approximation of $\partial_x \theta(x)$ with a truncation error of $O(\Delta x^4)$ is

$$[\partial_x \theta]_j \approx \frac{4}{3} \frac{\theta_{j+1} - \theta_{j-1}}{2\Delta x} - \frac{1}{3} \frac{\theta_{j+2} - \theta_{j-2}}{4\Delta x}.$$  

(2.63)

Note that to obtain higher order truncation errors we have to use points that are distances $2\Delta x$ away from the point $x_j$ itself.
Chapter 3

TIME MARCHING PROBLEMS

Most of the problems in the atmospheric and oceanographic sciences involve solving a time marching problem. Typically, we know the state of the atmosphere or the ocean at one specific time. Our task in numerical weather prediction (NWP) and numerical ocean weather prediction (NOWP) is then to use the governing equations of Section 1.1 on page 1 to find the state of the sphere in question at some later time. Such problems are known as initial value problems in mathematics.

A particular balance inherent in our governing equations, as displayed in (1.1) - (1.4), is a balance between the time rate of change of a tracer or variable in response to advective and diffusive fluxes. As the name indicates the problem is a combination of two different physical processes. The first is associated with advection, in which case the underlying PDE is hyperbolic. The second is associated with diffusion, in which case the underlying PDE is parabolic.

The advection and diffusion problems and their combination are of fundamental importance in meteorology and oceanography. Knowledge on how to solve these simple equations by numerical means is a “must” for everyone who aspires to become a meteorologist and/or oceanographer. In the following Chapters 4 and 5 we will therefore give insight into solving respectively the diffusion problem and the advection problem by use of numerical methods. In Chapter 9 (Section 9.3 on page 117) we also give insight into how to solve the combined advection-diffusion problem. We maintain that it is of fundamental importance to obtain knowledge on how to treat these terms numerically correct. At the same time these relatively simple problems conveniently serves the purpose of introducing some of the basic concepts needed to solve atmospheric and oceanographic problems employing numerical methods. They also nicely serves the purpose of illustrating some of the pitfalls.

Before venturing into details (Chapters 4 and 5) we first highlight some physical properties peculiar to each of the two processes. The motivation is that these important properties must be retained in any numerical solutions, or else the solution must be discarded as being false or incorrect. Thus to check the behavior of the solutions against these fundamental properties is part of what is often referred to as model verification\(^1\). When coding errors are thus found we

\(^1\)Model verification is the first step in a chain of activities commonly referred to as model quality assurance procedures (Anon, 1991; Lynch and Davies, 1995; Hackett et al., 1995)
3.1 The advection-diffusion equation

We now focus on the tracer conservation equation for a Boussinesq fluid. We start by recalling that the tracer equation for a Boussinesq fluid is given by (1.17). Neglecting possible tracer sources ($S_i = 0$) we get

$$\partial_t \theta + \nabla \cdot F_A + \nabla \cdot F_D = 0,$$

(3.1)

where $\theta$ is any dependent variable (or tracer), for instance potential temperature, $F_A$ is the advection flux vector and $F_D$ is the diffusive flux vector. If $\theta$ is the potential temperature then (3.1) is the conservation equation for internal energy or heat content except for the neglect of source terms.

Although their appearance in (3.1) are quite similar, we recall that the two flux vectors represent two quite different physical processes. The advective flux vector represents processes that transport the property of the tracer from one place to another via the motion. In contrast the diffusive flux vector represents processes in which the property $\theta$ is transferred from one location to another by turbulent mixing. While the former thus changes the tracer via the motion the latter changes the tracer via small scale, inherently chaotic processes that causes properties to be exchanged between two locations without invoking any mean motion. Thus it is a process similar to conduction. We therefore refer to the second term on the left-hand side of (3.1) as being the advective term, while we refer to the third term as being the diffusive term, hence the subscripts $A$ and $D$, respectively.

Since the two flux vectors represent two very contrasting physical processes, they naturally have very different mathematical formulations or parametrization. While the parametrization of the advective flux vector is

$$F_A = v \theta,$$

(3.2)

the formulation of the diffusive flux vector may take various forms. The most common one, called down the gradient diffusion, is simply

$$F_D = -\kappa \nabla \theta,$$

(3.3)

where $\kappa$ is the diffusion coefficient or conductive capacity. Equation (3.3) expresses that the larger the gradient (or difference) the larger the diffusive flux and hence the more effective diffusion is to decrease any differences in the tracer $\theta$ over small distances.

If we for a moment neglect the diffusive part of (3.1) we are left with a balance between the time rate of change of the tracer concentration and the divergence of the advective flux vector. We recognize this balance as the “wave equation” (cf. eq. 2.9 on page 13). Solving this equation is consequently often referred to as solving the advection problem. If we next for a moment

---

2Debugging simply means to weed out all errors in the program code.

3Its original formulation is due to a Dr. Adolf Eugen Fick who in 1855 formulated the parametrization (3.3) which is now referred to as Fickian diffusion.
neglect the advection part of (3.1) the time rate of change of the tracer concentration is balanced by the diffusion. Using the parametrization (3.3) for the diffusive flux, we are left with a typical parabolic problem (cf. eq. 2.5 on page 12). The resulting equation is called the diffusion equation, and solving it is referred to as solving the diffusion problem.

### 3.2 Diffusion

As alluded to above one property of the diffusive processes is that it even out small scale differences in the tracer fields. Note that it also even out noise created for instance by our choice of numerical methods when solving the equation numerically. We may show that the diffusion equation indeed has this property by analyzing the square of the tracer concentration. To arrive at an equation for the “variance” $\theta^2$ we first multiply the diffusion equation

$$\partial_t \theta = -\nabla \cdot F_D$$

by the tracer concentration $\theta$ itself to obtain

$$\partial_t \theta^2 = -2\theta \nabla \cdot F_D.$$  

(3.5)

The left hand side of (3.5) is the time rate of change of the variance. Let us assume that (3.4) and by implication (3.5) are valid within a volume $V$ bounded by a surface $\Omega$. Then we get the time rate of change of the variance by integrating (3.5) over the total volume $V$, or

$$\partial_t \int_V \theta^2 dV = -2 \int_{\Omega} \theta F_D \cdot \delta\sigma + 2 \int_V F_D \cdot \nabla \theta dV.$$  

(3.6)

Here the vector $\delta\sigma = n\delta\sigma$ where $n$ is a unit vector directed along the outward normal to the surface $\Omega$ and $\delta\sigma$ is an infinitesimal surface element. To derive (3.6) we also used the Gauss theorem. At the boundary $\Omega$ we must specify a boundary condition. We simply assume that the condition is either a Dirichlet or a Neuman condition. In the former case the condition is $\theta = 0$, while in the latter case the condition is $n \cdot F_D = 0$. In either case we observe that the first term on the right-hand side of (3.6) is zero. Hence (3.6) reduces to

$$\partial_t \int_V \theta^2 dV = 2 \int_V F_D \cdot \nabla \theta dV.$$  

(3.7)

If

$$F_D \cdot \nabla \theta \leq 0$$  

(3.8)

the right-hand side of (3.7) is negative. Hence we get

$$\partial_t \int_V \theta^2 dV \leq 0,$$  

(3.9)

\[4\] The square of the tracer concentration, or $\theta^2$, is a measure of the variance of the tracer concentration. In turn the variance is a measure of how noisy a tracer field is.
which shows that as long as (3.8) is satisfied then the diffusion term acts to even out any noise in the $\theta$ field. We notice that (3.8) is always satisfied as long as the diffusive flux vector $F_D$ is directed opposite to $\nabla \theta$. Under these circumstances we refer to the parametrization of the diffusive flux vector as being down the gradient. We recall from (3.3) that in the case of Fickian diffusion $F_D = -\kappa \nabla \theta$. Hence

$$F_D \cdot \nabla \theta = -\kappa (\nabla \theta)^2 \leq 0,$$

which reveals that Fickian diffusion is indeed down the gradient and thus always tends to even out any noise in our solution.

From this we make two observations. First diffusion always acts to damp out the variance in any solutions. Second if our numerical methods used to solve an atmospheric or oceanographic problem that does not include diffusion contains numerical diffusion (cf. Section 4.2) it will imply an artificial damping of the solution.

In this regard it is worthwhile to underscore that most problems in oceanography and meteorology are non-linear. While there is no transfer of energy from one wavelength to the next in a linear system, this is not true for a non-linear problems. In such systems energy input on long wavelengths (small wave numbers) is always in the end transferred to progressively shorter wavelengths (high wave numbers). This fact was described elegantly in the following rhyme credited to G. I. Taylor$^5$:

“Big whirls have smaller whirls that feed on their velocity, and little whirls have lesser whirls, and so on to viscosity.... in the molecular sense.”

However, when making the finite difference approximations to our PDEs the wavelengths that we resolve is limited by the specified spatial increments, say $2\Delta x$, often referred to as the Nyquist wavelength (or frequency in the time domain). Thus as the energy is cascading downwards toward shorter wavelengths we must, in our numerical solutions, mimic this process across the Nyquist wavelength to wavelengths which are not resolved by our grid. Since diffusion has the property of damping differences it is one tool at hand that may prove useful to handle what is known as non-linear instability (cf. Section 9.2 on page 113).

### 3.3 Advection

As we alluded to in Chapter 3.1 the advection problem is a balance between the first and second term on the left hand-side of (3.1), that is,

$$\partial_t \theta + \nabla \cdot F_A = 0,$$

$^5$Geoffrey Ingram Taylor (1886 - 1975) made fundamental contributions to turbulence, championing the need for developing a statistical theory, and performing the first measurements of the effective diffusivity and viscosity of the atmosphere. He is commonly remembered as the namesake for several basic fluid flow instabilities (Taylor - Couette, Rayleigh - Taylor, and Saffman - Taylor).
where $\theta$ is any tracer variable and $\mathbf{F}_A$ is the advective flux vector. As for the diffusion problem we are looking for solutions within a limited volume $V$ in space bounded by the surface $\Omega$, and for all times $t \in [0, \infty]^6$. On the surface $\Omega$ the equations are replaced by the boundary conditions, while the initial condition replaces the equations at time $t = 0$.

Let the advective flux be parametrized by the common parametrization $\mathbf{F}_A = \mathbf{u} \theta$, and let the boundary condition at the surface $\Omega$ be such that $\mathbf{F}_A \cdot \delta \sigma = 0\,^7$. Then by performing the same operation as in Section 3.2 we find that the total variance becomes

$$
\partial_t \int_V \theta^2 dV = 2 \int_V \mathbf{F}_A \cdot \nabla \theta dV = \int_V \mathbf{u} \cdot \nabla \theta^2 dV = - \int_V \theta^2 \nabla \cdot \mathbf{u} dV. \quad (3.12)
$$

Thus the total variance may increase or decrease depending on the sign of the velocity divergence. If the sum of the divergence is positive then the variance will decrease, while if it is negative then the variance will increase. The case $\nabla \cdot \mathbf{u} = 0$ is special. In this case the right hand-side of (3.12) is zero and hence any disturbances creating a variance in $\theta$ will just prevail, that is, the total variance is conserved.

As mentioned in Section 1.4 the Boussinesq ocean is to a good approximation divergence free due to its incompressibility (see also Gill, 1982, side 85). Thus in the ocean the advection process does not lead to any decrease or increase in the property being advected. Hence any disturbance generated in a limited domain may be advected to other locations undisturbed. This is not true for the atmosphere since the atmosphere is highly compressible. Thus in limited area where the divergence is positive ($\nabla \cdot \mathbf{u} > 0$), that is, the individual fluid parcels are drawn apart, any disturbance in the total tracer variance will be smoothed. In contrast the disturbances tends to increase in areas where $\nabla \cdot \mathbf{u} < 0$.

Finally we emphasize that the properties outlined above regarding the advection are important to retain when solving the advective problem by numerical means. In particular we stress that when the fluid is divergence free, like the ocean, then the total variance should be conserved. We also note that this is in stark contrast to the diffusion problem where all down the gradient diffusive fluxes give a decrease in the total tracer variance.

---

6In practice we have to limit the computation to a finite time span

7This is achieved by assuming $\mathbf{u} = 0$ or $\mathbf{u} \cdot \delta \sigma = 0$, that is, no flow across the boundary.
Chapter 4

THE DIFFUSION PROBLEM

In this chapter we present finite difference methods whereby the diffusion equation can be solved by numerical means. We first consider the diffusion equation in its simplest form. Hence we assume a one-dimensional diffusion process in space. Additionally we assume that the diffusive flux can be parametrized as a down the gradient diffusion and that the diffusion (mixing) coefficient is uniform in time and space. Under these circumstances the diffusion equation (3.4) in its continuous form reads,

$$\frac{\partial \theta}{\partial t} = \kappa \frac{\partial^2 \theta}{\partial x^2},$$  \hspace{1cm} (4.1)

where $\theta$ can be any variable, e.g., potential temperature, density, velocity components, and where $\kappa$ is the constant diffusion coefficient.

As alluded to in Section 2.3, we note that the PDE (4.1) is parabolic in nature. Thus the physical characteristic of the problem is that properties are transferred from one location to the next by conduction. We emphasize that this is very different from hyperbolic type problems, e.g., advection problems. The diffusion process acts simply to even out differences without dissipation. If we for instance start with a very narrow tracer distribution (cf. Figure 5.3 on page 62) diffusion acts to transfer these high values to adjacent locations at the expense of the peak value as time is marching on. Thus as time passes the peak is diminished while the values at adjacent locations increases. If we allow the diffusion process to go on forever within an infinite domain the tracer values becomes infinitely small, but will cover the whole infinite domain. In summary, the diffusion process transfers properties from one location to the next by conduction that acts to diminish peaks in a distribution so that the end result is a much smoother field.

An obvious example of a diffusion process is heat conduction (or turbulent mixing) in both atmospheres and oceans. Then $\theta$ is the potential temperature and $x$ any of the independent variables in space. Another classic atmosphere-ocean example of a diffusion problem is the so called Ekman problem, which in the atmosphere explains how the velocity is reduced in the planetary boundary layer due to friction at the surface. In the ocean the Ekman problem explains how the momentum due to surface traction is transferred downwards in the water column.
4.1 Finite difference approximations

Our concern is to find a numerical solution to (4.1) for all times for a given computational domain or area, e.g., \( x \in [0, L] \). At the two space boundaries and for the initial time \( t = 0 \) (4.1) is replaced by the respective boundary conditions. Let us for instance consider heat conduction in the atmosphere and ocean. Let \( x \) be the height (or depth) coordinate, and let \( \theta \) describe the potential temperature deviation away from a given mean temperature profile. We then assume that we know the anomaly at time \( t = 0 \), that is, \( \theta(x, 0) = \theta_0(x) \). Our task is then to find, by numerically solving (4.1), how the anomaly evolves in time between the two bounding heights \( x = 0, L \). By considering that \( \theta = 0 \) at \( x = 0 \) and \( x = L \) we imply that the boundary condition is a Dirichlet condition, that is, that the value of the dependent variable is fixed for all times. We also assume that the initial anomaly is different from the trivial solution \( \theta = 0 \), that is, \( \theta_0 \neq 0 \).

To find a numerical solution to (4.1) we follow the notation in Section 2.8. Thus we first divide the intervals \( x \in (0, L) \) and \( t \in (0, T) \), where \( T \) is some finite time, into respectively \( J \) and \( N \) sections of width \( \Delta x \) and \( \Delta t \). They then form a grid whose grid points are located at \( (x_j, t^n) \) where \( x_j = (j - 1)\Delta x \) and \( t^n = n\Delta t \), and where \( j \in [1, J + 1] \) and \( n \in [0, N] \) are counters such that \( x_{j+1} = L \) and \( t^N = T \) (see Figure 4.1).

Next we must define a finite difference approximation to the derivatives \( \partial_t \theta \) and \( \partial_x^2 \theta \) at the grid points. Using a forward in time approximation to express \( \partial_t \theta_j^n \) and a centered in space approximation to express \( \partial_x^2 \theta_j^n \) it follows from Section 2.6 that

\[
[\partial_t \theta]^n_j = \frac{\theta^{n+1}_j - \theta^n_j}{\Delta t}, \quad [\partial_x^2 \theta]^n_j = \frac{\theta^n_{j+1} - 2\theta^n_j + \theta^n_{j-1}}{\Delta x^2}.
\]

By substituting the expressions (4.2) into (4.1) we get

\[
\frac{\theta^{n+1}_j - \theta^n_j}{\Delta t} = \frac{\theta^n_{j-1} - 2\theta^n_j + \theta^n_{j+1}}{\Delta x^2}.
\]

Solving with respect to \( \theta^{n+1}_j \) gives

\[
\theta^{n+1}_j = \theta^n_j + \frac{\kappa \Delta t}{\Delta x^2} \left( \theta^n_{j-1} - 2\theta^n_j + \theta^n_{j+1} \right) ; \quad \begin{cases} j = 2(1)J \\ n = 0(1)N - 1 \end{cases}
\]

Note that (4.4) is valid for \( j = 2(1)J \) and for \( n = 0(1)N - 1 \) only. At the boundaries \( j = 1 \) and \( j = J + 1 \) and for \( n = 0 \) the boundary and initial conditions prevail.

To find \( \theta \) at the first time level \( n = 1 \) we substitute \( n = 0 \) into (4.4) and get

\[
\theta^1_j = \theta^0_j + \frac{\kappa \Delta t}{\Delta x^2} \left( \theta^0_{j-1} - 2\theta^0_j + \theta^0_{j+1} \right) ; \quad j = 2(1)J.
\]

Thus for the first “wet” point \( j = 2 \) we get

\[
\theta^1_2 = \theta^0_2 + \frac{\kappa \Delta t}{\Delta x^2} \left( \theta^0_1 - 2\theta^0_2 + \theta^0_3 \right).
\]
CHAPTER 4. THE DIFFUSION PROBLEM

4.1. FINITE DIFFERENCE FORM

Figure 4.1: Displayed is the employed grid we use to solve (4.1) by numerical means. The grid points in the $x, t$ directions are incremented by $\Delta x, \Delta t$, respectively. There is a total of $J + 1$ points along the $x$-axis and $N + 1$ points along the $t$-axis, counted by using the dummy indices $j, n$. The coordinates of the grid points are $x_j = (j - 1)\Delta x$ and $t^n = n\Delta t$, respectively.

All the $\theta^0_j$s on the right-hand side of (4.6) are known from the boundary and/or initial conditions including $j = 1$. We may then proceed to evaluate $\theta^1_1, \theta^1_2, \cdots$ up to and including $\theta^1_J$. For $j = J$ in particular, the last wet point, we get

$$\theta^1_J = \theta^0_J + \frac{\kappa \Delta t}{\Delta x^2} \left( \theta^0_{J-1} - 2\theta^0_J + \theta^0_{J+1} \right).$$

(4.7)

We note that again all the $\theta^0_j$s on the right-hand side are known from the initial and/or boundary conditions, including the value at $j = J + 1$, that is, $\theta^0_{J+1}$. This procedure thus provides values for $\theta^1_j$ at all the interior grid points. Note that at the boundaries $j = 1, J + 1 \theta$ is known from the boundary condition, that is $\theta^1_1 = 0$ and $\theta^1_{J+1} = 0$. This reflects the well known property of differential equations, whether they are PDEs of ordinary (ODEs), namely that they are valid only in the interior of a domain. At the boundaries (whether in time or space) the equations are replaced by the boundary condition. Thus (4.5) together with the boundary conditions gives us $\theta$ for all $j = 1(1)J$ at time level $n = 1$. 

31
We may then proceed to compute $\theta$ at time level $n = 2$ by substituting $n = 1$ into (4.4) to get

$$\theta_j^2 = \theta_j^1 + \frac{\kappa \Delta t}{\Delta x^2} (\theta_{j-1}^1 - 2\theta_j^1 + \theta_{j+1}^1) \quad ; \quad j = 2(1)J.$$ (4.8)

Having thus found $\theta$ at time level $n = 2$ we may proceed to time level $n = 3$ and so on for all time levels $n$ up to and including $n = N - 1$.

We note that since $x_{J+1} = L = J\Delta x$ we cannot choose $J$, $L$ and $\Delta x$ independently. Once two of them are chosen the third is given by the formula

$$J = \frac{L}{\Delta x},$$ (4.9)

Likewise follows that

$$N = \frac{T}{\Delta t}$$ (4.10)

showing that $N$, $\Delta t$ and $T$ also depend on each other.

Since we know $\theta$ at time $t = 0$ (or time level $n = 0$), the application of a forward, one-sided finite difference approximation in time, as for instance employed in (4.4), is the obvious choice. The accuracy of this scheme is thus $O(\Delta t)$, while the application of a centered finite difference approximation in space makes the spatial accuracy higher ($O(\Delta x^2)$). To increase the accuracy to the same level in time we may for instance employ a centered in time scheme for the time rate of change as well. We then get

$$[\partial_t \theta]_j^n = \frac{\theta_{j}^{n+1} - \theta_{j}^{n-1}}{2\Delta t},$$ (4.11)

in which case the finite difference approximation to (4.1) is

$$\frac{\theta_{j}^{n+1} - \theta_{j}^{n-1}}{2\Delta t} = \frac{\kappa \theta_{j-1}^n - 2\theta_j^n + \theta_{j+1}^n}{\Delta x^2} \quad ; \quad \{ j = 2(1)J \} \quad n = 0(1)N - 1,$$ (4.12)

or

$$\theta_{j}^{n+1} = \theta_{j}^{n-1} + \frac{2\Delta t}{\Delta x^2} (\theta_{j-1}^{n} - 2\theta_j^{n} + \theta_{j+1}^{n}) \quad ; \quad \{ j = 2(1)J \} \quad n = 0(1)N - 1.$$ (4.13)

To obtain the solution at the first time level $n = 1$, that is, to obtain $\theta_j^1$ we again substitute $n = 0$ into (4.13) which gives

$$\theta_j^1 = \theta_j^{-1} + \frac{2\Delta t}{\Delta x^2} (\theta_{j-1}^{0} - 2\theta_j^{1} + \theta_{j+1}^{0}) \quad ; \quad j = 2(1)J.$$ (4.14)

We must then require knowledge of $\theta_j^{-1}$. This corresponds to get knowledge of the potential temperature anomaly at a time $t < 0$, in this case at one time level prior to the first time level. By using the one-sided forward scheme we avoid this problem, but sacrifice accuracy. As shown in the next Sections 4.2 - 4.5 there are, however, more pressing needs that makes us shy away from using a centered in time, centered in space finite difference approximation to solve the diffusion equation numerically.
4.2 Numerical stability

Another major problem with the scheme (4.13) is that it is numerically unstable. This means that the numerical solution, instead of following the continuous solution, steadily deviates from it. Commonly this happens explosively just like an analytic instability (think of baroclinic and barotropic instabilities in the atmosphere and ocean). We therefore call this behavior numerical instability to distinguish it from the analytic instabilities that we rather like to simulate using our numerical model. For our numerical solution to have any legitimacy we must require that it is numerically stable. This is an absolute requirement and is formulated as follows:

A numerical scheme is stable if and only if the numerical solution is limited within any given finite time span

As a prelude to how we analyze the numerical scheme with respect to its numerical stability, let us first consider the analytic solution to (4.1). To this end we note that any good function \( \theta \) may be written as a sum of cosines and sines or even more compact as a sum of exponentials (see for instance Section 2.10 on page 22 or Lighthill, 1970, page 3)

\[
\theta(x, t) = \sum_{\alpha = -\infty}^{\infty} \Theta_\alpha(t) e^{i\alpha x} = \sum_{\alpha = -\infty}^{\infty} \theta_\alpha = \theta(x,t)
\]  

(4.15)

where \( \alpha \) is the wavenumber. Each component in (4.15), that is,

\[
\theta_\alpha = \Theta(t)e^{i\alpha x},
\]

(4.16)

is called a Fourier component, where \( \Theta(t) \) is an unknown time dependent amplitude. By substituting this component into (4.1) we obtain

\[
\partial_t \Theta = -\kappa \alpha^2 \Theta.
\]

(4.17)

which is an ordinary differential equation (ODE). Solving it with respect to \( \Theta \) gives

\[
\Theta = \Theta_0 e^{-\kappa \alpha^2 t}
\]

(4.18)

where \( \Theta_0 \) is the initial amplitude at time \( t = 0 \). Thus we find that the analytic solution to (4.1) is

\[
\theta(x, t) = \sum_{\alpha = -\infty}^{\infty} \Theta_0 e^{-\kappa \alpha^2 t} e^{i\alpha x}.
\]

(4.19)

We note that the amplitude of all the Fourier components decreases monotonically and exponentially as time increases. Moreover, we note that the part of the solution associated with the shortest waves (highest wave numbers) decreases faster than the part of the solution associated with the longer waves (low wave numbers). This is in accord with Section 3.2 where we concluded, based on (3.9), that diffusion acts to smooth out disturbances. What we additional learn from (4.19) is that this smoothing is selective in the sense that small scale disturbances are smoothed
fast while the longer periods are less prone to damping in the same time period. Thus diffusion acts like a filter efficiently smoothing the small scale noise, if any, without significantly damping the longer period motion.

As is obvious we would like the numerical solution in general to behave accordingly. In particular we expect the numerical solution to the diffusion equation to decrease monotonically in time. Thus if the numerical solution increases in time it is obviously wrong and possibly unstable. Note that this eventual instability has nothing to do with accuracy of the chosen scheme. Yet it is the initial truncation error inherent in our scheme that is allowed to grow uncontrolled when the solution is unstable. We will return to this in Section 4.3 below.

In mathematical terms the requirement of numerical stability is formulated by stating that for any finite time \( t = T > 0 \) there must exist a finite number, say \( B \), such that

\[
\left| \frac{\theta}{\theta_0} \right| \leq B, \quad (4.20)
\]

where \( \theta_0 \) is the value of dependent variable at time \( t = 0 \). For linear systems, and to certain degree also non-linear systems, it is possible to analyze the stability of the chosen scheme analytically. This should always be done before implementing the chosen scheme on the computer.

### 4.3 Stability analysis: von Neumann’s method

One such method is the so called von Neumann’s method. To analyze the stability von Neumann suggested to use a method somewhat similar to solving the equations analytically. The first step is to define a discrete Fourier component similar to the analytic one given in (4.16), that is,

\[
\theta_j^n = \Theta_n e^{i\alpha j \Delta x}, \quad (4.21)
\]

where \( \Theta_n \) is the discrete amplitude at time level \( n \) and \( \alpha \) is the wavenumber. We now define a growth factor \( G \) by

\[
G \equiv \frac{\Theta_{n+1}}{\Theta_n} \Rightarrow \Theta_{n+1} = G \Theta_n \quad \text{and} \quad \Theta_{n-1} = G^{-1} \Theta_n. \quad (4.22)
\]

Thus \( G \) gives the amplification of the amplitude \( \Theta \) as we proceed from one time level to the next. We observe that (4.22) is formally similar to (4.20), except that the growth factor \( G \) is defined as the ratio between the next and the former time level, that is, between time level \( n + 1 \) and time level \( n \), while (4.20) is the ratio between the value at a random time level and the initial value. Letting \( n = 0 \) in (4.22) then gives

\[
\Theta_1 = G \Theta_0 \quad (4.23)
\]

where \( \Theta_0 \) is the initial amplitude. By letting \( n = 1 \) in (4.22) and making use of (4.23) we obtain

\[
\Theta_2 = G \Theta_1 = G^2 \Theta_0 \quad (4.24)
\]

Continuing by letting \( n = 3, 4, \ldots \) up to a random number \( m \) then gives

\[
\Theta_m = G^m \Theta_0 \quad (4.25)
\]
Thus $G^m$ is the ratio between the amplitude at the random time level $m$ or random time $t = m\Delta t$ and the initial amplitude. Thus (4.20) is satisfied if

$$|G| \leq 1$$

(4.26)
since then $G^m$ decreases as the time level or time increases\(^1\). The criterion (4.26) is called *von Neumann’s condition for stability*. Note that it is a sufficient condition, not a necessary condition. We return to this in Section 4.4 below.

As our first example we use von Neumann's method to analyze the forward in time, centered in space scheme for the diffusion as given by (4.4). Thus we first substitute (4.21) into (4.4) to obtain

$$\Theta_{n+1} = \Theta_n + \frac{\kappa\Delta t}{\Delta x^2} (e^{i\alpha \Delta x} - 2 + e^{-i\alpha \Delta x}) \Theta_n$$

(4.27)

where the common factor $e^{i\alpha j \Delta x}$ is removed. Noting that $e^{i\alpha \Delta x} + e^{-i\alpha \Delta x} = 2 \cos \alpha \Delta x$ we get

$$\Theta_{n+1} = \left[ 1 - 2 \frac{\kappa\Delta t}{\Delta x^2} (1 - \cos \alpha \Delta x) \right] \Theta_n.$$  

(4.28)

We thus find the growth factor by simply dividing (4.28) by $\Theta_n$, 

$$G = 1 - 2 \frac{\kappa\Delta t}{\Delta x^2} (1 - \cos \alpha \Delta x).$$

(4.29)

To satisfy (4.26) we observe that

$$-1 \leq 1 - 2 \frac{\kappa\Delta t}{\Delta x^2} (1 - \cos \alpha \Delta x) \leq 1.$$  

(4.30)

Since $0 \leq (1 - \cos \alpha \Delta x) \leq 2$ the right-hand side inequality is satisfied for all $\Delta t$ and $\Delta x$. The inequality on the left-hand side, however, is satisfied if and only if

$$\frac{\kappa\Delta t}{\Delta x^2} (1 - \cos \alpha \Delta x) \leq 1.$$  

(4.31)

Recall that $0 \leq (1 - \cos \alpha \Delta x) \leq 2$, and hence (4.31) is satisfied for all wavenumbers $\alpha$ if

$$\frac{\kappa\Delta t}{\Delta x^2} \leq \frac{1}{2} \quad \text{or} \quad \Delta t \leq \frac{\Delta x^2}{2\kappa}.$$  

(4.32)

This condition also ensures that (4.30) is satisfied, and hence that von Neumann’s condition (4.26) is satisfied as well. Furthermore (4.32) tells us that we cannot choose $\Delta x$ and $\Delta t$ independently. Once $\Delta x$ is chosen the time step $\Delta t$ must be chosen in accord with (4.32). We therefore say that the forward in time, centered in space scheme (4.4) is *conditionally stable* under the condition (4.32).

We also observe from (4.31) that the waves that first violate the inequality are waves with wavenumbers given by

$$\cos \alpha \Delta x = -1$$

(4.33)

---

1 Confer Computer Problem No. 1
corresponding to those waves that maximizes \(1 - \cos \alpha \Delta x\). The solutions to (4.33) are wavenumbers \(\alpha_m\) given by
\[
\alpha_m \Delta x = (2m - 1)\pi; \quad m = 1, 2, \ldots
\]  
with corresponding wavelengths
\[
\lambda_m = \frac{2\pi}{\alpha_m} = \frac{2\Delta x}{2m - 1}.
\]  
The most dominant of these waves is the wave corresponding to \(m = 1\). Thus the most unstable wave has wavelength
\[
\lambda_1 = 2\Delta x.
\]  
This implies that the numerical instability will appear as “\(2\Delta x\)” noise, that is, noise of wavelength \(2\Delta x\), and commonly as a saw tooth pattern as displayed in Figure 4.2. (cf. Computer Problem No. 2).

Figure 4.2: Displayed are solutions of the diffusion equation using the scheme (4.4) for respectively \(K = \kappa \Delta t / \Delta x^2 = 0.45\) (left panel) and \(K = 0.55\) (right panel). The solutions are shown for the time levels \(n = 0\), \(n = 50\) and \(n = 90\). Note the saw tooth like pattern in the right panel for \(n = 90\) not present in the left panel. This indicates that the stability condition (4.32) is violated for \(K = 0.55\), but not for \(K = 0.45\).

We mentioned that the forward in time, centered in space scheme (4.4) for the diffusion equation is a conditionally stable scheme. If no such condition can be found that makes the chosen FD approximation satisfies the von Neumann’s condition (4.26), then we say that the scheme is \textit{unconditionally unstable}. If von Neumann’s condition is satisfied regardless of our choice of spatial and temporal increments (e.g., \(\Delta x\) and \(\Delta t\)), then we say that the scheme is \textit{unconditionally stable}. If the special case \(|G| = 1\) is true then we in addition say that the scheme is \textit{neutrally stable}.
It is worthwhile mentioning that when \( |G| < 1 \) it follows from (4.22) that \( |\Theta_{n+1}| < |\Theta_n| \). Thus, inherent for all schemes for which \( |G| < 1 \) is that they include artificial numerical energy dissipation\(^2\). We emphasize that even if the physical problem does not exhibit energy dissipation the numerical solution may exhibit such a decrease. We therefore refer to this artificial energy dissipation as numerical dissipation. We note that this dissipation depends on the absolute value of the growth factor and hence by implication on our choice of scheme and spatial and temporal increments. It is therefore of importance to ensure that the numerical dissipation is as small as possible by making choices so that the absolute value of the growth factor is as close to one as possible.

For problems that include natural energy dissipation it is therefore important to ensure that the numerical energy dissipation is small compared to the physical dissipation. We therefore always favor neutral schemes (\( |G| = 1 \)), since such schemes are energy conserving, a highly desirably property. If this is not possible we recommend to choose the time step and the space increments so as to minimize the numerical energy dissipation. This is the same as requiring \( |G| \) to be as close to one as possible. Regarding the forward in time, centered in space scheme this implies that we have to choose a time step \( \Delta t \) that is small enough to satisfy (4.32), but at the same time is large enough to make \( \Delta t \approx \Delta x^2 / 2\kappa \).

### 4.4 The necessary stability condition

We mentioned above that von Neumann’s condition is a sufficient condition. This implies that if (4.26) is satisfied then the scheme is definitively stable. The question is if it is too strict, that is, if it is also the necessary condition?

To determine this we return to the original requirement as formulated in (4.20). Substituting the discrete Fourier component (4.21) into (4.20) then gives

\[
\frac{|\Theta_{n+1}|}{\Theta_0} \leq B \quad \Rightarrow |G|^n \leq B. \tag{4.37}
\]

Taking the natural logarithmic on both sides then gives

\[
n \ln |G| \leq \ln B \equiv B'. \tag{4.38}\]

Even if von Neumann’s condition is too strict \( |G| \) cannot be very much larger than one. Thus we may write \( |G| = 1 + \epsilon \) where \( \epsilon \) is a small (\( \epsilon \ll 1 \)) positive number. Hence it follows that \( \ln |G| = \ln(1 + \epsilon) \approx \epsilon \). Furthermore we note that \( t^n = n\Delta t \) or that at \( n = t^n / \Delta t \). When we substitute these expressions into (4.38) we obtain

\[
\epsilon \leq \frac{B'\Delta t}{t^n} = O(\Delta t). \tag{4.39}
\]

Thus the necessary condition that satisfies the numerical stability requirement is

\[
|G| \leq 1 + O(\Delta t). \tag{4.40}
\]

\(^2\)In this context energy dissipation means that the amplitude of the solution decreases in time.
This shows that von Neumann’s condition (4.26) is indeed too strict. However, most physical problems, even those containing instabilities, always involve some physical energy dissipation. Thus for all practical purposes we may apply the sufficient condition $|G| \leq 1$ when analyzing the numerical stability of our schemes, in particular if $|G| \lesssim 1$.

Finally, we remark that the growth factor $G$ associated with the one-dimensional diffusion equation, as displayed by (4.29), is a scalar. For multi-variable and multi-dimensional problems the growth factor will commonly be a tensor or matrix, say $\mathcal{G}$. The sufficient condition is then that its spectral radius is less than or equal to one. This is tantamount to requiring that the largest eigenvalue of $\mathcal{G}$ is less than or equal to one.

### 4.5 Explicit and implicit schemes

The schemes (4.4) and (4.13) both contain the variable at time level $n + 1$ on the left-hand side. In addition all the terms on the right-hand side are evaluated at the time level $n$ or earlier ($n - 1, n - 2, \ldots$). We refer to such schemes as being explicit. In contrast, if the right-hand side includes variables evaluated at the new time level $n + 1$ we refer to the scheme as being implicit. Furthermore if we treat a multi-variable problem, e.g., the shallow water equations, where some of the terms are treated as being explicit and some implicit we commonly refer the scheme to as being semi-implicit.

Explicit schemes, as exemplified by (4.4) and (4.13), are always relatively simple to solve. Once the unknowns are known for one time level at all grid points, the computation of the next time level is straightforward, we just proceed from one grid point to the next. This is in most cases not true for implicit and semi-implicit schemes.

Whether a scheme is explicit or implicit also impacts the stability of the scheme. Let us for instance consider the centered in time, centered in space scheme applied to the diffusion equation (4.1). If we use an explicit scheme we then evaluate all the terms on the right hand-side at the time level $n$. Accordingly the scheme becomes the one displayed in (4.13). If we choose to make the scheme implicit we would then evaluate all the terms on the right-hand side at the new time level $n + 1$. The scheme then becomes

$$\theta_j^{n+1} = \theta_j^{n-1} + \frac{2\Delta t}{\Delta x^2} \left( \theta_{j-1}^{n+1} - 2\theta_j^{n+1} + \theta_{j+1}^{n+1} \right) ; \begin{cases} j = 2(1)J \\ n = 0(1) \ldots \end{cases}$$

(4.41)

Let us first analyze the stability of the explicit scheme using von Neumann’s method. Substitution of the Fourier component (4.21) into (4.13), removing the common factor $e^{i\alpha j \Delta x}$ and using (4.22) then gives

$$G = G^{-1} - \frac{4\Delta t}{\Delta x^2} (1 - \cos \alpha \Delta x).$$

(4.42)

Multiplying by $G$ and rearranging terms yields the equation

$$G^2 + 2\lambda G - 1 = 0,$$

(4.43)

where

$$\lambda = \frac{2\Delta t}{\Delta x^2} (1 - \cos \alpha \Delta x) \geq 0,$$

(4.44)
to determine the growth factor. Solving (4.42) with respect to we get the two solutions

\[ G_{1,2} = -\lambda \pm \sqrt{1 + \lambda^2}. \] (4.45)

We recall that in order to be numerically stable both solutions must satisfy von Neumann’s condition. We observe that

\[ |G_2| = \lambda + \sqrt{1 + \lambda^2} \geq 1, \] (4.46)

and hence that the centered in time, centered in space explicit scheme for the diffusion equation is unconditionally unstable. Thus:

**Never use a centered in time, centered in space scheme for the diffusion problem. It is always unconditionally unstable.**

Then what about the implicit scheme (4.41)? To analyze the stability of the scheme (4.41) we first rearrange the terms to obtain

\[ \theta_{j-1}^{n+1} - \left( 2 + \frac{\Delta x^2}{2\kappa\Delta t} \right) \theta_j^{n+1} + \theta_{j+1}^{n+1} = -\frac{\Delta x^2}{2\kappa\Delta t} \theta_j^{n-1} ; \quad \begin{cases} j = 2(1)J \\ n = 0(1) \ldots \end{cases} \] (4.47)

We note in passing that the implicit formulation require us to solve for \( \theta \) at time level \( n + 1 \) at the three grid points \( j - 1, j, \) and \( j + 1 \) simultaneously. It may however be efficiently solved employing an elliptic solver as shown below in Section 4.8. Curiously enough the implicit formulation of the parabolic diffusion equation turns it into an elliptic numerical equation.

To determine the growth factor we again employ von Neumann’s method and substitute the Fourier component into (4.47) to give

\[ G \left[ 2 \cos \alpha \Delta x - \left( 2 + \frac{\Delta x^2}{2\kappa\Delta t} \right) \right] = -\frac{\Delta x^2}{2\kappa\Delta t} G^{-1}. \] (4.48)

Multiplying by \( \frac{2\kappa\Delta t}{\Delta x^2} G \) and solving with respect to the growth factor \( G \) then gives

\[ |G_1| = |G_2| = \frac{1}{\sqrt{1 + \frac{4\kappa\Delta t}{\Delta x^2}(1 - \cos \alpha \Delta x)}} \] (4.49)

Note that \( |G_{1,2}| \leq 1 \) for all \( \Delta t \) and \( \Delta x \). Hence the implicit formulation of the centered in time centered in space scheme is unconditionally stable. This property is shared by all implicit and semi-implicit schemes; they are always unconditionally stable. In contrast we have just shown that the explicit formulation gave an unconditionally unstable scheme.

We furthermore observe that the implicit formulation (4.41) does not contain any constraint on the time step \( \Delta t \), and hence we may choose \( \Delta t \) to be as long as we wish. However, we also observe that the growth factor \( |G| \) is less than one for all wavelengths longer than \( 2\Delta x \), and that the it decreases with increasing time step. Thus the implicit formulation always contains numerical dissertation. Furthermore it increases with increasing choice of the time step. Again this is a property shared by all implicit schemes. Although implicit schemes are unconditionally
4.6. DUFORT-FRANKEL CHAPTER 4. THE DIFFUSION PROBLEM

stable, we strongly advice to choose a time step \( \Delta t \) so as to bring the growth factor as close to one as possible. The rationale is that we must control the numerical dissipation inherent in all implicit schemes. Thus although implicit schemes have no restrictions on the length of the time step its length is nevertheless constrained by the need to keep the numerical dissipation as small as possible.

4.6 Consistency and inconsistency: DuFort-Frankel

One important aspect about the schemes we use to solve our governing equations numerically are that they should mimic their continuous counterpart. We must therefore require that the employed scheme converges toward the continuous equations in the limit when \( \Delta t \to 0 \) and the space increments \( \Delta x, \Delta y, \Delta z \to 0 \) independently. Note that this requirement is independent on how they go to zero and independent of how fast each of them goes to zero. If this requirement is met we say that the scheme is consistent. If not we say that the scheme is inconsistent. Together with the absolute requirement of numerical stability it forms the two fundamental properties that our schemes should obey.

As is obvious all schemes where the finite difference approximations are based on Taylor series expansions as outlined in Section 2.6 satisfies the consistency requirement. Since both of the schemes (4.4) and (4.13) are based on a Taylor series expansion, they are both prime examples of consistent schemes.

We may, however, quite easily construct numerical schemes without having to resort to Taylor series expansions. It is particular in cases when the scheme is not constructed based on Taylor series expansions that we need to analyze its consistency. If the scheme turns out to be inconsistent we must require as a minimum that the dominant physical processes that the governing equations simulate are consistently represented in the finite difference analogue. One example of such a scheme elaborated below is the Dufort-Frankel scheme.

To construct the Dufort-Frankel scheme we start with the consistent centered in time, centered in space scheme (4.13). As shown above (Section 4.5) this scheme is unconditionally unstable when applied to the diffusion equation. However, in many cases the diffusion term is added just to even out small scale noise, that is, to dissipate energy that accumulates on the smaller scales. In these cases the term does not represent the dominant physics and we may relax on our consistency requirement. To construct the scheme we observe that the value of \( \theta_j^n \) at the grid point \( x_j, t^n \) in space and time can be thought of as a linear interpolation in time of the two adjacent grid points \( x_j, t^{n+1} \) and \( j, n - 1 \), or

\[
\theta_j^n = \frac{1}{2} (\theta_j^{n+1} + \theta_j^{n-1}).
\]  
(4.50)

Using this as a substitute for \( \theta_j^n \) on the right-hand side of (4.13) we obtain

\[
\theta_j^{n+1} = \theta_j^{n-1} + 2 \frac{K \Delta t}{\Delta x^2} (\theta_j^{n-1} - \theta_j^{n-1} - \theta_j^{n+1} + \theta_j^{n+1}) ; \quad \left\{ \begin{array}{l}
j = 2(1)J \\
n = 0(1)N - 1.
\end{array} \right.
\]  
(4.51)

We first note that the introduction of the term \( \theta_j^{n+1} \) on the right-hand side makes the new scheme implicit. As all implicit schemes we therefore expect it to be stable. Curiously enough, by simply
adding some implicit to the centered in time, centered in space scheme we have turned it into an unconditionally stable scheme (see Exercise 4 on page 48). We note that in contrast to the implicit scheme (4.41), the implicity is now limited to the single term \( \theta_{j}^{n+1} \) involving only the space grid point \( x_j \). We may move this term from the right-hand side of (4.51) to its left-hand side. After some rewriting we get

\[
\theta_{j}^{n+1} = \left[ \theta_{j}^{n-1} + \chi (\theta_{j-1}^{n} - \theta_{j}^{n} + \theta_{j+1}^{n}) \right] (1 + \chi)^{-1}. \tag{4.52}
\]

where

\[
\chi = \frac{2 \kappa \Delta t}{\Delta x^2}. \tag{4.53}
\]

Thus the numerical algorithm associated with the implicit DuFort-Frankel scheme (4.51) is solved explicitly. This is one reason why it has become so popular, in particular in oceanography, (e.g., Adamec and O’Brien, 1978). A second reason, mentioned in the previous paragraph, is that the scheme is implicit and thus, like all implicit schemes, is unconditionally stable.

It remains to analyze the consistency of the scheme. To this end we employ the Taylor series expansions of Section 2.6,

\[
\theta_{j}^{n \pm 1} = \theta_{j}^{n} \pm [\partial_x \theta]_{j}^{n} \Delta x + \frac{1}{2} [\partial_x^2 \theta]_{j}^{n} \Delta x^2 \pm \frac{1}{6} [\partial_x^3 \theta]_{j}^{n} \Delta x^3 + \mathcal{O}(\Delta x^4), \tag{4.54}
\]

and

\[
\theta_{j}^{n \pm 1} = \theta_{j}^{n} \pm [\partial_t \theta]_{j}^{n} \Delta t + \frac{1}{2} [\partial_t^2 \theta]_{j}^{n} \Delta t^2 \pm \frac{1}{6} [\partial_t^3 \theta]_{j}^{n} \Delta t^3 + \mathcal{O}(\Delta t^4). \tag{4.55}
\]

Substitution of these series in (4.51) then gives

\[
[\partial_t \theta]_{j}^{n} - \kappa [\partial_x \theta]_{j}^{n} = -\zeta [\partial_t^2 \theta]_{j}^{n} + \mathcal{O}(\Delta t^2) + \mathcal{O}(\Delta x^2). \tag{4.56}
\]

where

\[
\zeta = \kappa \left( \frac{\Delta t}{\Delta x} \right)^2. \tag{4.57}
\]

To be consistent all the terms on the right-hand side of (4.56) must converge to zero in the limit \( \Delta x \to 0 \) and \( \Delta t \to 0 \). This is, however, not the case for the first term on the right-hand side which tends to infinity if \( \Delta x \) tends to zero faster than \( \Delta t \). We therefore note that (4.56) only converges to the continuous equation if \( \zeta \to 0 \) when \( \Delta x \to 0 \) and \( \Delta t \to 0 \). This implies that the scheme is consistent if and only if \( \Delta t \) tends to zero faster than \( \Delta x \). Thus there is a condition associated with the consistency of the scheme, and hence, in line with the formulation used for the stability condition, we refer to the DuFort-Frankel scheme as a conditionally consistent scheme under the condition that \( \Delta x \to 0 \) slower than \( \Delta t \to 0 \).

As already mentioned at the end of Section 3.2 the diffusion term is often used as a numerical method or trick to dissipate energy contained on the smaller scales in atmospheric and oceanographic models. Commonly this “noise” is created due to the presence of non-linear terms in the governing equations. This leads to non-linear interaction among the various wavelengths which in turn is responsible for a more or less continuous cascade of energy towards progressively smaller and smaller scales. If we neglect to dissipate the energy contained in the tail of the
energy spectrum we get an accumulation of energy at the $2\Delta x - 4\Delta x$ scales. In turn this accumulation at some time or another into the integration leads to a violation of the linear, numerical stability criterion and the numerical model blows up.

When the diffusion term is used for this purpose it does not represent any of the physical processes that we want to resolve. Rather it is introduced as a numerical method to avoid our model to blow up. Nevertheless it does represent low energy physics acting on scales smaller than those resolved by our grid. It is therefore a parametrization of these processes. Since this parametrization and/or the parameters it contains may change in accord with the models resolution we refer to it as subgrid scale (SGS) parametrization. Note that the SGS processes act on scales smaller than our grid resolution.

### 4.7 The Crank-Nicholson scheme

We will now consider another popular scheme called the Crank-Nicholson scheme. Like the Dufort-Frankel scheme it is also implicit. Its popularity is due to two facts. First, like all implicit schemes, it is unconditionally stable. Second it has a second order accuracy in time and space.

We start by defining the convenient notation

$$
\delta^2 x \psi^n_j = \psi^n_{j-1} - 2\psi^n_j + \psi^n_{j+1},
$$

(4.58)

which permit us to write the explicit, forward in time, centered in space scheme (4.4) as

$$
\theta^{n+1}_j = \theta^n_j + \frac{\kappa \Delta t}{\Delta x^2} \delta^2 x \theta^n_j.
$$

(4.59)

Recall that this scheme is conditionally stable under the condition

$$
\chi \equiv \frac{2\kappa \Delta t}{\Delta x^2} \leq 1.
$$

(4.60)

We may now construct an implicit scheme by changing the second term on the right-hand side of (4.59) from time level $n$ to time level $n + 1$. The result is

$$
\theta^{n+1}_j = \theta^n_j + \frac{\kappa \Delta t}{\Delta x^2} \delta^2 x \theta^{n+1}_j.
$$

(4.61)

We recall from Section 4.5 that implicit schemes are always stable. Thus (4.61) is an unconditionally stable scheme. Note that it is also consistent since it is based on Taylor series expansions. We also note that (4.61) is similar to (4.41) except that (4.61) is forward in time while (4.41) is centered in time.

We now combine the two schemes (4.59) and (4.61) to obtain

$$
\theta^{n+1}_j = \theta^n_j + \kappa \frac{\Delta t}{\Delta x^2} \left[ \gamma \delta^2 x \theta^{n+1}_j + (1 - \gamma) \delta^2 x \theta^n_j \right],
$$

(4.62)

3Note that for a given grid size $\Delta x$ the resolution equals the Nyquist wavelength $2\Delta x$. 

42
where $\gamma$ is a number so that $0 \leq \gamma \leq 1$. If $\gamma = 0$ then (4.62) reduces to the explicit scheme (4.59). If $\gamma = 1$ then (4.62) reduces to the implicit scheme (4.61). If $\gamma$ is between 0 and 1 then the scheme contains implicit as well as explicit terms. Such schemes are sometimes referred to as a semi-implicit scheme. To analyze the stability of the scheme we use von Neumann’s method. The growth factor $G$ is (cf. Exercise 5 on page 48)

$$G = \frac{1 - (1 - \gamma)\chi(1 - \cos \alpha \Delta x)}{1 + \gamma\chi(1 - \cos \alpha \Delta x)},$$

(4.63)

where $\chi$ is as given in (4.53). We recall that the condition $|G| \leq 1$ or $-1 \leq G \leq 1$ is a sufficient condition for numerical stability. From (4.63) follows that $G \leq 1$ is always satisfied, while $G \geq -1$ is satisfied if

$$\chi(1 - 2\gamma) \leq 1.$$  \hspace{1cm} (4.64)

For $\frac{1}{2} \leq \gamma \leq 1$ (4.64) is automatically satisfied. Under these circumstances the scheme is stable regardless of the value chosen for the increments $\Delta x$ and $\Delta t$. This implies that the scheme is unconditionally stable as long as $\frac{1}{2} \leq \gamma \leq 1$. This is not surprising because under these circumstances the weight is on the implicit part. As a corollary we note that this also proves that the pure implicit scheme (4.61), which follows from (4.62) by letting $\gamma = 1$, is indeed unconditionally stable. If however $0 \leq \gamma < \frac{1}{2}$ the weight is on the explicit part. Under these circumstances the scheme is conditionally stable under the condition (4.64). We note that for $\gamma = 0$, in which case (4.62) equals the forward in time, centered in space finite difference approximation for the diffusion equation as displayed in (4.4), we indeed retrieve the condition (4.32) of Section 4.3, that is, $\chi \leq 1$.

The value $\gamma = \frac{1}{2}$ is special. It constitutes the critical value at which the scheme (4.62) is still unconditionally stable. For this special value we get

$$\theta_{j+1} = \theta_j + \frac{1}{2}\kappa\Delta t \left[ \theta_{j-1} - 2\theta_j + \theta_{j+1} \right] + \mathcal{O}(\Delta x^2),$$

(4.65)

which is the Crank-Nicholson scheme. The scheme is special also in another respect. Despite the fact that we employ a forward in time finite difference approximation for the time rate of change of the variable $\theta$, it becomes of second order accuracy in time as well as in space. To prove it we start by utilizing the Taylor series expansions (2.21) and (2.22) as outlined in Section 2.6 on page 15. By substituting these series into the centered differences on the right-hand side of (4.65) we first get

$$\frac{\theta_{j+1} - \theta_j}{\Delta t} = \frac{1}{2}\kappa \left[ [\partial_x^2 \theta]^n_{j+1} + [\partial_x^2 \theta]^n_j \right] + \mathcal{O}(\Delta x^2).$$

(4.66)

Expanding $\theta_{j+1}^n$ and $[\partial_x^2 \theta]^n_{j+1}$ using Taylor series gives

$$\frac{\theta_{j+1}^n - \theta_j^n}{\Delta t} = [\partial_t \theta]^n_j + \frac{1}{2} [\partial_x^2 \theta]^n_j \Delta t + \mathcal{O}(\Delta t^2),$$

(4.67)

$$[\partial_x^2 \theta]^n_{j+1} = [\partial_x^2 \theta]^n_j + [\partial_t (\partial_x^2 \theta)]^n_{j+1} \Delta t + \mathcal{O}(\Delta t^2).$$

(4.68)
4.8 A DIRECT ELLIPTIC SOLVER

Substituting these series in (4.66) and rearranging terms we get

\[
\left[ \partial_t \theta \right]_j^n = \kappa \left[ \partial_x^2 \theta \right]_j^n - \frac{1}{2} \left\{ \left[ \partial_t^2 \theta \right]_j^n - \kappa \left[ \partial_t \left( \partial_x^2 \theta \right) \right]_j^n \right\} \Delta t + \mathcal{O}(\Delta t^2) + \mathcal{O}(\Delta x^2), \tag{4.69}
\]

Furthermore, we apply the continuous diffusion equation (4.1) to obtain

\[
\left[ \partial_x^2 \theta \right]_j^n = \kappa \left[ \partial_t \left( \partial_x^2 \theta \right) \right]_j^n. \tag{4.70}
\]

Thus the second term on the right-hand side of (4.69) vanishes and (4.69) becomes

\[
\left[ \partial_t \theta \right]_j^n = \kappa \left[ \partial_x^2 \theta \right]_j^n + \mathcal{O}(\Delta t^2) + \mathcal{O}(\Delta x^2). \tag{4.71}
\]

Thus besides being unconditionally stable, the Crank-Nicholson scheme is also of second order accuracy in time even though we employ a forward in time finite difference approximation. These two facts is the reason why the Crank-Nicholson scheme is popular when solving true diffusive problems. However, as shown in the next section it has one disadvantage compared to the more standard schemes. To solve it numerically we have to employ an elliptic solver.

4.8 A direct elliptic solver

If we inspect the Crank-Nicholson scheme (4.65) in more detail we find that including the implicit terms on the right-hand side of (4.65) turns the original parabolic equation into an apparent elliptic equation. By adding the new terms the original local algorithm is turned into a non-local or global algorithm. The implication is that the solution \( \theta_{n+1}^j \) at time level \( n+1 \) in addition to depend on the solution at the previous time level \( n \) also depends on \( \theta_{n+1}^{j-1} \) and \( \theta_{n+1}^{j+1} \), that is, depends on the solution at the adjacent space points \( \pm \Delta x \) away at the new time level \( n+1 \). We may illustrate this by rearranging the terms in (4.65) to obtain

\[
-\frac{\kappa \Delta t}{2 \Delta x^2} \theta_{n+1}^j + \left( 1 + \frac{2 \kappa \Delta t}{2 \Delta x^2} \right) \theta_{n+1}^j - \frac{\kappa \Delta t}{2 \Delta x^2} \theta_{n+1}^{j+1} = \theta_{n}^j + \frac{\kappa \Delta t}{2 \Delta x^2} \left[ \theta_{n-1}^j - 2 \theta_{n}^j + \theta_{n+1}^j \right]. \tag{4.72}
\]

Thus we cannot solve for \( \theta_{n+1}^j \) without knowing \( \theta_{n-1}^j \) and/or \( \theta_{n+1}^j \). Let us consider that we solve (4.72) for increasing values of \( j \). Then for any arbitrary \( j \) we have already solved for \( \theta_{n+1}^{j-1} \), and it thus known. However, we have not yet solved for \( j+1 \), and thus \( \theta_{n+1}^{j+1} \) is unknown.

Many of the model codes employed in numerical weather and numerical ocean weather prediction today employ semi-implicit methods. We are therefore in need of a method whereby such problem can be solved. Such methods are commonly referred to as elliptic solvers. Moreover, since we deal with time marching problems, we have to apply the elliptic solver for each time step. Thus we in addition need and elliptic solver that is efficient in the sense that it is fast on any computer. The most efficient elliptic solvers are those referred to as direct elliptic solvers.\footnote{In the infancy of numerical weather prediction (NWP) most elliptic solvers where so called iterative or indirect elliptic solvers. Even though the may be accelerated, as for instance in the special iterative elliptic solver called “Successive over-relaxation”, they are much slower than the direct methods.}

44
One such method is the so called \textit{Gauss elimination} which we will use as an example. It consists of two steps. The first is called a \textit{forward sweep}. Next we find the final solution by performing a \textit{backward substitution}.

To get started, we first rewrite (4.72) into a more general form,

\[ a_j \theta_{j-1}^{n+1} + b_j \theta_j^{n+1} + c_j \theta_{j+1}^{n+1} = h_j^n \quad ; \quad j = 2(1)J, \quad (4.73) \]

where \(a_j\), \(b_j\), and \(c_j\) represent the coefficients in (4.72). The use of the subscript \(j\) attached to these coefficients is to acknowledge that they in general are functions of space. Likewise \(h_j^n\) on the right-hand side represents all “forcing” terms, that is, our knowledge of the solution at the previous time step(s). We also note that we are required to solve (4.73) within a finite domain. Thus at \(j = 1\) and at \(j = J + 1\) the variable \(\theta\) is determined by the boundary conditions. For convenience we assume that these are simple Dirichlet conditions, in which case \(\theta_1^n\) and \(\theta_{J+1}^n\) are known functions.

For convenience we will also drop the superscript \(n\) and \(n + 1\). Thus we are required to solve,

\[ a_j \theta_{j-1} + b_j \theta_j + c_j \theta_{j+1} = f_j, \quad j = 2(1)J, \quad (4.74) \]

under the conditions

\[ \theta_1 = \hat{\theta}_0, \quad \text{and} \quad \theta_{J+1} = \hat{\theta}_L, \quad (4.75) \]

where \(\hat{\theta}_0\) and \(\hat{\theta}_L\) are known functions. We observe that (4.74) in tensor form may compactly be written as

\[ \mathbf{A} \cdot \mathbf{\theta} = \mathbf{h}', \quad (4.76) \]

where the tensor \(\mathbf{A}\) is the tridiagonal matrix

\[
\mathbf{A} = \begin{bmatrix}
    b_2 & c_2 & 0 & \ldots & 0 & 0 & 0 \\
    a_3 & b_3 & c_3 & \ldots & 0 & 0 & 0 \\
    0 & a_4 & b_4 & \ldots & 0 & 0 & 0 \\
    \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
    \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
    0 & 0 & 0 & \ldots & a_{J-1} & b_{J-1} & c_{J-1} \\
    0 & 0 & 0 & \ldots & 0 & a_J & b_J
\end{bmatrix}.
\]

(4.77)

and the vectors \(\mathbf{\theta}\) and \(\mathbf{h}'\) are, respectively,

\[
\mathbf{\theta} = \begin{bmatrix}
    \theta_2 \\
    \theta_3 \\
    \vdots \\
    \theta_{J-1} \\
\end{bmatrix},
\]

(4.78)

and

\[
\mathbf{h}' = \begin{bmatrix}
    h_2 - a_2 \hat{\theta}_0 \\
    h_3 \\
    \vdots \\
    h_{J-1} - c_{J-1} \hat{\theta}_L
\end{bmatrix}.
\]

(4.79)
Forward sweep

We are now ready to perform the forward sweep. The idea is to replace all elements of the matrix $\mathcal{A}$ positioned in the lower left half with zeroes. At the same time it is convenient to normalize the diagonal elements, that is, turn everyone of them into the value 1. We start with the equation for $j = 2$. From (4.74) follows

$$b_2 \theta_2 + c_2 \theta_3 = h'_2.$$  \hfill (4.80)

We then normalize by dividing by $b_2$

$$\theta_2 + d_2 \theta_3 = \frac{h'_2}{b_2},$$  \hfill (4.81)

where

$$d_2 = \frac{c_2}{b_2} \quad \text{and} \quad w_2 = \frac{h'_2}{b_2}. \hfill (4.82)$$

For $j = 3$ we now obtain from (4.74) that

$$a_3 \theta_2 + b_3 \theta_3 + c_3 \theta_4 = h'_3.$$  \hfill (4.83)

Substituting for $\theta_2$ from (4.81) and normalizing gives

$$\theta_3 + d_3 \theta_4 = w_3,$$  \hfill (4.84)

where

$$d_3 = \frac{c_3}{b_3 - d_2 a_3} \quad \text{and} \quad w_3 = \frac{h'_3 - a_3 w_2}{b_3 - d_2 a_3}. \hfill (4.85)$$

We now define the recursion formula

$$d_j = \begin{cases} \frac{c_j}{b_j} & ; \; j = 2 \\ \frac{c_j}{b_j - d_{j-1} a_j} & ; \; j = 3(1)J - 1 \\ 0 & ; \; j = J \end{cases} \hfill (4.86)$$

and

$$w_j = \begin{cases} \frac{h'_j}{b_j} & ; \; j = 2 \\ \frac{h'_j - a_j w_{j-1}}{b_j - d_{j-1} a_j} & ; \; j = 3(1)J \end{cases} \hfill (4.87)$$

Note that all the coefficients $d_j$ and $w_j$ can be calculated once and for all. For an arbitrary $j$ we may therefore write

$$\theta_j + d_j \theta_{j+1} = w_j, \quad j = 2(1)J.$$  \hfill (4.88)

We observe that from (4.86) $d_J = 0$. Hence for $j = J$ we obtain the very simple equation

$$\theta_J = w_J,$$  \hfill (4.89)

where $w_J$ is already computed by application of (4.87). In matrix form (4.77) now reads

$$\mathcal{A}' \cdot \theta = w,$$  \hfill (4.90)
where the matrix $\mathbf{A}'$ is

$$\mathbf{A}' = \begin{bmatrix}
1 & d_2 & 0 & \ldots & 0 & 0 \\
0 & 1 & d_3 & \ldots & 0 & 0 \\
0 & 0 & 1 & \ddots & \vdots & \vdots \\
\vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\
0 & 0 & 0 & \ldots & 1 & d_{J-2} \\
0 & 0 & 0 & \ldots & 0 & 1
\end{bmatrix},$$

(4.91)

and the vector $\mathbf{w}$ is

$$\mathbf{w} = \begin{bmatrix}
w_2 \\
w_3 \\
w_4 \\
\vdots \\
w_{J-2} \\
w_{J-1}
\end{bmatrix}.$$

(4.92)

**Backward substitution**

We are now ready to perform the backward substitution. First we note that all the $w_j$’s and the $d_j$’s are known using the recursion formula (4.86) and (4.87). Second we note from (4.89) that $\theta_j$ is simply given by $w_j$ and that the latter is known from (4.87). Thus we are in a position where we can solve for all the remaining $\theta_j$’s for $j = 2(1)J - 1$ simply by employing (4.88) backwards. Indeed solving (4.88) with respect to $\theta_j$ we get

$$\theta_j = w_j - d_j \theta_{j+1} \text{ for } j = J - 1(-1)2.$$  

(4.93)

The Gauss elimination method is very simple to program, and it is also very efficient and fast on the computer. An example on the usefulness of this method, in which you are also required to program the method, is given in Computer Problem 3 named “Yoshida’s equatorial jet current” in the accompanied, but separate Computer Problem notes. We urge the reader to do this problem, at least solve the resulting ODE by employing the Gauss elimination method.

**Exercises**

1. Show that the scheme (4.13) is unconditionally unstable. Hint: Show that $|G| > 1$ regardless of the choice made for $\Delta t$ and $\Delta x$.

2. Show that if $|G| = 1$ then the chosen scheme has no numerical dissipation.

3. Show that the growth factor associates with the scheme (4.41) is

$$G = \left[ 1 + \frac{4\kappa \Delta t}{\Delta x^2} (1 - \cos \alpha \Delta x) \right]^{-\frac{1}{2}}$$

(4.94)
and hence that the scheme is unconditionally stable. Also show that $|G| < 1$ for all wavelengths. Note that $|G|$ decreases as $\Delta t$ increases.

4. Show that the growth factor for the DuFort-Frankel scheme (4.51) is

$$G_{1,2} = \frac{\gamma \cos \alpha \Delta x \pm \sqrt{1 - \gamma^2 (\sin \alpha \Delta x)^2}}{1 + \gamma} \quad \text{where} \quad \gamma = \frac{2\kappa \Delta t}{\Delta x^2}, \quad (4.95)$$

5. Show that the expression (4.63) is indeed the expression for the growth factor of the scheme (4.62) when using von Neumann’s analysis method.
Chapter 5

THE ADVECTION PROBLEM

In this chapter we will investigate possible numerical methods whereby the advection equation (3.11) may be solved by numerical means. As we did for the diffusion problem we will reduce it to its simplest form\(^1\), and hence we consider a one-dimensional advection process, that is, let \( \mathbf{F} = u \hat{\mathbf{i}} \). Under these circumstances the advection equation (3.11) in its continuous form becomes

\[
\partial_t \theta + \partial_x (u \theta) = 0. \tag{5.1}
\]

Here \( u(x, t) \) is the advection speed along the \( x \)-axis. Note that in general \( u \) varies in time and space in which case the problem is non-linear. We will in most instances below assume that it is uniform in time and space. We therefore let \( u = u_0 = \text{constant} \), and hence (5.1) is written.

\[
\partial_t \theta + u_0 \partial_x \theta = 0. \tag{5.2}
\]

The general true solution to (5.2) is

\[
\theta = \theta(x - u_0 t). \tag{5.3}
\]

If the initial condition is a harmonic (monochromatic) wave, say

\[
\theta(x, 0) = \Theta_0 e^{i\alpha x}, \tag{5.4}
\]

where \( \alpha \) is the wave number and \( \Theta_0 \) the amplitude, then the true solution to the advection equation becomes

\[
\theta(x, t) = \Theta_0 e^{i\alpha(x-u_0 t)}. \tag{5.5}
\]

This particular solution is a wave propagating in the positive \( x \) direction with a phase speed given by \( u_0 \). As alluded to in Chapter 1, this solution is typical of hyperbolic systems. Indeed, the solutions (5.3) and (5.5) is such that if we travel along with the advection speed we will experience no change in the property \( \theta \). If we, however, observe the wave from a fixed position in space, the property \( \theta \) will change in accord with (5.3) as the “wave” passes by.

\(^1\)“Make things as simple as possible, but no simpler” Albert Einstein (1879-1955)
5.1. FINITE DIFFERENCE FORM  

Using Fourier transforms as outlined in Section 2.11 we may also write the solution to (5.2) formally as

\[ \theta(x, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{\theta}(t, \alpha) e^{-i\alpha x} d\alpha. \]  

(5.6)

where \( \hat{\theta}(t, \alpha) \) is the Fourier transform of \( \theta \), that is,

\[ \hat{\theta}(t, \alpha) = \int_{-\infty}^{\infty} \theta(x, t) e^{i\alpha x} dx. \]  

(5.7)

We may also Fourier transform the advection equation (5.2) which then becomes

\[ \frac{d\hat{\theta}}{dt} = i\alpha u_0 \hat{\theta}. \]  

(5.8)

Solving this equation we get

\[ \hat{\theta}(t, \alpha) = \hat{\Theta}(\alpha) e^{i\alpha(x - u_0 t)}. \]  

(5.9)

We note that the advection equation (5.2) contains first order derivatives in time and space only. We are therefore only allowed to specify two boundary conditions. One of these is the initial condition. Let us represent the initial condition at time \( t = 0 \) by the \( \theta_0(x) \). The Fourier transform of \( \theta_0 \) is then

\[ \hat{\theta}_0(\alpha) = \int_{-\infty}^{\infty} \theta_0(x) e^{i\alpha x} dx, \]  

(5.10)

and hence \( \hat{\Theta} = \hat{\theta}_0(\alpha) \). By use of (5.6) we can therefore formally write the solution to the one-dimensional advection equation (5.2), satisfying the initial condition, as

\[ \theta(x, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{\theta}_0(\alpha) e^{i\alpha(x - u_0 t)} d\alpha. \]  

(5.11)

This formal solution accentuates that the general true solution to the advection equation as given in (5.3) consists of waves of various wavelengths (wavenumber) all of which propagates at the same speed \( u_0 \). Furthermore, (5.11) underscores that the dominant waves, that is, the waves containing most energy, are those associated with the initial condition.

5.1 Finite difference approximations

Our concern is to develop an algorithm by which (5.2) can be solved by numerical means. To find a suitable finite difference approximation we follow the procedure used to derive a finite difference approximation to the diffusion equation. Different approximation will then naturally lead to different schemes. Our task is then to satisfy ourselves that the chosen scheme is numerically stable and consistent. If not the scheme is of no use to us and have to be discarded. Hence once a finite difference approximation is chosen we have to analyze it with respect to its stability and consistency. The former is performed making use of von Neumann’s method (cf. Section 4.3),
while consistency is analyzed via use of Taylor series (cf. Section 2.6) as outlined in Section 4.6 (page 41).

Let us start by applying the successful forward in time and centered in space scheme that worked well for the diffusion equation. Thus we first construct a forward in time finite difference approximations for the time rate of change, using Taylor series expansions, that is,

$$\left[ \frac{\partial_t \theta}{\Delta t} \right]_j^n = \frac{\theta_{j}^{n+1} - \theta_{j}^{n}}{\Delta t}. \quad (5.12)$$

In a similar fashion, we next construct a centered in space finite difference approximation for the first order space differential in (5.2), that is,

$$\left[ \frac{\partial_x \theta}{\Delta x} \right]_j^n = \frac{\theta_{j}^{n} - \theta_{j-1}^{n}}{2\Delta x}. \quad (5.13)$$

Substitution of (5.12) and (5.13) into (5.2) then gives

$$\frac{\theta_{j}^{n+1} - \theta_{j}^{n}}{\Delta t} + u_0 \frac{\theta_{j+1}^{n} - \theta_{j-1}^{n}}{2\Delta x} = 0,$$

which when solved with respect to \(\theta_{j}^{n+1}\) gives

$$\theta_{j}^{n+1} = \theta_{j}^{n} - u_0 \frac{\Delta t}{2\Delta x} (\theta_{j+1}^{n} - \theta_{j-1}^{n}), \quad \left\{ \begin{array}{l} j = 2(1)J - 1 \\ n = 0(1) \ldots \end{array} \right. \quad (5.15)$$

Since the finite difference approximation (5.15) is based on Taylor series expansions, we know apriori that the consistency requirement is satisfied. It therefore remains to analyze its stability to satisfy ourselves that the scheme is useful to us. To this end we make use of von Neumann’s method, and start by substituting the discrete Fourier component (4.21) into (5.15). Thus we get

$$\Theta_{n+1} = \Theta_{n} - \frac{u_0 \Delta t}{2\Delta x} (e^{i\alpha \Delta x} - e^{-i\alpha \Delta x}) \Theta_{n}\quad (5.16)$$

where the common factor \(e^{i\alpha \Delta x}\) is tossed away. Recalling the definition of the growth factor (4.22), and noting that \(e^{i\alpha \Delta x} - e^{-i\alpha \Delta x} = 2i \sin \alpha \Delta x\), we find

$$G = 1 - i\lambda,$$

where

$$\lambda = \frac{u_0 \Delta t}{\Delta x} \sin \alpha \Delta x. \quad (5.18)$$

We observe that the growth factor now is a complex number with a real part given by 1 and an imaginary part given by \(\lambda\). According to von Neumann’s method we are required to evaluate the absolute value of the growth factor. To this end we use the well known property of imaginary numbers, namely that its absolute value equals the square root of the sum of the squares of the real and imaginary parts\(^2\). Thus

$$|G| = \sqrt{1 + \lambda^2}. \quad (5.19)$$

\(^2\)Let \(A = a + ib\) be an imaginary number with real part \(a\) and imaginary part \(b\). Then \(|A| = \sqrt{AA^*} = \sqrt{a^2 + b^2}\) where \(A^* = a - ib\) is the complex conjugate of \(A\).
We observe that since $\lambda^2$ is a positive definite the radical is always larger than or equal to zero and hence $|G| \geq 1$. Since $|G| = 1$ only for the special wavenumbers that make $\sin \alpha \Delta x = 0$ it follows that the scheme is in general unconditionally unstable. We then have the somewhat curious result that although the forward in time, centered in space scheme worked fine for the diffusion problem, it is totally unacceptable with regard to the advection problem.

\begin{quote}
Never use a forward in time, centered in space scheme for the advection problem. It always leads to an unconditionally unstable scheme.
\end{quote}

This does not come as a total surprise. As alluded to in Chapter 2 the advection equation and the diffusion equation represent quite different physics and have quite different characteristics. While the diffusion equation is parabolic, the advection equation is hyperbolic. We should therefore expect that a FD scheme that works well diffusion not necessarily works well for advection.

## 5.2 Some earlier stable and consistent schemes

As listed in, e.g., O’Brien (1986) (page 165 and onwards) there are many stable schemes that have been suggested over the past to solve the advection equation. The reason is that advection is one of the most prominent atmospheric and oceanic processes. Many of the schemes are constructed to minimize unwanted properties of other schemes (e.g., numerical diffusion as detailed in Section 5.7 below, or numerical dispersion as detailed in Section 5.9 below), while yet other schemes are constructed with focus on their efficiency on the computer.

With the advent of todays supercomputers with their dramatic increase in power and speed, many of the earlier schemes are now superfluous. The earlier requirements of computer efficiency is simply lessened, and the focus is therefore shifted to derive schemes that provide better conservation properties, and that contain higher order accuracy, say schemes of $O(\Delta t^4, \Delta x^4)$ or higher (cf. Section 9.1 on page 111). Such schemes includes flux corrective schemes (cf. Section 5.8 on page 63), and lagrangian schemes (cf. Section 5.5 on page 57).

It is nevertheless constructive to analyze some of the earlier schemes before going into details about the more recently suggested schemes. Thus we start by analyzing a few of the earlier schemes, namely the leapfrog scheme, the upwind or upstream scheme and the Lax-Wendroff scheme.

### The leapfrog scheme

One of the first schemes employed in atmospheric and oceanic problems is the centered in time and centered in space scheme, a scheme that we strongly emphasized was totally useless for the diffusion problem. To construct this scheme we use Taylor series expansions. A centered in time finite difference approximation for the time rate of change is thus

$$\left[ \frac{\partial t}{\partial t} \right]_j^n = \frac{\theta_j^{n+1} - \theta_j^{n-1}}{2\Delta t}, \quad (5.20)$$

52
while a centered in space finite difference approximation to the first order space derivative is

\[
[\partial_x \theta]^n_j = \frac{\theta^n_{j+1} - \theta^n_{j-1}}{2\Delta x}.
\] (5.21)

Since we have used centered approximation we note that the truncation error is of \( \Delta x^2 \) and \( \Delta t^2 \).

By substituting (5.20) and (5.21) into (5.2) we get after some rearranging of terms

\[
\theta^{n+1}_j = \theta^{n-1}_j - u_0 \frac{\Delta t}{\Delta x} (\theta^n_{j+1} - \theta^n_{j-1}), \quad \{ j = 2(1)J - 1 \}
\] (5.22)

This scheme is commonly referred to as the **leapfrog** scheme. The reason is that we use information from all the points surrounding the \( x_j, t^n \) point in \( x,t \) space, but do not incorporate information from the point \( x_j, t^n \) itself. In a sense we are “leapfrogging” the point \( x_j, t^n \). We note that since it is derived by exclusively using Taylor series, it is a consistent scheme.

The scheme is traditionally fairly popular for several reasons. For one the scheme is, as shown in Section 5.3 (page 55), neutrally stable under the condition that \( u_0 \Delta t \leq \Delta x \), the so called **Courant-Friedrich-Levy condition** or CFL condition. Thus there is no numerical or artificial damping or energy dissipation associated with the scheme when going from one time level to the next, a highly desirable property. Secondly the scheme is of second order accuracy. Finally, the scheme works fast and efficiently on the computer.

It has, however, some disadvantageous properties. First, as detailed in Section 5.9 (page 66), the scheme contains what is referred to as **numerical dispersion**, which sometimes leads to negative tracer concentrations. Secondly the scheme, as detailed in Section 5.10 below, also contains what is referred to as **unphysical modes** that has to be dealt with.

**The upstream scheme**

Because of these somewhat unwanted properties of the leapfrog scheme the so called **upwind** or **upstream** scheme became quite popular early on, and is quite common even today. As the name indicates the scheme make use of information exclusively from upstream to calculate the value at the new time level. It is a two time level scheme which is forward in time and one-sided in space. If the advection velocity is positive it is backward in space, and forward in space if the advection velocity is negative. Thus, again using Taylor series expansions, we get

\[
\theta^{n+1}_j = \theta^n_j - u_0 \frac{\Delta t}{\Delta x} \left\{ \begin{array}{ll}
\theta^n_{j+1} - \theta^n_{j-1} & u_0 \geq 0 \\
\theta^n_{j+1} - \theta^n_{j-1} & u_0 < 0
\end{array} \right. \quad (5.23)
\]

The scheme is stable under the CFL condition or as long as \( u_0 \Delta t \leq \Delta x \). We note for later convenience that when \( u_0 > 0 \) then (5.23) may be written as

\[
\theta^{n+1}_j = (1 - C) \theta^n_j + C \theta^n_{j-1}
\] (5.24)

where

\[
C = \frac{u_0 \Delta t}{\Delta x}
\] (5.25)
is known as the Courant number.

One of the major advantages of the upwind scheme is that it conserves the fact that tracer concentration is a positive definite quantity. Furthermore we observe that it is a consistent scheme since it is derived using Taylor series. Being a two level scheme it also works fast and efficient on the computer. Despite of these circumstances the upwind scheme has one major drawback. It contains, as detailed in Section 5.7 on page 61, what we refer to as numerical diffusion. Depending on the choices we make regarding the time step and the space increment the numerical diffusion may be large and sometimes larger than the actual physical diffusion of the original problem. It therefore tends to even out the solution artificially as time progresses. In particular areas where large gradients appear, e.g., frontal areas, are prone to an artificial diffusion. Thus fronts are diffused which in turn inhibits realistic baroclinic instability processes. In addition we observe that the upwind scheme has truncation errors that is first order in time and space, that is, its accuracy is $O(\Delta t)$ and $O(\Delta x)$, which is one order of magnitude less than the leapfrog scheme. Because of these rather disadvantageous properties this author does not recommend the use of the upstream scheme.

The Lax-Wendroff scheme

To avoid or lessen the impact of the first order numerical diffusion inherent in upwind schemes, and to increase its accuracy, Lax and Wendroff (1960) advocated the use of the now named Lax-Wendroff scheme. It consists of two steps. The first is a diffusive step in which the solution at the mid time level $n + \frac{1}{2}$ is found by employing the so called diffusive scheme. This is basically a forward in time, centered in space scheme in which the value at the grid point $j$ at time level $n$, $\theta_j^n$, is replaced by an interpolated value using the adjacent grid points,

$$\theta_j^n = \frac{1}{2} (\theta_{j+1}^n + \theta_{j-1}^n). \quad (5.26)$$

Thus the first step of the scheme is

$$\theta_j^{n+\frac{1}{2}} = \frac{1}{2} (\theta_{j+1}^n + \theta_{j-1}^n) - u_0 \frac{\Delta t}{4\Delta x} (\theta_{j+1}^n - \theta_{j-1}^n). \quad (5.27)$$

We note that even though the forward in time, centered in space scheme is unstable the trick of replacing $\theta_j^n$ by half the sum of its nearest neighbors makes the scheme stable under the condition that the Courant number is less than or equal to one. The next step is to employ the leapfrog scheme (5.22) to find the solution at time level $n + 1$,

$$\theta_j^{n+1} = \theta_j^n - u_0 \frac{\Delta t}{2\Delta x} (\theta_{j+1}^{n+\frac{1}{2}} - \theta_{j-1}^{n+\frac{1}{2}}). \quad (5.28)$$

Alternatively we can use the first step to find the solution at time level $n + 1$ and then use the leapfrog scheme to find the solution at time level $n + 2$, that is, alternate between using the diffusive scheme and the leapfrog scheme. If we now eliminate the dependence on $n + \frac{1}{2}$ by substituting (5.27) into (5.28) we get

$$\theta_j^{n+1} - \theta_j^n = -\frac{1}{4} C (\theta_{j+2}^n - \theta_{j-2}^n) + \frac{1}{8} C^2 (\theta_{j+2}^n - 2\theta_j^n + \theta_{j-2}^n), \quad (5.29)$$
where \( C = u_0 \Delta t / \Delta x \) is the Courant number.

The question is then whether the scheme (5.29) is consistent and stable? In contrast to the leapfrog scheme and the upwind scheme, who are both derived using Taylor series, it is not obvious that the Lax-Wendroff scheme is consistent. In fact the first step is inconsistent. To prove it we simply substitute the respective Taylor series into (5.27) and get

\[
\begin{align*}
\partial_t \theta^n_j &= -u_0 [\partial_x \theta]_j^n + u_0 [\partial_x^2 \theta]_j^n \Delta t + O(\Delta t^2) + O(\Delta x^2),
\end{align*}
\]

(5.30)

where \( \kappa = \Delta x^2 / \Delta t \). Hence we notice that the advection equation (5.2) is not recovered when \( \Delta x \to 0 \) and \( \Delta t \to 0 \) independently. In contrast we may prove, by performing a similar analysis, that the Lax-Wendroff scheme is consistent. To this end we first substitute the respective Taylor series into (5.29). The result is

\[
\begin{align*}
\partial_t \theta^n_j &= -u_0 [\partial_x \theta]_j^n + \frac{1}{2} \left( [\partial_t^2 \theta]_j^n - u_0 [\partial_t^2 \theta]_j^n \right) \Delta t + O(\Delta t^2) + O(\Delta x^2).
\end{align*}
\]

(5.31)

Since (5.2) implies that \( \partial_t^2 \theta = -u_0 \partial_x (\partial_t \theta) = u_0^2 \partial_x^2 \theta \), we can toss away the second term on the right-hand side, and hence we get

\[
\begin{align*}
\partial_t \theta^n_j &= -u_0 [\partial_x \theta]_j^n + O(\Delta t^2) + O(\Delta x^2).
\end{align*}
\]

(5.32)

We first observe that (5.2) is recovered when \( \Delta x \to 0 \) and \( \Delta t \to 0 \) independently. Hence the scheme is indeed consistent. Second we observe that the terms neglected are \( O(\Delta x^2) \) and \( O(\Delta t^2) \), and hence that the scheme is second order accurate.

**Implicit scheme**

As for the diffusion equation we may also construct an implicit scheme for the advection equation. This is easily constructed by using a backward in time, centered in space scheme. Thus

\[
\begin{align*}
\theta^{n+1}_j &= \theta^n_j - u_0 \frac{\Delta t}{2 \Delta x} \left( \theta^{n+1}_{j+1} - \theta^{n+1}_{j-1} \right),
\end{align*}
\]

(5.33)

We observe that this scheme is \( O(\Delta t) \) and \( O(\Delta x^2) \), that is first order in time and second order in space accurate.

### 5.3 Stability of the earlier schemes: The CFL condition

So far we have only analyzed the schemes with respect to their consistency. What about their stability? And if stable, under what conditions are they stable?

We start by analyzing the leapfrog scheme (5.22). To this end we make use of von Neumann’s method as outlined several times already and which was introduced in Section 4.3. Thus we first replace the dependent variable \( \theta \) in (5.22) by its discrete Fourier component (4.21) to give

\[
\begin{align*}
\Theta_{n+1} &= \Theta_{n-1} - 2iu_0 \Delta t \Delta x \sin \alpha \Delta x \Theta_n,
\end{align*}
\]

(5.34)
To find the growth factor $G$ we first make use of (4.22) and then multiply by the growth factor to obtain

$$G^2 + 2i\lambda G - 1 = 0,$$

where

$$\lambda = \frac{u_0}{\Delta x} \sin \alpha \Delta x.\quad (5.36)$$

We note that the second order equation for the growth factor now contains the imaginary number $i$. The solution to this equation is

$$G_{1,2} = -i\lambda \pm \sqrt{1 - \lambda^2}.\quad (5.37)$$

Thus the growth factor is complex, and under the condition that the radical is positive, it contains both a real part and an imaginary part. This was to be expected since the factor in front of the first order term in (5.35) is imaginary. Under these circumstances, and again using the theorem that the absolute value of a complex number is the square root of the complex number itself multiplied by its complex conjugate we obtain (cf. eq. 5.19 of Section 5.1)

$$|G_{1,2}| = \sqrt{G_{1,2}G_{1,2}^*} = 1.\quad (5.38)$$

Since by definition $\Theta_{n+1} \equiv |G|\Theta_n$ it follows that there is no artificial or numerical damping or distortion involved when going from one time level to the next. The scheme (5.22) is therefore \textit{neutrally stable} implying that the energy associated with \(\theta\) is conserved. We note that this is in line with the property of advection processes outlined in Section 3.3.

To obtain this desirable property the condition we must require that the radical in (5.37) is positive. Hence the condition

$$1 - \lambda^2 \leq 0 \quad \text{or} \quad |\lambda| \leq 1\quad (5.39)$$

must be satisfied. Since $-1 < \sin \alpha \Delta x < 1$ it follows that if

$$|u_0| \frac{\Delta t}{\Delta x} \leq 1\quad (5.40)$$

then (5.39) is satisfied, that is, the condition (5.40) is a sufficient condition for stability for the leapfrog scheme (5.22). Moreover under this condition the leapfrog scheme also becomes neutrally stable. The condition or criterion (5.40) is the CFL condition. The ratio or number

$$C = \frac{u_0 \Delta t}{\Delta x}\quad (5.41)$$

is the \textit{Courant number}. Since $\Delta x$ more often than not is given by the need to resolve the spatial structure or typical wavelengths of the physical problem, the CFL condition becomes a stringent upper bound on the time step $\Delta t$, that is, $\Delta t \leq \Delta x / |u_0|$. Therefore the larger the advection speed the smaller the time step, and the smaller the grid size the smaller the time step.

Regarding the stability of the upstream scheme and the Lax-Wendroff scheme we refer to Exercise 1 at the end of this Chapter. We end by investigating the stability of the implicit scheme (5.33) again using von Neumann’s method. Thus we get

$$\Theta_{n+1} = \Theta_n - iC \sin \alpha \Delta x \Theta_{n+1},\quad (5.42)$$
which results in a growth factor given by

\[ G = \frac{1}{\sqrt{1 + C^2 \sin^2 \alpha \Delta x}} < 1 \quad (5.43) \]

for all finite time steps. Thus the implicit scheme, as expected, is stable. Moreover it avoid the restrictive CFL condition.

Finally we note that like the the grid size also the time step must be sufficient to resolve the typical periods of the physical problem. Commonly the typical period is much longer than the Nüquist frequency \(2\Delta t\), and hence the CFL condition in most cases puts a much more stringent requirement on \(\Delta t\) than the requirement of resolving the typical periods of the physical problem. Thus for most meteorological and oceanographic problems the resolution requirement is on the grid size.

### 5.4 How to fix the intial boundary value problem

Although the leapfrog scheme is conditionally and neutrally stable it is not without disadvantageous properties. One of the problems with the scheme is associated with the number of initial conditions required, a problem already touched upon in Section 4.1 regarding the diffusion problem. Thus the question is how to start the time marching procedure. The forward in time centered in space scheme is one possibility, but as shown in Section 5.1 this is unconditionally unstable. However, we may nevertheless make use of this scheme when applied to a single time step.

Thus we start by using the scheme

\[ \theta_j^1 = \theta_j^0 - u_0 \frac{\Delta t}{2\Delta x} (\theta_{j+1}^0 - \theta_{j-1}^0) \quad (5.44) \]

For the time level 2 and onwards we then use the leapfrog scheme (5.22). The step (5.44) and more generally the forward in time, centered in space scheme is usually referred to as the Euler scheme. We emphasize that although the Euler scheme is unconditionally unstable it does not ruin the solution when applied for one time step only. It may even be used from time to time to avoid the unphysical mode inherent in the leapfrog scheme (cf. Section 5.10).

### 5.5 The method of characteristics

To help understand why the upwind scheme becomes unstable when the CFL condition is not satisfied we will first solve the advection equation using the method of characteristics.

**Theory**

Let the slopes

\[ \frac{D^* x}{dt} = u(x, t) \quad (5.45) \]
define special curves in the $t, x$ space (cf. Fig. 5.1), and let us simultaneously define the special differential operator

$$D^* \equiv \partial_t + D^* x \partial_x.$$  \hfill (5.46)

Then the advection equation (5.1) may be rewritten to yield

$$\frac{D^* \theta}{dt} = 0 \quad \text{along the slopes} \quad \frac{D^* x}{dt} = u.$$ \hfill (5.47)

We commonly refer to the slopes defined by (5.45) as the \textit{characteristics} and (5.45) itself as the \textit{characteristic equation}. Since the solutions to (5.1) also are solutions to (5.47) we often refer (5.47) as the \textit{compatibility equation}.

Figure 5.1: Sketch of the characteristics in the $x, t$ plane. For $u = u_0 = \text{constant} > 0$ the characteristics are the straight lines sloping to the the right in $x, t$ space as given by (5.45). If $x = L$ marks the end of the computational domain, then all information about the initial condition is lost for times $t > t_c$.

We observe that (5.47) tells us that $\theta$ is conserved along the characteristics (5.45). Thus if we know the solution at time $t = 0$, that is, $\theta(x, t = 0)$ for $0 < x < L$, then the solution at any later time $t > 0$ and at any particular point $x$ in space is found by simply following the characteristic back from the point $x, t$ toward the initial time $t = 0$ as illustrated in Figure 5.1. For $u = u_0 = \text{constant}$ the characteristics deform to straight lines with positive slopes $1/u_0$ when $u_0 > 0$. From Figure 5.1 we may conclude that after a critical time $t = t_c = L/u_0$ all information about the initial distribution of $\theta$ is lost. Indeed for $t > t_c$ it follows that the solution in the computational domain $0 < x < L$ is determined wholly by the boundary condition at $x = 0$. Since (5.1) only contains the first derivative with respect to $x$, only one condition is in
$x$ is allowed. The boundary at $x = L$ is therefore open in the sense that there is no boundary condition that replaces the differential equation there. The physical space therefore, in principle, continues to infinity. Thus the boundary $x = L$ is a numerical boundary necessitated by the fact that any computer, however large, are limited in its capacity. This problem is especially compound for oceanographic models, since the oceanic spatial scales are small compared to the similar scales in the atmosphere. We will come back to this problem in Chapter 8 where we will investigate details concerning conditions constraining the solutions at open boundaries.

We note in passing that since (5.1) and (5.47) are compatible, a solution to (5.47) is also automatically a solution to (5.1). We may therefore solve (5.47) employing numerical methods as well as (5.1) in which case it is referred to as the method of characteristics. We will make use of this fact to give a physical interpretation of the CFL criterion. We also note that the method of characteristic may be applied to much more complex problems and systems than the simple advection equation used merely as an example here (e.g., Lister, 1966; Røed and O’Brien, 1983).

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{method_of_characteristics.png}
\caption{Sketch of the method of characteristics. The distance between the grid points are $\Delta t$ in the vertical and $\Delta x$ in the horizontal direction. The sloping solid line is the characteristic through the point $j, n + 1$. It is derived from (5.45) and the slope is given by $1/u (u > 0)$. The point labeled $Q$ is therefore a distance $u\Delta t$ to the left of $x_j$. As long as $u\Delta t < \Delta x$ then $Q$ is located between $x_{j-1}$ and $x_j$. If however $u\Delta t > \Delta x$ then the point $Q$ is located to the left of $x_{j-1}$.}
\end{figure}
Numerical considerations

Since in this simple case (5.47) tells us that \( \theta \) is conserved along the characteristics, the problem is reduced to find the characteristics in the \( x, t \) space, that is, to solve (5.45). As is now common we then divide the computational domain in the \( x, t \) space into a grid as displayed in Figure 5.2. Let us for a minute consider \( u \) as being known for all \( x_j \) and at all time levels \( t_n \). Then at the time level \( t_{n+1} \) the characteristic through \( x_j \) is simply given by the characteristic equation (5.45).

To find the characteristic we make a simple finite difference approximation of (5.45) considering that \( u_{n+1}^j \) can be approximated by \( u_n^j \). Hence

\[
\frac{x_j - x_Q}{\Delta t} = u_j \quad \text{or} \quad x_Q = x_j - u_j^n \Delta t
\]

where \( x_Q \) is the point at which the characteristic crosses the time level \( t^n \). Since we know both \( u_j^n \) and \( x_j \), (5.48) is really an equation which determines the location of \( x_Q \). We note that as long as \( u_j^n \Delta t < \Delta x \) then \( x_Q \) is located between the grid points \( x_j \) and \( x_{j-1} \). Since \( \theta_j^n \) is known for all grid points, we may interpolate linearly between the adjacent grid points to find \( \theta_Q^n \), or the value of \( \theta \) at the location \( x_Q \) at time level \( t = t^n \). To this end we may use a two point linear interpolation. Thus

\[
\theta_Q^n = \theta_{j-1}^n + \frac{x_Q - x_j^n}{\Delta x} (\theta_j^n - \theta_{j-1}^n)
\]

Substituting \( x_Q \) from (5.48) into (5.49) than gives

\[
\theta_Q^n = (1 - C)\theta_j^n + C\theta_{j-1}^n,
\]

where

\[
C = \frac{u_j^n \Delta t}{\Delta x}
\]

is the Courant number. Since \( \theta \) in accord with (5.47) is conserved along the characteristic we get as a first guess

\[
\theta_j^{n+1} = \theta_Q^n = (1 - C)\theta_j^n + C\theta_{j-1}^n.
\]

Since the characteristic are the curves along which information is propagated (5.52) tells us that \( \theta_j^{n+1} \), that is, the value of \( \theta \) at the point \((x_j, t^{n+1})\) in time and space equals the value of \( \theta \) at the point \((x_Q, t^n)\) in time and space (cf. Figure 5.2).

5.6 Physical interpretation of the CFL condition

Figure 5.2 is drawn for \( u = u_0 > 0 \), and may be used to visualize the CFL criterion for the upwind scheme. First we note that since \( u_0 > 0 \) it follows from (5.48) that \( x_Q < x_j \). Moreover (5.48) also gives that the distance between \( x_Q \) and \( x_j \) is \( u \Delta t \). Thus if we additionally desire that \( x_{j-1} \leq x_Q \) then \( u \Delta t \leq \Delta x \). If we compare this result with the upwind scheme as given in (5.23) we observe that for \( u = u_0 > 0 \) the information used to compute \( \theta_j^{n+1} \) does originate from the two points \( \theta_j^n \) and \( \theta_{j-1}^n \). In fact we may rewrite (5.23) for \( u = u_0 > 0 \) to give

\[
\theta_j^{n+1} = (1 - C)\theta_j^n + C\theta_{j-1}^n, \quad C = \frac{u_0 \Delta t}{\Delta x}
\]
which matches (5.52) exactly. Thus from (5.53) follows that the upwind scheme may be interpreted as the value of $\theta$ at the time level $n+1$, that is, $\theta^n_{j+1}$, is found by a simple weighting of the values $\theta^n_j$ and $\theta^n_{j-1}$ using the Courant number as weight. What the method of characteristics (5.52) reveals is that the latter interpretation is only valid as long as $u \Delta t \leq \Delta x$. This is exactly what the CFL criterion demands in order to make the numerical upwind scheme stable, that is, the Courant number must be less than one or that (5.40) must be satisfied.

Moreover, if $u \Delta t > \Delta x$ then the characteristic through $x_j$ at time level $n+1$ (cf. Figure 5.2) will cross the time level $n$ to the left of $x_{j-1}$, that is, $x_{j-2} < x_Q < x_{j-1}$. Under these circumstances the upwind scheme will still use (5.53) to calculate $\theta$ at the new time level, that is, continue to use the weighted information using values at $x_j$ and $x_{j-1}$ at the previous time level. This is obviously wrong and use of (5.53) will lead to an major error. If this is allowed to continue for time step after time step the error accumulates and will finally give rise to a numerical instability.

The speed defined by $\Delta x$ and $\Delta t$, that is, $\Delta x/\Delta t$, is most often referred to as the signal speed of the grid. The CFL criterion (5.40) may therefore be interpreted as a condition which constrains the signal speed of the grid to be larger than the advection speed $u$. In other words, the advection speed must be small enough to let the area of dependence be between within $x_{j-1}$ and $x_{j+1}$ at time level $n$.

5.7 Numerical diffusion

Although the leapfrog scheme is neutrally stable we have just shown that it has a major disadvantage; it is dispersive. In particular, as displayed in Figure 5.5, this is true when the resolution is poor, that is, in areas where $\Delta x$ is inadequate to resolve the dominant wavelength. Also the impact of the dispersion increases with decreasing Courant.

As a result the upstream scheme was for a long time the favored advection scheme. Unfortunately also the upstream scheme is far from perfect. It contains what is referred to as numerical diffusion. The name derives from the fact that the numerical scheme inherently contains terms that acts in a fashion similar to physical diffusion, that is, tend to diminish differences in the tracer distribution. This is examplified in Figure 5.3 where an initial narrow, peak like tracer distribution spreads out while being advected. In contrast the analytic solution that the numerical solution tries to mimic is one in which the initial tracer distribution is advected without change. We underscore that this does not imply that any tracer is lost. The numerical diffusion process, just like its physical counterpart, does conserve the total tracer content. What happens is that the numerical diffusion smooth out any differences in the initial tracer concentration. Thus it redistributes the initial tracer distribution while conserving the initial total tracer content. This is evident in Figure 5.3. Comparing the area under the dash-dot curve and the area under the solid curve they are actually the same. Note again that this redistribution is artificial and arises due to the application of the upstream scheme.

To analyse the origin of the numerical diffusion in the upstream scheme let us reconsider
5.7. NUMERICAL DIFFUSION  \hspace{1cm} \textit{CHAPTER 5. THE ADVECTION PROBLEM}

Figure 5.3: Displayed is an example of the diffusion inherent in the upwind scheme. The solid curve shows the initial distribution at time level \( n = 0 \), while the dashed curve (red) shows the distribution at time level \( n = 200 \). The Courant number is \( C = 0.5 \). Cyclic boundary conditions are used at the boundaries of the computational domain.

(5.23). We first rewrite it in terms of an advective flux defined by

\[
F_j^n = \frac{1}{2} \left[ (u + |u|)\theta_j^n + (u - |u|)\theta_{j+1}^n \right] \frac{\Delta t}{\Delta x},
\]

(5.54)

We note that when \( u \geq 0 \) then \( u = |u| \) in which case \( F_j^n = \theta_j^n |u| \frac{\Delta t}{\Delta x} \). When \( u < 0 \) then \( u = -|u| \) in which case \( F_j^n = -\theta_{j+1}^n |u| \frac{\Delta t}{\Delta x} \). Thus (5.23) becomes

\[
\theta_j^{n+1} = \theta_j^n - (F_j^n - F_{j-1}^n),
\]

(5.55)

which is valid regardless of the sign of \( u \). If we substitute each of the terms in (5.55) by its associated Taylor series expansion, that is,

\[
\theta_j^{n+1} = \theta_j^n + [\partial_t \theta]_j^n \Delta t + \frac{1}{2} [\partial^2_t \theta]_j^n \Delta t^2 + \mathcal{O}(\Delta t^3)
\]

\[
\theta_{j\pm 1}^n = \theta_j^n \pm [\partial_x \theta]_j^n \Delta x + \frac{1}{2} [\partial^2_x \theta]_j^n \Delta x^2 \pm \mathcal{O}(\Delta x^3)
\]

(5.56)

and note that by use of (5.23) \([\partial^2_x \theta]_j^n = |u|^2 [\partial^2_x \theta]_j^n\), then we obtain

\[
[\partial_t \theta]_j^n = -u [\partial_x \theta]_j^n + \frac{1}{2} |u| (\Delta x - |u| \Delta t) [\partial^2_x \theta]_j^n + \mathcal{O}(\Delta x^2) + \mathcal{O}(\Delta t^2).
\]

(5.57)

Defining

\[
\kappa^* = \frac{1}{2} (1 - |C|) |u| \Delta x,
\]

(5.58)
where $C$ is the Courant number as defined in (5.40), (5.57) may be written
\begin{equation}
[\partial_t \theta]_j^n = -u [\partial_x \theta]_j^n + \kappa^* [\partial_x^2 \theta]_j^n + O(\Delta x^2) + O(\Delta t^2).
\end{equation}
(5.59)

Thus to second order in time and space we solve the following equation
\begin{equation}
\partial_t \theta + u \partial_x \theta = \kappa^* \partial_x^2 \theta.
\end{equation}
(5.60)

We recognize (5.60) as an advection-diffusion equation (cf. Chapter 3) with a diffusion coefficient $\kappa^*$ given by (5.58). The terms of order $O(\Delta x)$ and $O(\Delta t)$ which we neglected when employing the upstream scheme therefore give rise to a diffusion. This diffusion is unphysical and an artifact that appears due to the numerical method used. It is therefore a numerical diffusion inherent in the scheme. The strength of the numerical diffusion is determined by the diffusion coefficient defined in (5.58). We note that the diffusion is insignificant if the Courant number is close to or equals one. This corresponds to the upper limit of the CFL criterium (5.40) for stability, and is associated with a near neutrally stable scheme. The diffusion term also goes to zero when $\Delta x$ and $\Delta t$ goes to zero. In that sense the upstream scheme is consistent.

### 5.8 Minimizing numerical diffusion: Flux correction

In contrast to the second order leapfrog scheme the first order upwind (or upstream) scheme has the advantage that it is a positive definite scheme. Thus if the distribution of say $\theta(x,t)$ at some arbitrary time $t$ is such that $\theta \geq 0$ for all $x$ then also $\theta \geq 0$ for all later times $t = t + n\Delta t, \quad n = 1, 2, \ldots$. Another important property, as exemplified in Figure 5.3 (page 62), is that the position of the peak values are correctly propagated at any time without any dispersion. These are valuable properties well worth retaining in any scheme. The question arises if it is possible to construct a scheme that retains these properties while at the same time avoids, or at least minimizes, the numerical diffusion inherent in the scheme?

There are several schemes that offers a solution. Here we will present one of them called MPDATA\(^3\), a scheme first suggested by Smolarkiewicz (1983) (see also Smolarkiewicz and Margolin, 1997). The key element in MPDATA is to correct the diffusive flux inherent in upstream scheme. MPDATA therefore belongs to a class of schemes known as flux corrective schemes. To illustrate the method we first note that the advection equation (5.1) may be written
\begin{equation}
\partial_t \theta + \partial_x F_A = 0,
\end{equation}
(5.61)
where $F_A = u\theta$. As shown in Section 5.7 solving (5.61) using the upwind scheme results in a solution that satisfies (5.60) rather than (5.61). Thus, rather than solving (5.61) we appear to solve
\begin{equation}
\partial_t \theta + \partial_x (F_A + F_D) = 0,
\end{equation}
(5.62)

\(^3\)MPDATA is an abbreviation of “Multiple Positive Definite Advection-Transport Algorithm”
where \( F_D^* = -\kappa^* \partial_x \theta \) with \( \kappa^* \) given by (5.58). Thus the upwind scheme introduces an artificial or numerical diffusion represented by the diffusive flux \( F_D^* \). To avoid this unwanted diffusion Smolarkiewicz (1983) suggested to solve

\[
\partial_t \theta + \partial_x (F_A + F_A^*) = 0 \tag{5.63}
\]

rather than (5.61). Here \( F_A^* = u^* \theta \) is an artificially introduced advective flux where \( u^* \) is called the *antidiffusion velocity*. The idea is to let the artificially introduced advective flux \( F_A^* \) exactly oppose the diffusive flux \( F_D^* = -\kappa^* \partial_x \theta \) introduced by the upwind scheme. We achieve this if we let \( F_A^* = -F_D^* \), which conveniently defines the antidiffusive velocity by

\[
u^* \equiv \frac{\kappa^* \partial_x \theta}{\theta}. \tag{5.64}\]

To repeat, the idea of Smolarkiewicz (1983) is to introduce an artificial, advective flux that exactly compensate for the artificial diffusion generated by the first order upwind scheme. Theoretically adding the advective flux \( F_A^* = u^* \theta \) to the left-hand side of the continuous advection equation (5.61), as displayed in (5.63), exactly neutralizes the numerical diffusion arising when employing the upwind scheme. The introduction of the advective flux is therefore called a *flux correction method*.

We note that according to (5.64) \( u^* = 0 \) when \( \partial_x \theta = 0 \). Thus the propagation of the extreme values are not affected by adding the antidiffusive velocity, or corrective flux. The positions of the extrema after time \( t \) are therefore conserved. Moreover, we observe that \( u^* \) is proportional to \( \partial_x \theta \). It is therefore maximum where the gradient is steepest and its sign follows the sign of the slope. If for instance the distribution after some time is as displayed in Figure 5.3 (page 62), the effect of adding the artificial, advective flux is nil where the distribution has its maximum value. Thus the propagation of the maxima in the initial distribution is unaffected and its position is conserved. To the left of the “top” \( \partial_x \theta > 0 \). Hence, the added artificial flux enhances the natural flux there in an amount just right to offset the numerical diffusive flux. To the right of the top \( \partial_x \theta \) changes sign, and hence the antidiffusive flux changes sign as well. Thus, as expected, the antidiffusive flux helps to enhance the gradients, regardless of the sign of the slope, and the enhancement is most effective where \( \partial_x \theta \) is steepest. Moreover the antidiffusive flux is just right to neutralize the artificial or numerical diffusive flux introduced when using the upwind scheme. Finally we observe that when computing the antidiffusive velocity numerically we will have a problem if \( \theta \) tends to zero faster than \( \partial_x \theta \) in which case the antidiffusive velocity \( u^* \) tends to infinity. To avoid this Smolarkiewicz (1983) suggested to add a small number to the denominator in (5.64) when computing the antidiffusive velocity.

The numerical implementation suggested by Smolarkiewicz (1983) is equally simple. He suggested to perform the compensation in two steps by using a so called *predictor-corrector method*. In the first step, the *predictor step*, we compute a prediction \( \theta^* \) based on the true advection equation (5.61) using a low order advection algorithm\(^4\). Using the upwind scheme for this purpose we get

\[
\theta^*_j = \theta^n_j - \left[ (F_A)^n_j - (F_A)_{j-1}^n \right], \tag{5.65}\]

\(^4\)A low order scheme is an algorithm with a truncation error of \( O(\Delta t) \) and \( O(\Delta x) \).
where
\[
(F_A)_j^n = \frac{1}{2} \left[ (u_{j}^n + |u_{j}^n|)\theta_{j}^n + (u_{j}^n - |u_{j}^n|)\theta_{j+1}^n \right] \frac{\Delta t}{\Delta x}. \tag{5.66}
\]

We note that this step retains all the advantageous properties of the upwind scheme. We know, however, that the predictor solution \(\theta_j^*\) is “infected” by a numerical diffusion which in its continuous form is represented by a diffusive flux that reads \(F_D^* = -\kappa^* \partial_x \theta\), where \(\kappa^*\) is given (5.58). This causes the prediction \(\theta_j^*\) in general, and in particular for Courant numbers less than one, to appear less steep and generally smoother than its analytic or continuous counterpart. This is particularly evident in areas where the initial distribution has steep gradients as for instance visualized in Figure 5.3.

In the second step, the corrector step, we solve the advection equation (5.63) without the original advection term, that is
\[
\partial_t \theta + \partial_x (F_A^*)_j = 0, \tag{5.67}
\]
again using the low order upwind scheme. Hence the corrected solution \(\theta_j^{n+1}\) reads
\[
\theta_j^{n+1} = \theta_j^* - [(F_A^*)_j - (F_A^*)_{j-1}], \tag{5.68}
\]
where
\[
(F_A^*)_j = \frac{1}{2} \left[ (u_{j}^* + |u_{j}^*|)\theta_{j}^* + (u_{j}^* - |u_{j}^*|)\theta_{j+1}^* \right] \frac{\Delta t}{\Delta x}. \tag{5.69}
\]

To compute the antidiffusive velocity \(u_j^*\), we may for instance use a centered scheme when computing the gradient \(\partial_x \theta^*\). Thus using the analytic expression (5.64) we obtain
\[
u_{j}^* = \frac{1}{4} (1 - |C_{j}|) |u_{j}^n| \left( \frac{\theta_{j+1}^* - \theta_{j-1}^*}{\theta_j^* + \epsilon} \right). \tag{5.70}
\]
Note that we have added, as suggested by Smolarkiewicz (1983), a small number \(\epsilon\) in the denominator to ensure that \(u_j^*\) goes to zero when both \(\theta_j^*\) and \(\partial_x \theta_j^*\) goes to zero at the same time. If we make use of (5.70) to compute the antidiffusive flux its maximum values correspond to areas where the predictor slopes are maximum. Thus in the second step the gradients are restrengthened. Moreover, it has no effect where the predictor slopes are zero. Thus the position of the maximum is unchanged during the corrector step. As an example look at the dotted curve in Figure 5.3. If this was the predictive step the largest correction will be affected along the two flanks and thus steepen the diffused gradients. Note also that since the area under the curve is conserved when employing the upwind scheme, the maximum value increases during the corrector step. The solution therefore retains all the advantageous properties of the upwind scheme, and appears to avoid the artificial slackening of the steep gradients when applying the upwind scheme only. Moreover we observe that the corrector step makes the solution correct to \(O(\Delta x^2)\) and \(O(\Delta t^2)\). Hence MPDATA is a second order scheme that in theory compensates exactly for the artificial diffusive flux inherent in the lower order upwind scheme.

It should be noted, that since we employed an upwind scheme also in the corrector step the MPDATA data method also contains some artificial diffusion, however, small. This in turn may be minimized by rerunning the corrector step once more using the corrected solution as input.
5.9 Numerical dispersion

Figure 5.4: Solutions to the advection equation using the MPDATA scheme suggested by Smolarkiewicz (1983). Left panel corresponds to a scaling factor of 1.0 (no scaling), while the right-hand panel employs a scaling factor of 1.3. The Courant number is 0.5 in both cases. Solid, black lines show the initial value (time step $n = 0$), while the red dotted lines show the solution after 200 time steps. The green dashed lines are after 400 time steps while the blue, dash-dot lines are after 800 time steps. Cyclic boundary conditions are employed. (cf. Computer problem No. 6 in the Computer Problem notes).

This is the beginning of an iterative procedure where the number of iterative steps are determined by the required accuracy only.

A simpler and cheaper method (in terms of consumed computer time) is to slightly over-estimate the antidiffusive velocity by multiplying (5.70) by a scaling factor, a method already suggested by Smolarkiewicz (1983). Thus we redefine the antidiffusive velocity to read

$$u^*_j = \frac{1}{4} S_c (1 - |C^n_j|) |u^n_j| \left( \frac{\theta^*_j+1 - \theta^*_j-1}{\theta^*_j + \epsilon} \right),$$  

(5.71)

where $S_c$ is the scaling factor. As an example Figure 5.4 shows the solution to a pure advection equation employing MPDATA, in which the initial distribution is a narrow Gaussian bell function. In the upper panel the scaling factor is set to one (no scaling), while in the lower panel a scaling factor $S_c = 1.3$ is used.

5.9 Numerical dispersion

The concept of dispersivity is well known from other branches of physics and geophysics. In particular it is a common phenomenon regarding wave dynamics. By throwing stones in a still water most of us have indeed experienced it in practice. After the splash we observe that circular
waves propagates away from the original splash point in such a way that that longer waves leads progressively shorter waves. The reason for this is that the phase speed, say \( c \), depends on the wavenumber (or wavelength), that is, waves of different wavelengths propagate at different speeds. Regarding gravity waves in deep waters the longer the wavelength the faster the phase speed. Thus the longer waves will lead the progressively shorter waves. The same is also true for other types of waves for instance planetary Rossby waves.

Mathematically this is expressed through the dispersion relation \( \omega = \omega(\alpha) \) where \( \omega \) is the frequency and \( \alpha \) is the wavenumber. Recall that the phase speed is

\[
c = \frac{\omega}{\alpha}.
\]

If the frequency is a linear function of \( \alpha \), then the phase speed becomes a constant and all waves propagate at the same speed, that is, \( \partial_\alpha c = 0 \). The solution is then said to be non-dispersive. In the general case, however, \( \omega \) is a non-linear function of the wavenumber \( \alpha \). Then \( \partial_\alpha c \neq 0 \) and hence the phase speed depends on the wavelength. The solution is then said to be dispersive. Recall that the energy contained in the wave propagates with the group velocity defined by

\[
c_g = \partial_\alpha \omega = \alpha \partial_\alpha c + c.
\]

Hence if the wave is non-dispersive \( c_g = c \), that is, the energy contained in the wave propagates at the same speed as the wave itself. On the other hand if the wave is dispersive then both the phase speed and the group velocity depends on the wavelength. Moreover if \( \partial_\alpha c < 0 \), which is the case for gravity waves, then the waves travel at speeds faster than their group velocity. Thus gravity waves tend to travel faster than their energy is propagated\(^5\).

If we apply a wave solution to the advection equation (5.1), that is, \( \theta = \Theta e^{i\alpha(x-ct)} \), we find that the phase speed is \( c = u \). Thus if \( u = u_0 \), that is, is uniform in time and space the phase speed becomes constant. Under these circumstances all the waves propagate with the same velocity, namely \( u_0 \), and hence the true solution to the advection equation is non-dispersive. The question then arises. Is this true for the numerical solution?

To investigate this we apply a similar analysis based on the finite difference approximation to the advection equation (5.1). Let us consider the leapfrog scheme (5.22). We know that this scheme is neutrally stable. Furthermore, as long as all the gradients are well resolved by our grid it is a superb scheme in the sense that it is a stable and consistent scheme with no numerical dissipation or diffusion. But what about the dispersivity? To get started we first substitute the discrete Fourier component

\[
\theta_j^n = \Theta_0 e^{i\alpha(j\Delta x - cn\Delta t)},
\]

into (5.22). This gives

\[
2i \sin(\alpha c \Delta t) = -2i u_0 \frac{\Delta t}{\Delta x} \sin(\alpha \Delta x)
\]

which is written

\[
c = \frac{1}{\alpha \Delta t} \arcsin \left( u_0 \frac{\Delta t}{\Delta x} \sin \alpha \Delta x \right).
\]

\(^5\)For capillary waves \( \partial_\alpha c > 0 \) and hence a capillary wave travels at a speed slower than its energy.
Obviously $\partial_\alpha c \neq 0$, and hence the leapfrog scheme is dispersive. Since this dispersivity is due to the finite difference approximation, that is, associated with our numerical solver, we commonly refer to this as *numerical dispersion*. We note that as $\Delta x$ and $\Delta t$ goes to zero, but such that the stability condition is satisfied, that is,

$$C \leq 1 \quad \text{or} \quad u_0 \Delta t \leq \Delta x$$

(5.77)

then $c \to u_0$ and the wave becomes non-dispersive.

![Dispersion relation](dispersion relation)

**Figure 5.5:** Numerical dispersion for the leapfrog scheme. The curves depict the numerical phase speed as a function of the wavenumber based on (5.76) for various values of the Courant number $C = u \Delta t / \Delta x$. The vertical axis indicates the phase speed $c$ normalized by the advection speed $u$. The horizontal axis indicates the wavenumber normalized by $\pi / \Delta x$ where $\Delta x$ is the space increment or the grid size. The analytic dispersion curve is just a straight line corresponding to the phase speed $c = u$, that is $c/u = 1$. Note that as the wavenumber increases (that is the wavelength decreases) the numerical phase speed deviates more and more from the correct analytic phase speed for all values of the Courant number. For wave numbers which gives $\alpha \Delta x / \pi > 0.5$, that is for waves of wavelengths $\lambda < 4 \Delta x$ the slope of the curves indicates that the group velocity is negative. Thus for waves with wavelengths shorter than $4 \Delta x$ the energy is propagating in the opposite direction of the waves.

This is visualized in Figure 5.5 showing the normalized phase speed as a function of the normalized wave number for various Courant numbers. The figure clearly exhibit the dispersive nature of the leapfrog scheme. We also observe from Figure 5.5, as was first noted by Grotjhan and O’Brien (1976), that the dispersivity gets worse the less the Courant number. In fact from (5.73) follows that $c_g$ becomes zero when

$$\alpha \partial_\alpha c = -c.$$  

(5.78)
By use of (5.76) it follows that this is true for wavenumbers satisfying \( \cos(\alpha \Delta x) = 0 \), that is, for wavenumbers
\[
\alpha_m = \frac{1}{2} (2m - 1) \pi, \quad m = 1, 2, \ldots
\] (5.79)

We therefore conclude that the longest wave for which \( c_g = 0 \) is for \( m = 1 \), that is \( \lambda = 2\pi/\alpha_1 = 4\Delta x \). As displayed in Figure 5.5 this corresponds to the normalized wavenumber \( \alpha' = 0.5 \). For higher wavenumbers, that is, waves whose wavelength are shorter than \( 4\Delta x \) the group velocity actually becomes negative. Thus if the wave is poorly resolved the leapfrog scheme will actually propagate energy opposite to the wave itself. This is clearly unphysical and must be avoided.

It is therefore extremely important that the scales that dominates the property that is advected is well resolved. Let the dominant wavenumber be \( \alpha \). Then by looking at Figure 5.5 \( \Delta x \) must be chosen so that \( \alpha \Delta x < 0.3\pi \) for Courant numbers close to one and even less if the Courant number is smaller.

### 5.10 Unphysical solutions and numerical modes

In Section 5.7 and Section 5.9 we showed that two of the most popular numerical schemes for solving the advection equation, namely the upwind scheme and the leapfrog scheme, harbored some unwanted properties. While the upwind or upstream scheme was shown to contain numerical diffusion, the leapfrog scheme turned out to contain numerical dispersion. Although these properties, as their name suggests, have a physical interpretation they are nonetheless results of the employed scheme and hence unphysical or a numerical artifact. In Section 5.8 we showed how the numerical diffusion to some extent could be avoided by use of flux correction. Below (Section 9.4 on page 120) we will show that there exists a method called the spectral method which avoids the dispersion inherent in the leapfrog scheme. However, this method is applicable only for global models. For limited area models there is no such remedy available.

We will now show that the leapfrog scheme contains an additional unphysical property, namely false numerical modes. Let us start by assuming that the initial condition is as given in (5.4), that is, a harmonic wave with wave number \( \alpha \) and amplitude \( \theta_0 \). The true solution is then given by (5.5), that is, the solution is a single monochromatic wave with wavenumber \( \alpha \) propagating with the phase speed \( c = u_0 \) in the positive \( x \) direction.

To reveal that the leapfrog scheme actually contains two modes we solve the leapfrog scheme analytically. We start by assuming that the initial condition is a monochromatic wave with wavenumber \( \alpha \) as given by (5.4). Recalling from Section 4.2 (page 33) that any good function can be written in terms of an infinite sum of waves, we note that if we can find the solution for one monochromatic wave we find the solution to any arbitrary initial condition by summing up (in wavenumber space) over all possible wavenumbers. Furthermore we recall from the stability analysis of Section 4.3 on page 34 using induction that the solution may be written
\[
\theta_j^n = G^n \Theta_0 e^{i\alpha_j \Delta x}.
\] (5.80)

where \( G \) is the growth factor and \( \Theta_0 \) is the initial amplitude. We underscore that \( G \) in general is a complex number. Recall that the growth factor for the leapfrog scheme actually has two
solutions for the growth factor as given by (5.37) on page 56. For our purpose we rewrite these two solutions as

\[ G_1 = g_r - ig_i, \quad \text{and} \quad G_2 = -(g_r + ig_i), \quad (5.81) \]

where the real and imaginary parts are given by

\[ g_r = \sqrt{1 - \lambda^2} \quad \text{and} \quad g_i = \lambda, \quad (5.82) \]

respectively, and where \( \lambda \) is given by (5.36). Note that we have assumed that the CFL condition for stability is satisfied so that \( g_r \) is a real number. Recalling that any complex number \( P = a + ib \) may be written

\[ P = |P|e^{i\phi} \quad (5.83) \]

we note that \( |G_{1,2}| = 1 \) and that the two solutions becomes

\[ G_1 = e^{-i\alpha c \Delta t} \quad \text{and} \quad G_2 = (-1)e^{i\alpha c \Delta t} \quad (5.85) \]

where \( c = c(\alpha) \) is the dispersive phase speed defined in (5.76). Substituting these solutions into (5.80) we obtain

\[ \theta_j^n = \Theta_1 e^{i\alpha(j\Delta x - cn\Delta t)} + (-1)^n\Theta_2 e^{i\alpha(j\Delta x + cn\Delta t)}, \quad (5.86) \]

where \( \Theta_1 \) and \( \Theta_2 \) are as yet two unknown constants. Since we have only one initial condition at our disposal, we can only obtain a relationship between them. Utilizing the initial condition (5.4) we find

\[ \theta_j^n = (\Theta_2 - \Theta_0)e^{i\alpha(j\Delta x - cn\Delta t)} + (-1)^n\Theta_2 e^{i\alpha(j\Delta x + cn\Delta t)}. \quad (5.87) \]

We note that the first term on the right-hand side of (5.87) is a wave propagating in the positive \( x \) direction with phase speed \( c \) and amplitude \( \Theta_2 - \Theta_0 \). In contrast the second term on the right-hand side is a wave propagating in the negative \( x \) direction, but with the same phase speed. Furthermore, the latter has an amplitude alternating between \( \pm \Theta_2 \). We conclude that by making use of the leapfrog scheme, the finite difference solution contains two solutions in the form of two waves propagating in opposite directions with the dispersive phase speed given by (5.76). In contrast the true solution (5.5) to the advection equation contains only one wave that propagates in the positive \( x \) direction with phase speed \( u_0 \). The two waves that occur in the finite difference solution is due to the fact that the leapfrog scheme is of second order. As such it requires us to give two boundary conditions in time. We are therefore in need of an additional condition to determine the remaining constant.

The latter problem is associated with the initial boundary problem discussed in Section 5.4 above, but manifests itself in the unknown constant \( \Theta_2 \) in (5.87). One remedy suggested in Section 5.4 is to apply an Euler step as the first step. Note that this was suggested to start the time marching problem, otherwise we had to assign a value to \( \theta \) at the time step prior to the initial time. We recall that the Euler step (5.44) is

\[ \theta_j^1 = \theta_j^0 - u_0 \frac{\Delta t}{2\Delta x} (\theta_{j+1}^0 - \theta_{j-1}^0). \quad (5.88) \]
Substituting the initial condition (5.4) into (5.88) and (5.87) (letting \( n = 1 \)) the additional condition becomes

\[
\Theta_0[1 - i \sin(\alpha c \Delta t)]e^{i\alpha j \Delta x} = (\Theta_2 - \Theta_0)e^{-i\alpha c \Delta t} - \Theta_2 e^{i\alpha c \Delta t}
\]

and hence that

\[
\Theta_2 = \Theta_0 \frac{1 - \cos(\alpha c \Delta t)}{2 \cos(\alpha c \Delta t)}.
\]

The complete finite difference solution is then

\[
\theta_j^n = \Theta_0 \left\{ \frac{1 + \cos(\alpha c \Delta t)}{2 \cos(\alpha c \Delta t)} e^{i\alpha (j \Delta x - cn \Delta t)} + (-1)^n \frac{1 - \cos(\alpha c \Delta t)}{2 \cos(\alpha c \Delta t)} e^{i\alpha (j \Delta x + cn \Delta t)} \right\}.
\]

We observe, as mentioned in the previous Section 5.9, that as long as the stability criteria (5.40) is satisfied then \( c \to u_0 \) when \( \Delta t \) and \( \Delta x \) goes to zero independently. Underthese circumstances we take note that \( [1 + \cos(\alpha c \Delta t)]/2 \cos(\alpha c \Delta t) \to 1 \) and \( (1 - \cos(\alpha c \Delta t))/2 \cos(\alpha c \Delta t) \to 0 \). Thus as we allow \( \Delta t \to 0 \) and \( \Delta x \to 0 \) the first term on the right-hand side of (5.91) approaches the true solution while the second term vanishes. We conclude therefore that the second term is an unphysical mode while the first term is the physical mode. The occurrence of the two modes in the leapfrog scheme are sometimes referred to as “time splitting” in the literature.

It is of utmost importance that we do control the unphysical mode. Unless we do it may create a lot of noise in our calculations, or in extreme cases even numerical instabilities. This is particularly true since it alternates between attaining positive and negative values.

### 5.11 How to get rid of the unphysical mode: The Asselin filter

The simplest way to get rid of the numerical or unphysical mode in the leapfrog scheme is, from time to time, to make use of a one-sided, forward in time, centered in space step from time to time as outlined in (5.15), that is, perform an Euler step. Although, as shown in Section 5.1, the Euler scheme is numerically unstable, we may still apply it for single time steps without destroying the stability of for instance the leapfrog scheme.

Another method, originally suggested by Robert (1966) and further developed by Asselin (1972), is to apply a time filtering technique. That a time filter will work the trick is obvious since the unphysical mode alternates between positive and negative values from time step to time step. We start by remarking that in general a time filter invoking the neighboring time levels only may be written

\[
\overline{\theta}(x, t) = \gamma \theta(x, t + \Delta t) + (1 - 2\gamma)\theta(x, t) + \gamma \theta(x, t - \Delta t),
\]

where \( \overline{\theta}(x, t) \) is the filtered function and \( \gamma \) is a weighting parameter. By use of the notation introduced in Section 2.8 we may rewrite (5.92) to yield

\[
\overline{\theta}_j^n = \theta_j^n + \gamma [\theta_j^{n+1} - 2\theta_j^n + \theta_j^{n-1}].
\]

Note that from (5.75) follows \( \lambda = u_0 \Delta t \sin(\alpha \Delta x) = \sin(\alpha c \Delta t) \).
Note that if $\gamma = 0$ we retrieve the original function, while for $\gamma = \frac{1}{4}$ the filter is the standard 1-2-1 filter,

$$\overline{\theta}_j^n = \frac{1}{4}[\theta_j^{n+1} + 2\theta_j^n + \theta_j^{n-1}], \quad (5.94)$$

which gives twice the weight to the mid time level $n$. To investigate the properties of the filter let us study one period or frequency $\omega$ only. We start by representing the function $\theta$ by its Fourier component

$$\theta_j^n = \hat{\theta}_j e^{i\omega n \Delta t}. \quad (5.95)$$

Substituting this into (5.93) gives

$$\overline{\theta}_j^n = R\theta_j^n, \quad (5.96)$$

where the ratio

$$R(\gamma) = \frac{\overline{\theta}_j^n}{\theta_j^n} = 1 - 2\gamma + 2\gamma \cos \omega \Delta t \quad (5.97)$$

is the response function of the filter. For the 1-2-1 filter, that is, for $\gamma = \frac{1}{4}$, the response function becomes

$$R\left(\frac{1}{4}\right) = \frac{1}{2}(1 + \cos \omega \Delta t) \quad (5.98)$$

Since the period is $T = 2\pi/\omega$ we notice that for the standard 1-2-1 filter $R = 0$ for waves of period $T = 2\Delta t$, while $R = \frac{1}{2}$ for waves of period $T = 4\Delta t$. Thus waves or noise on the Nyquist frequency, that is, waves of periods $2\Delta t$, cannot exist and hence vanish. Since the unphysical mode inherent in the leapfrog scheme alternates between attaining negative and positive values from time step to time step, its dominant wave period is exactly $2\Delta t$. Hence the 1-2-1 filter is perfect in this sense. Noise of scales close to slightly longer periods, say $4\Delta t$, is damped to half of their original energy. The effect on the longer periods is minimal. It was these advantageous properties of the 1-2-1 filter that lead Robert (1966) and Asselin (1972) to suggest to use this method to damp the unphysical mode inherent in the leapfrog scheme.

In practice we do this as follows. Let us assume that the filtered solution has been determined for time level $n - 1$, that is, assume that $\overline{\theta}_j^{n-1}$ is known and has been stored. Let us furthermore assume that the unfiltered value at time level $n$, that is, $\theta_j^n$, has been stored as well. We then first apply the leapfrog scheme to compute the function $\theta$ at the new time level $n + 1$, that is, $\theta_j^{n+1}$, using the unfiltered values of $\theta$ at time level $n$ and the filtered values of $\theta$ at time level $n - 1$. For instance, if we consider the simple advection equation (5.1) and make use of the leapfrog scheme (5.22) to advance to the next time level we obtain

$$\theta_j^{n+1} = \theta_j^n - u \frac{\Delta t}{\Delta x} (\theta_j^{n+1} - \theta_j^{n-1}). \quad (5.99)$$

Note that we in this first step have used the filtered value at time level $n - 1$ to compute $\theta_j^{n+1}$. The next step is to compute the filtered values at time level $n$ using the filter (5.93), that is,

$$\overline{\theta}_j^n = \theta_j^n + \gamma \left[ \theta_j^{n+1} - 2\theta_j^n + \overline{\theta}_j^{n-1} \right]. \quad (5.100)$$
In the third and final step we replace the filtered values at time level $n - 1$ by the new filtered values at time level $n$. We can safely do this because after we have computed the new filtered values at time level $n$ the filtered values at time level $n - 1$ are obsolete. We may now proceed to the next time level to compute $\theta_j^{n+2}$, and so on.

We take note, however, that the Asselin filter affects the numerical stability and that it produces numerical diffusion. It is possible to show that while the numerical diffusion increases with increasing values of the weighing parameter $\gamma$ the critical value for stability decreases. The latter implies that the stability criterion becomes more strict and that we have to diminish the time step $\Delta t$. This fact entails that although we wish to employ the simple 1-2-1 filter since it exactly kills the unphysical mode, it becomes unstable unless we decrease the time step. Decreasing the time step in turn means that our computations becomes less efficient. It is therefore common to apply a lower value for the weighing function, say $\gamma = 0.08$. Note that even a weak Asselin filter eventually modifies the longer wave periods by diffusion. Hence we must apply the Asselin filter with care and not necessarily for every time step.

**Exercises**

1. Show that the CFL criterion for the leapfrog scheme, the diffusive scheme and the upwind scheme all are given by (5.40).

2. Show that (5.86) is a solution to (5.22). Moreover, show that (5.91) follows from (5.86) when the initial distribution is given by (5.88), and where (5.15) is made use of to find $\theta_j^{-1}$. Hint: Show first that $G_{1,2}$ from (5.37) may be written

\[ G_1 = e^{-i\chi}, \quad G_2 = e^{i(\chi+\pi)} \]  

where $\chi$ is given by (5.87).
Chapter 6

THE SHALLOW WATER PROBLEM

We have argued above that two of the most important processes in the atmosphere and ocean is advection and mixing (diffusion). This is definitely true regarding tracer transport and spreading, which is indeed governed by advection-diffusion equations. An important variable in these equations is the velocity by which the tracers are advected. In the atmosphere and ocean this is determined using the momentum equations, e.g., (1.1) presented in Chapter 1 on page 2. Thus we enter the realm of atmosphere-ocean dynamics. An important part of these dynamics, which make atmosphere-ocean dynamics stand out from ordinary fluid dynamics, is the effect of the Earth’s rotation. In addition, dynamics contains multiple dependent variables and is therefore a convenient way of introducing methods whereby partial differential equations containing more than one dependent variable may be solved numerically.

Following Albert Einstein’s mantra of making things as simple as possible, but no simpler, we investigate the numerical solution to a particular simplified set of the momentum equations, namely the shallow water equations and some subsets thereof. These equations are simple, yet they include the essence of atmosphere-ocean dynamics. Moreover they are complex enough to appreciate the methods whereby the complete momentum equations are solved numerically. One of the main reasons for this is that the fully three-dimensional, barotropic/baroclinic equations of motion can be described in terms of so called vertical normal modes, where each mode is governed by a set of shallow water equations. If we for instance discretize a numerical model into say \( N \) vertical modes we get one set of equations for each vertical mode, that is, \( N \) set of equations. Each of these sets has a so called “equivalent depth” (or equivalent geopotential height) corresponding roughly to the height of the coordinate surface above ground/bottom. An illustrative example is the motion of a barotropic atmosphere or ocean\(^1\), which is the first and foremost vertical normal mode. The motion of such a fluid is reasonably well described by the shallow water equations presented in Section 1.5 on page 6, that is, by equations (1.26) and (1.27) (or eqs. 1.29 and 1.30) on page 7.

In summary the shallow water equations are a particularly suitable set with regard to describing fundamental properties of the atmosphere-ocean dynamics. For one it highlights the importance of geostrophy (cf. Section 1.6 on page 8) or the so called geostrophic balance\(^2\), a

\(^1\)Recall that for a barotropic fluid density and pressure surfaces are concurrent

\(^2\)The geostrophic balance is a balance between the Coriolis acceleration term and the pressure forcing.
very stringent and important dynamic balance that constrains atmosphere-ocean dynamics.

We start by rewriting (1.29) and (1.30) replacing \( \hat{u} \) by \( u \). In scalar form we get,

\[
\begin{align*}
\partial_t u + u \partial_x u + v \partial_y u - fu &= -\partial_x \phi, \\
\partial_t v + u \partial_x v + v \partial_y v + fu &= -\partial_y \phi, \\
\partial_t \phi + u \partial_x \phi + v \partial_y \phi &= -\phi (\partial_x u + \partial_y v),
\end{align*}
\]

where we have used the geopotential \( \phi = gh \) to replace the thickness (or geopotential height) \( h \) and where we have neglected the forcing terms and friction. We notice that the system (6.1) - (6.3) is non-linear, and that it contains multiple dependent variables as well as multiple independent variables. In fact it contains three dependent variables, namely the two horizontal, depth integrated, velocity components \( u, v \) and the geopotential \( \phi \). The multiple independent variables are \( x, y, t \). The shallow water equation are therefore not only a set of equations that describe important and fundamental aspects of the atmosphere-ocean dynamics. They are also a convenient set of equations whereby the numerical treatment of equations containing more than one dependent variable as well as several independent variables may be introduced within a geophysical fluid context. Finally we observe that, mathematically speaking, the system (6.1) - (6.3) is complete in the sense that we have three equations to solve for the three unknowns \( u, v, \phi \).

In the following we will also further simplify the equations above by assuming the the motion is one-dimensional so that \( \partial_y = 0 \). Under these circumstances (6.1) - (6.3) becomes

\[
\begin{align*}
\partial_t u + u \partial_x u - fu &= -\partial_x \phi, \\
\partial_t v + u \partial_x v + fu &= 0, \\
\partial_t \phi + u \partial_x \phi &= -\phi \partial_x u,
\end{align*}
\]

Since there is a fundamental difference between linear and non-linear systems, we divide the presentation below into linear and non-linear versions of the shallow water equations. Furthermore, to highlight some salient fact of importance when solving the shallow water equations by numerical means, we also separate between non-rotating and rotating systems.

### 6.1 Linearization

To linearize the shallow water equations we start by assuming that the dependent variables can be written in terms of a time independent, basic state plus a perturbation, that is,

\[
\begin{align*}
uu &= \bar{u} + u', \quad v = \bar{v} + v' \quad \text{and} \quad \phi = \bar{\phi} + \phi',
\end{align*}
\]

where the perturbed velocity (or basic state) is \( \bar{u} = \bar{u}i + \bar{v}j \) and the perturbed geopotential is \( \bar{\phi} \). Substituting these expressions into (6.1) - (6.3), noting that \( \partial_t \bar{u} = \partial_t \bar{v} = \partial_t \bar{\phi} = 0 \) and neglecting
terms containing products of perturbations, we get

\[
\begin{align*}
\partial_t u &+ \bar{u}\partial_x \bar{u} + \bar{u}\partial_x u + u\partial_x \bar{u} + \bar{v}\partial_y \bar{u} + v\partial_y u - f\bar{v} - f v = -\partial_x \bar{\phi} - \partial_x \phi, \\
\partial_t v &+ \bar{u}\partial_x \bar{v} + \bar{u}\partial_x v + u\partial_x \bar{v} + \bar{v}\partial_y \bar{v} + v\partial_y v + f\bar{u} + f u = -\partial_y \bar{\phi} - \partial_y \phi, \\
\partial_t \phi &+ \bar{u}\partial_x \bar{\phi} + \bar{u}\partial_x \phi + u\partial_x \bar{\phi} + v\partial_y \bar{\phi} + \bar{v}\partial_y \phi + v\partial_x \bar{\phi} = -\bar{\phi}\partial_x \bar{u} - \bar{\phi}\partial_x u - \phi'\partial_x \bar{u} - \bar{\phi}\partial_y \bar{v} - \bar{\phi}\partial_y v - \phi'\partial_y \bar{v}
\end{align*}
\]

(6.8)

where we have dropped the primes on the perturbation quantities for clarity.

If we let the basic state be one at the rest, that is, let \( \bar{u} = \bar{v} = 0 \) and \( \bar{\phi} = \bar{\phi}_0 = c_0^2 \), where \( c_0 = \sqrt{gH_0} \) and \( H_0 \) is a constant geopotential height, we arrive at a particular simple linear subset, namely,

\[
\begin{align*}
\partial_t u - fv &= -\partial_x \phi, \\
\partial_t v + fu &= -\partial_y \phi, \\
\partial_t \phi &= -c_0^2 (\partial_x u + \partial_y v)
\end{align*}
\]

(6.11-6.13)

If we in addition require that the ensuing motion has reached a steady state (\( \partial_t = 0 \)) we get

\[
\begin{align*}
v &= \frac{1}{f}\partial_x \phi \quad \text{and} \quad u = -\frac{1}{f}\partial_y \phi.
\end{align*}
\]

(6.14)

The steady state solution is therefore on in geostrophically balance. We note that under these circumstances (6.13) is trivially satisfied and is therefore obsolete.

We may derive another special subset by assuming that the basic state is in geostrophic balance, that is,

\[
\begin{align*}
\bar{v} &= \frac{1}{f}\partial_x \bar{\phi} \quad \text{and} \quad \bar{u} = -\frac{1}{f}\partial_y \bar{\phi}.
\end{align*}
\]

(6.15)

Substituting (6.15) into (6.8) - (6.10) we then get

\[
\begin{align*}
\partial_t u + \bar{u}\partial_x u + \bar{v}\partial_y u - f v &= -\partial_x \phi, \\
\partial_t v + \bar{u}\partial_x v + \bar{v}\partial_y v + f u &= -\partial_y \phi, \\
\partial_t \phi + \bar{u}\partial_x \phi + \bar{v}\partial_y \phi &= -\bar{\phi}(\partial_x u + \partial_y v).
\end{align*}
\]

(6.16-6.18)

As we have done previously we may further simplify these systems by assuming that the problem is one-dimensional. This is achieved by letting \( \partial_y = 0 \). For the first set (6.11) - (6.13) we then get

\[
\begin{align*}
\partial_t u - fv &= -\partial_x \phi, \\
\partial_t v &= 0, \\
\partial_t \phi &= -c_0^2 \partial_x u.
\end{align*}
\]

(6.19-6.21)
For the second set (6.16) - (6.18) we similarly get
\[ \begin{align*}
\partial_t u + \bar{u} \partial_x u - f v &= -\partial_x \phi, \\
\partial_t v + \bar{u} \partial_x v + f u &= 0, \\
\partial_t \phi + \bar{u} \partial_x \phi + \bar{\phi} \partial_x u &= 0,
\end{align*} \]
(6.22)
(6.23)
(6.24)

6.2 Linear, non-rotating shallow water equations

We start by considering a particularly simple subset namely the non-rotating \((f = 0)\) version of (6.19) - (6.21), that is,
\[ \begin{align*}
\partial_t u &= -\partial_x \phi, \\
\partial_t \phi &= -c_0^2 \partial_x u.
\end{align*} \]
(6.25)
(6.26)

We note that under these circumstances \(v = \text{constant} = 0\), and hence we are left with two equations for the two unknowns \(u\) and \(\phi\). We observe that the system is hyperbolic. This is for instance easily shown if we substitute (6.25) into (6.26). We then get
\[ \begin{align*}
\partial_t^2 \phi - c_0^2 \partial_x^2 \phi &= 0,
\end{align*} \]
(6.27)
which is the linear wave equation.

To explore the solutions to (6.25) and (6.26) we investigate one Fourier component. Thus we assume that the solution may be written in forms of waves, viz.,
\[ \begin{align*}
\phi &= \phi_0 e^{i\alpha(x-ct)} \quad \text{and} \quad u = u_0 e^{i\alpha(x-ct)}
\end{align*} \]
(6.28)
where \(u_0\) and \(\phi_0\) are arbitrary wave amplitudes and \(c = \omega/\alpha\) is the wave speed. Substituting (6.28) into (6.25) and (6.26) and tossing away common non-zero factors we get
\[ \begin{align*}
-cu + \phi &= 0, \\
c_0^2 u - c\phi &= 0.
\end{align*} \]
(6.29)
(6.30)

Defining a vector \(x\) by
\[ x = \begin{bmatrix} u \\ \phi \end{bmatrix}, \]
(6.31)
we may formulate (6.29) and (6.30) as the homogeneous linear equation,
\[ \mathbf{A} \cdot \mathbf{x} = 0, \]
(6.32)
where the tensor \(\mathbf{A}\) is
\[ \mathbf{A} = \begin{bmatrix} -c & 1 \\ c_0^2 & -c \end{bmatrix}. \]
(6.33)
For non-trivial solutions to exist, the determinant of the tensor \(\mathbf{A}\) must be zero, which gives the following dispersion relation
\[ c^2 - c_0^2 = 0. \]
(6.34)
This equation has two solutions for the wave speed \( c \), namely

\[
c_1 = -c_0 \quad \text{and} \quad c_2 = +c_0. \tag{6.35}
\]

Thus the true solution is

\[
\phi = \phi_1 e^{i\alpha(x-c_0 t)} + \phi_2 e^{i\alpha(x+c_0 t)} \tag{6.36}
\]

that is, two waves propagating with a constant phase speed \( c_0 \) in opposite directions\(^3\).

Our task is to solve (6.25) and (6.26) numerically to give a solution that mimics the true solution. As alluded to in Section 2.4 the advection equation bears a strong resemblance to the wave equation. We therefore expect the leapfrog scheme, that is, a centered in time, centered in space (CTCS) scheme to work well. Using centered finite difference approximations for the differential terms we get

\[
\frac{u_{j}^{n+1} - u_{j}^{n-1}}{2\Delta t} + \frac{\phi_{j}^{n+1} - \phi_{j}^{n-1}}{2\Delta x} = 0, \tag{6.37}
\]

\[
\frac{\phi_{j}^{n+1} - \phi_{j}^{n-1}}{2\Delta t} + \frac{u_{j+1}^{n} - u_{j-1}^{n}}{2\Delta x} = 0. \tag{6.38}
\]

Since we made use of Taylor series to derive the approximation we know apriori that the scheme is consistent. To satisfy ourselves that the system is stable we use von Neumann’s analysis method. Thus we start by substituting the discrete Fourier components

\[
u_{j}^{n} = U_{n} e^{i\alpha j \Delta x} \quad \text{and} \quad \phi_{j}^{n} = g H_{n} e^{i\alpha j \Delta x} \tag{6.39}
\]

into (6.37) and (6.38). We then get

\[
U_{n+1} - U_{n-1} = -2ig \frac{\Delta t}{\Delta x} H_{n} \sin \alpha \Delta x, \tag{6.40}
\]

\[
H_{n+1} - H_{n-1} = -2iH_{0} U_{n} \sin \alpha \Delta x. \tag{6.41}
\]

Since \( n \) is only a time step counter (6.40) gives

\[
U_{n+2} - U_{n} = -2ig \frac{\Delta t}{\Delta x} H_{n+1} \sin \alpha \Delta x, \tag{6.42}
\]

and similarly

\[
U_{n} - U_{n-2} = -2ig \frac{\Delta t}{\Delta x} H_{n-1} \sin \alpha \Delta x. \tag{6.43}
\]

We then add (6.42) and (6.43) to get

\[
U_{n+2} - 2U_{n} + U_{n-2} = -2ig \frac{\Delta t}{\Delta x} (H_{n+1} - H_{n-1}) \sin \alpha \Delta x. \tag{6.44}
\]

Finally we substitute for \( (H_{n+1} - H_{n-1}) \) using (6.41) to get

\[
U_{n+2} - 2\lambda U_{n} + U_{n-2} = 0 \tag{6.45}
\]

\(^3\)Note that \( u \) has a similar solution
where
\[ \lambda = 1 - 2c_0^2 \left( \frac{\Delta t}{\Delta x} \right)^2 \sin^2 \alpha \Delta x. \] (6.46)

Next we define a growth factor \( G \) by \( U_{n+2} = GU_n \). Substitution into (6.45) then results in a second order equation to solve for the growth factor. Solving it we get the two solutions
\[ G_{1,2} = -\lambda \pm \sqrt{\lambda^2 - 1} = -\lambda \pm i \sqrt{1 - \lambda^2}. \] (6.47)

Thus the growth factor is complex and has an imaginary part if and only if \( \lambda^2 \leq 1 \). Under these circumstances
\[ |G_{1,2}| = \sqrt{\lambda^2 + 1 - \lambda^2} = 1, \] (6.48)

and the scheme is neutrally stable. Recall that this result depends on the fact that the radical in (6.46) is positive definite. Thus stability is ensured if and only if \(-1 \leq \lambda \leq 1 \). The right-hand inequality is trivially satisfied, but the left-hand one requires
\[ c_0^2 \left( \frac{\Delta t}{\Delta x} \right)^2 \leq 1 \Rightarrow |C| = \left| c_0 \right| \frac{\Delta t}{\Delta x} \leq 1, \quad \text{or} \quad \Delta t \leq \frac{\Delta x}{|c_0|}, \] (6.49)

where \( C \) is the Courant number. Thus, just as when applying the leapfrog scheme for the simple advection equation (cf Section 5.3) we arrive at the result that the scheme is neutrally stable (contains no dissipation) and that the Courant number has to be less than or equal to one. We observe, however, that the velocity that enters the definition of the Courant number is the phase speed \( c_0 \) rather than advection velocity \( u_0 \). Since the phase speed normally is much larger than the advective velocity, that is, \(|c_0| >> |u_0|\), it follows that the constraint on the time step is much more stringent for the wave equation than for the advection equation.

### 6.3 Staggered grids

We note that the system (6.25) and (6.26) contains four integration constants; two in time and two in space. The system therefore only allows us to specify two boundary condition in \( x \). Let us assume that we are to solve (6.25) and (6.26) for \( x \in 0, L > \) and for \( t \in 0, T \). We then have three options. We may (i) specify two conditions at \( x = 0 \), (ii) two condition at \( x = L \) or (iii) one condition at \( x = 0 \) and one condition at \( x = L \). Let us assume that the fluid is contained between two solid walls. The natural condition is then to put \( u = 0 \) at \( x = 0, L \) (impermeable walls). Numerically this translates to specifying that \( u_{i}^n = 0 \) and \( u_{j}^n = 0 \). To show that this causes a problem we start by rewriting (6.37) and (6.38) to give
\[ u_{j}^{n+1} = u_{j}^{n-1} - \frac{\Delta t}{\Delta x} \left( \phi_{j+1}^n - \phi_{j-1}^n \right), \] (6.50)
\[ \phi_{j}^{n+1} = \phi_{j}^{n-1} - c_0 \frac{\Delta t}{\Delta x} \left( u_{j+1}^n - u_{j-1}^n \right), \] (6.51)
where \( n = 0(1)N \) and \( j = 2(1)J \). To do our calculations we start at \( j = 2 \) to find the value at time level \( n = 1 \), that is, let \( n = 0 \) and \( j = 2 \) in (6.50) and (6.51), viz.,

\[
\begin{align*}
  u^1_2 &= u^{-1}_2 - \frac{\Delta t}{\Delta x} (\phi^0_3 - \phi^0_1), \\
  \phi^1_2 &= \phi^{-1}_2 - c_o^2 \frac{\Delta t}{\Delta x} (u^0_3 - u^0_1).
\end{align*}
\]

This is all fine since all the values on the right-hand sides are known using the initial conditions. The problem arises for the next time level \( n = 2 \) which reads

\[
\begin{align*}
  u^2_2 &= u^0_2 - \frac{\Delta t}{\Delta x} (\phi^1_3 - \phi^1_1), \\
  \phi^2_2 &= \phi^0_2 - c_o^2 \frac{\Delta t}{\Delta x} (u^1_3 - u^1_1).
\end{align*}
\]

The term that poses a problem is \( \phi^1_1 \) in (6.54) which is unknown since we are simply unable to calculate it using our equations. Please observe that we have a similar problem at the other boundary \( x = L \). Thus if we try to remedy this problem by using option (i) above, that is, specify \( \phi \) at \( x = 0 \) as well, the problem aggravates at \( x = L \). However tempting we are not allowed to specify more than two conditions in space total. If we continue and specifies \( \phi \) at the boundaries in addition to \( u \), we run into the problem of over-specifying the system, a dangerous path. We do get numbers out of the computer, they may even look reasonable, but they are wrong. This author strongly advocates against exploring such an avenue.

To avoid the problem Mesinger and Arakawa (1976) suggested to use what is referred to as staggered grids. Instead of calculating the two variables \( u \) and \( \phi \) at the same points, we simply stagger one of them with respect to the other, say one half grid length apart along the \( x \)-axis. Thus we calculate \( u \) at \( x_{j+\frac{1}{2}} \)-points and \( \phi \) at \( x_j \)-points. The finite difference approximation to scheme (6.25) and (6.26) then becomes

\[
\begin{align*}
  u^{n+1}_{j+\frac{1}{2}} &= u^{n-1}_{j+\frac{1}{2}} - \frac{2\Delta t}{\Delta x} (\phi^n_{j+1} - \phi^n_j), \\
  \phi^{n+1}_j &= \phi^{n-1}_j - c_o^2 \frac{2\Delta t}{\Delta x} (u^n_{j+\frac{1}{2}} - u^n_{j-\frac{1}{2}}).
\end{align*}
\]

Note the appearances of the factor \( 2\Delta t \). It appears since the distance between two adjacent points in the finite difference approximation of \( \partial_x \) in the staggered formulation is \( \Delta x \) rather than \( 2\Delta x \).

To avoid the cumbersome use of the \( j+\frac{1}{2} \) notation we start by visualizing the points associated with \( \phi_j \) as being located in one grid and the points associated with \( u_{j+\frac{1}{2}} \) to be located in another grid of the same size. We are then in a position where we can think of \( u_{j+\frac{1}{2}} \) belonging to the grid that is staggered one half grid length with respect to the \( \phi \)-grid as illustrated in Figure 6.1 on page 81. Further illustrated in Figure 6.1 is that we may think of \( \phi_j \) and \( u_{j+\frac{1}{2}} \) as being located in the same cells. As illustrated we then number the cells so that \( \phi_j \) and \( u_{j+\frac{1}{2}} \) are in the same cell numbered \( j \). Using the cell numbering structure rather than the point structure, we may than rewrite (6.56) and (6.57). Recall that (6.56) is calculated at \( u \)-points and (6.57) at \( \phi \)-points. Thus
we get

\[ u_j^{n+1} = u_j^{n-1} - 2 \frac{\Delta t}{\Delta x} (\phi_{j+1}^{n} - \phi_j^{n}) , \quad (6.58) \]

\[ \phi_j^{n+1} = \phi_j^{n-1} - c \frac{2 \Delta t}{\Delta x} (u_j^{n} - u_{j-1}^{n}) . \quad (6.59) \]

Thus we avoid the cumbersome one-half notation.

It remains to answer the question whether the staggering has an impact on the stability. To investigate this we note first note that care has to be exercised when constructing the discrete Fourier components. Since they are associated with the point numbering we have to take into account that the \( u \) and \( \phi \) points are one half grid-length apart. Thus the discrete Fourier components are

\[ \phi_j^n = \Phi_n e^{i \alpha j \Delta x} \quad \text{and} \quad u_j^n = U_n e^{i \alpha (j + \frac{1}{2}) \Delta x} \quad (6.60) \]

respectively. Substituting these expressions into (6.58) and (6.59), or (6.56) and (6.57), we get

\[ U_{n+1} - U_{n-1} = -4 i \frac{\Delta t}{\Delta x} \Phi_n \sin \left( \frac{\alpha \Delta x}{2} \right) , \quad (6.61) \]

\[ \Phi_{n+1} - \Phi_{n-1} = -4 i c \frac{2 \Delta t}{\Delta x} U_n \sin \left( \frac{\alpha \Delta x}{2} \right) . \quad (6.62) \]

Eliminating \( U_n \) we get

\[ \Phi_{n+2} - 2 \Phi_n + \Phi_{n-2} = -16 \sin^2 \left( \frac{\alpha \Delta x}{2} \right) \Phi_n . \quad (6.63) \]

Moreover, defining a growth factor by

\[ G \equiv \frac{\Phi_{n+2}}{\Phi_n} \quad (6.64) \]

and using the formula \( \sin^2 \psi = 2(1 - \cos \psi) \) we get

\[ G^2 - 2 \lambda G + 1 = 0 , \quad (6.65) \]
6.4. LINEAR, ROTATING CHAPTER 6. THE SHALLOW WATER PROBLEM

where

\[ \lambda = 1 - 4c_0^2 \left( \frac{\Delta t}{\Delta x} \right)^2 (1 - \cos \alpha \Delta x). \] (6.66)

The growth factor therefore has two complex conjugate solutions given by

\[ G_{1,2} = \lambda \pm i \sqrt{1 - \lambda^2}. \] (6.67)

Thus as long as \( \lambda^2 \leq 1 \) we get that \( |G_{1,2}| = 1 \). The staggered scheme is therefore neutrally stable as expected. However, this is only true as long as \( \lambda^2 \leq 1 \), which requires that

\[ |c_0| \frac{\Delta t}{\Delta x} \leq \frac{1}{2}, \quad \text{or} \quad \Delta t \leq \frac{\Delta x}{2|c_0|}. \] (6.68)

We observe that (6.68) is a more stringent CFL condition compared to the CFL condition (6.49) associated with the non-staggered grid. This is no surprise. When we staggered the grids we effectively decreased the distance between two adjacent points to one-half the original grid length. Thus the distance between two adjacent points in the staggered grid, say \( \Delta x_{stagg} \), is simply \( \Delta x_{stagg} = \Delta x/2 \). Using \( \Delta x_{stagg} \) instead of \( \Delta x \) the CFL condition becomes the expected

\[ \Delta t \leq \frac{\Delta x_{stagg}}{|c_0|}. \] (6.69)

6.4 Linear, rotating shallow water equations

We now throw in the Effect of the Earth’s rotation by retaining the Coriolis terms. We first study their linear, one-dimensional version, that is, (6.22) - (6.24).

As we did for the non-rotating case (cf. Section 6.2) it is worthwhile to analyze the various motions supported by the subset (6.22) - (6.24). To this end we again use the fact that the solution consists of a sum of Fourier components and analyze the solution for one Fourier component. Thus we assume a solution,

\[ x = x_0 e^{i\alpha(x-ct)}, \] (6.70)

where \( \alpha \) as earlier is the wavenumber in the \( x \) direction and \( x \) denotes a vector consisting of the now three dependent variables, that is,

\[ x = \begin{bmatrix} u \\ v \\ \phi \end{bmatrix} \quad \text{and} \quad x_0 = \begin{bmatrix} u_0 \\ v_0 \\ \phi_0 \end{bmatrix}, \] (6.71)

where \( x_0 \) is the amplitude.

Inserting (9.86) into the linearized equations (6.22) - (6.24) on page 77 we get

\[ i\alpha (\bar{u} - c) u - f v + i\alpha \phi = 0, \] (6.72)

\[ f u + i\alpha (\bar{u} - c) v = 0, \] (6.73)

\[ i\alpha \bar{\phi} u + i\alpha (\bar{u} - c) \phi = 0. \] (6.74)
which in turn may be formulated as the homogeneous linear equation,
\[ \mathbf{A} \cdot \mathbf{x} = 0, \quad (6.75) \]
where the tensor (or square matrix) \( \mathbf{A} \) is
\[
\mathbf{A} = \begin{bmatrix}
  i\alpha (\bar{u} - c) & -f & i\alpha \\
  f & i\alpha (\bar{u} - c) & 0 \\
i\alpha \bar{\phi} & 0 & i\alpha (\bar{u} - c)
\end{bmatrix}.
\quad (6.76)
\]
For non-trivial solutions to exist, the determinant of the tensor \( \mathbf{A} \) must be zero. Thus we get
\[ i\alpha (\bar{u} - c) \{-\alpha^2 (\bar{u} - c)^2 + f^2 + \alpha^2 \bar{\phi}\} = 0. \quad (6.77) \]
This equation has three solutions for the phase speed \( c \), namely
\[ c_1 = \bar{u}, \quad (6.78) \]
\[ c_{2,3} = \bar{u} \pm \sqrt{c_0^2 + \left(\frac{f}{\alpha}\right)^2}, \quad (6.79) \]
where
\[ c_0 = \sqrt{\bar{\phi}} = \sqrt{gH}. \quad (6.80) \]
The first solution is simply the geostrophic balance in (6.14) and (6.15). We easily derive this interpretation by substituting \( c_1 = \bar{u} \) from (6.78) into (6.72). The latter equation then becomes
\[ -fv + i\alpha \phi = 0 \quad \text{or} \quad v = \frac{1}{f}i\alpha \phi. \quad (6.81) \]
Using the Fourier solution backwards we thus recover the geostrophic balance (6.15), that is,
\[ v = \frac{1}{f}\partial_x \phi. \quad (6.82) \]
The two other solutions represented by \( \pm \sqrt{c_0^2 + \left(\frac{4}{\alpha}\right)^2} \) are combined inertia and gravity waves, so called inertia-gravity waves. The inertia part is associated with oscillating frequencies \( \omega = f \), that is, frequencies proportional to the inertia frequency or inertial oscillation. The gravity waves are associated with wave speeds \( c = \sqrt{\bar{\phi}} = c_0 \).

To solve (6.22) - (6.24) numerically using a finite difference method we replace the derivatives by finite difference approximations. To this end we first note that commonly the inertia-gravity mode has a much higher frequency than the Rossby mode \( \bar{\phi}^2 \gg \bar{u}^2 \), and that these modes becomes unstable first. For simplicity we therefore put \( \bar{u} = 0, \bar{v} \), that is, we are back to the problem with a basic state at rest. Since the problem is hyperbolic we apply the centered in time, centered in space (CTCS) leapfrog scheme on a non-staggered grid. We then get
\[ u_j^{n+1} - u_j^{n-1} = 2f \Delta t v_j^n - \frac{\Delta t}{\Delta x} (\phi_{j+1}^n - \phi_{j-1}^n), \quad (6.83) \]
\[ v_j^{n+1} - v_j^{n-1} = -2f \Delta t u_j^n, \quad (6.84) \]
\[ \phi_j^{n+1} - \phi_j^{n-1} = -c_0^2 \frac{\Delta t}{\Delta x} (u_{j+1}^n - u_{j-1}^n). \quad (6.85) \]
Replacing the variables by their discrete Fourier components

\[ u_n^j = U_n e^{i \alpha j \Delta x}, \quad v_n^j = V_n e^{i \alpha j \Delta x}, \quad \phi_n^j = \Phi_n e^{i \alpha j \Delta x}, \]

we get

\[ U_{n+1} - U_{n-1} = 2 f \Delta t V_n - 2 i \Phi_n \frac{\Delta t}{\Delta x} \sin \alpha \Delta x, \quad (6.87) \]
\[ V_{n+1} - V_{n-1} = -2 f \Delta t U_n, \quad (6.88) \]
\[ \Phi_{n+1} - \Phi_{n-1} = -2 i c_0^2 U_n \frac{\Delta t}{\Delta x} \sin \alpha \Delta x. \quad (6.89) \]

Eliminating \( V_n \) and \( \Phi_n \) we get\(^4\)

\[ U_{n+2} - 2 \lambda U_n + U_{n-2} = 0, \quad (6.90) \]

where

\[ \lambda = 1 - 2 (C^2 \sin^2 \alpha \Delta x + f^2 \Delta t^2) \quad \text{where} \quad C = c_0 \frac{\Delta t}{\Delta x} \quad (6.91) \]

Defining the growth factor as before we get two complex conjugate solution \( G_{1,2} = \lambda \pm i \sqrt{1 - \lambda^2} \) provided the radical is positive definite. As expected \( |G_{1,2}| = 1 \) and hence that the CTCS scheme is neutrally stable. The impact of throwing in the Coriolis effect is inherent in the expression for \( \lambda \). To ensure that the growth factor is complex we must require \( \lambda^2 \leq 1 \). This is met if

\[ c_0^2 \left( \frac{\Delta t}{\Delta x} \right)^2 \sin^2 \alpha \Delta x + f^2 \Delta t^2 \leq 1. \quad (6.92) \]

Thus the CFL criterion for stability becomes,

\[ \Delta t \leq \frac{\Delta x}{\sqrt{c_0^2 + f^2 \Delta x^2}}. \quad (6.93) \]

Since the first term in the radical dominates it follows that in practice

\[ \Delta t \leq \frac{\Delta x}{c_0} \quad \text{or} \quad C \leq 1 \quad (6.94) \]

where, as so many times earlier, \( C = c_0 \Delta t / \Delta x \) is the Courant number.

In an atmospheric model the largest equivalent depth is approximately 10 km giving a speed of the inertia-gravity waves of the order of 300m/s. This is considerably more than the wind speed and sets strong limitations to how long time steps we can take. In an ocean model the situation is the same. Although the oceanic equilibrium depth is one order of magnitude less, about 1km, the wave speed is still about 100m/s considerably larger than a typical ocean current.

---

\(^4\)This most efficiently done by first raising the \( n \) by one in (6.87) giving an expression involving \( U_{n+2}, U_n, V_{n+1} \) and \( \Phi_{n+1} \). Next decreasing \( n \) by one gives an expression containing \( U_n, U_{n-2}, V_{n-1} \) and \( \Phi_{n-1} \). Subtracting and using (6.88) and (6.89) then results in (6.90).
CHAPTER 6. THE SHALLOW WATER PROBLEM 6.5. NON-ROTATING DYNAMICS

speed. Thus both in the ocean and the atmosphere the CFL condition limits the time steps to minutes and sometimes seconds\(^5\). In contrast to the atmosphere the inertia-gravity waves contain the tidal motion and the storm surge signal. Thus in ocean models we are restricted to such limitations on the time step if we want to simulate these important oceanic features explicitly.

6.5 Non-rotating, non-linear shallow water equations

The next subset we consider is one in which we keep the non-linear terms in the system (6.1) - (6.3), but throw out the Coriolis terms and all terms differentiated with respect to \(y\). We then get

\[
\begin{align*}
\partial_t u + u \partial_x u + \partial_x \phi &= 0, \\
\partial_t \phi + u \partial_x \phi + \phi \partial_x u &= 0.
\end{align*}
\]

We take note that again \(v = 0\) and that we are left with a system of two equation for the two unknowns \(u, \phi\).

Rewriting (6.95) and (6.96) in terms of the volume flux \(U = hu\) and noting that \(\phi = gh\) we get

\[
\begin{align*}
\partial_t U + \partial_x \left( \frac{U^2}{h} \right) + \frac{1}{2} g \partial_x h^2 &= 0, \\
\partial_t h + \partial_x U &= 0.
\end{align*}
\]

The system is still hyperbolic so it is natural to employ a CTC S (leapfrog) scheme. Hence

\[
\begin{align*}
U_j^{n+1} &= U_j^{n-1} - \frac{\Delta t}{\Delta x} \left\{ \left[ \frac{U^2}{h} \right]_{j+1}^{n} - \left[ \frac{U^2}{h} \right]_{j}^{n} + \frac{1}{2} g \left[ h^2 \right]_{j+1}^{n} - \left[ h^2 \right]_{j-1}^{n} \right\}, \\
h_j^{n+1} &= h_j^{n-1} - \frac{\Delta t}{\Delta x} \left( U_{j+1}^{n} - U_{j-1}^{n} \right),
\end{align*}
\]

The question then arises if the scheme is stable. Since the equations are non-linear the analysis is not as straightforward as for a linear system. In fact throwing in the non-linear terms adds to the complexity of the possibilities for an unstable solution. The reason for this is that the non-linear terms are able to redistribute energy among the different wavelengths present in the problem, something which is impossible in a linear system. In fact as time evolves the non-linear terms acts to cascade the energy progressively towards smaller and smaller wavelengths. The energy thus ends up at the shortest wavelength that we resolve in our grid (2\(\Delta x\)) and eventually causes any scheme that works well for a linear system to blow up. This is called non-linear instability and is treated in more detail in Section 9.2 on page 113.

\(^5\)For an ocean of equilibrium depth \(H = 4 \cdot 10^3\text{m}, g = 10\text{ms}^{-2}\) and non-eddy resolving grid size \(\Delta x = 20\text{km}\) follows from (6.94) that \(\Delta t < 141\text{s}\) or slightly more than two minutes.
6.6 The semi-implicit method

From the analysis above we notice that by introducing a pressure force (in addition to advection) the CFL criterion becomes much more stringent (shorter time-step). It is therefore tempting to treat terms responsible for this behavior implicitly while we treat other terms explicitly. Such a method is commonly referred to as a semi-implicit method.

For clarity we start with a one dimensional shallow water problem, that is,

\[
\begin{align*}
\partial_t u &= A_u - \partial_x \phi, \\
\partial_t v &= A_v, \\
\partial_t \phi &= A_\phi - \Phi \partial_x u,
\end{align*}
\]

where \(A_u, A_v\) and \(A_\phi\) include the non-linear as well as the Coriolis terms. We learned previously that the terms responsible for this behavior was the pressure terms. From Section 5.2 we learned that treating any term implicitly avoid this restriction on the time step. It is therefore tempting to treat the pressure terms, that is, the \(\partial_x \phi\) and \(\Phi \partial_x u\) terms implicitly while integrating the remaining terms explicitly. The finite difference approximation form of the equations above then becomes,

\[
\begin{align*}
\frac{u^{n+1} - u^{n-1}}{2\Delta t} &= [A_u]^n - [\partial_x \phi]^{n+1}, \\
\frac{v^{n+1} - v^{n-1}}{2\Delta t} &= [A_v]^n, \\
\frac{\phi^{n+1} - \phi^{n-1}}{2\Delta t} &= [A_\phi]^n - \Phi [\partial_x u]^{n+1},
\end{align*}
\]

To proceed, the first two equations are solved with respect to \(u_j^{n+1}\) and \(v_j^{n+1}\) respectively giving,

\[
\begin{align*}
u_j^{n+1} &= u_j^{n-1} + 2\Delta t[A_u]^n - 2\Delta t\partial_x \phi^{n+1} \\
v_j^{n+1} &= v_j^{n-1} + 2\Delta t[A_v]^n
\end{align*}
\]

derivation with respect to \(x\) and insertion into the equation for \(\phi\) gives a Helmholtz equation,

\[
\Phi \Delta t^2[\partial_x^2 \phi]^{n+1} - \phi^{n+1} = B
\]

where \(B\) contains known quantities at time levels \(n, n-1, \ldots\)

With proper boundary conditions (\(\phi\) or its normal derivative at lateral boundaries), these equations may easily be solved by standard numerical methods called elliptic solvers (cf. Section 4.8). Having obtained \(\phi^{n+1}\), we easily find \(u^{n+1}\) and \(v^{n+1}\). This method is widely used in atmospheric models as we do not have to take the gravity mode speed \(c_0 = \sqrt{\Phi} = \sqrt{gH}\) into account when estimating an upper bound for the time step. Thus we avoid the restrictive CFL condition and thus makes it is possible to use longer time-steps. Note that we cannot do this in the ocean because then the inertia-gravity waves contains the signal as alluded to above.

86
6.7 The Semi-Lagrangian method

The semi-lagrangian method is constructed in order to take even longer time steps for advection than the Leapfrog method permits. In addition it is very accurate. For each time step one calculates where parcels arriving at a grid point came from. These departure points will normally differ from the standard grid-points and the values of the prognostic variables at time level $n$ must be estimated by interpolation from the surrounding grid point values. The new value at the grid points at time level $n + 1$ due to advection are then simply the value at the departure point at time level $n$. The one dimensional rendition of this method is the method of characteristics as outlined in Section 5.5.

By using an interpolation method with sufficient accuracy, for instance cubic interpolation, the method is more accurate than using second order differences and leapfrog time-integration. In addition the stability is independent of the length of the time step as long as the departure-points values are calculated by interpolation (not extrapolation). However, the accuracy strongly depends on how accurate one can estimate the departure points and with very long time steps the estimates will be inaccurate. Another drawback is that the method requires additional computations and this overhead may be as costly as taking a shorter time step with a simpler method. The method is widely used in atmospheric models; often in combination with the semi-implicit method.
Chapter 7

GENERAL VERTICAL COORDINATES

Most modern models employed in the meteorological and oceanographic community replace the normal geopotential vertical coordinate with a new coordinate. For instance in the atmosphere is it quite common to formulate the governing equations using the pressure as the vertical coordinate. In the ocean it has become quite popular to use a terrain-following coordinate or the potential density as the vertical coordinate. The former are also referred to as \( \sigma \)-coordinate models, while the latter are most often referred to as isopycnal models. Also models that use a hybrid vertical coordinate, that is, combine for instance an isopycnal coordinate with a terrain-following coordinate is now in development.

To obtain the governing equations in the new coordinate system it is common to transform them using their formulation in ordinary Cartesian or geopotential coordinates. Since there are more than one vertical coordinate in use we will first transform the equations using a general vertical coordinate, that is, a coordinate system that replaces the geopotential height coordinate with a general vertical coordinate.

7.1 Transformation to a general vertical coordinate

In general we transform from one coordinate system of independent variables, say \((x, y, z, t)\), to another system, say \((x', y', s, t')\), by specifying how the independent variables in the transformed system depend on the independent variables of the original system. Here we will only replace the vertical height coordinate \(z\). Accordingly we define the transformation simply by

\[
x' = x, \quad y' = y, \quad s = s(x, y, z, t), \quad \text{and} \quad t' = t,
\]

where we have only replaced the normal geopotential height coordinate \(z\) with a general vertical coordinate \(s\) while the horizontal coordinates are left unchanged in three-dimensional space. To ensure that the transformation is unique we must require that \(s\) is a monotone function of height \(z\). Mathematically this means that the gradient of \(s\) with respect to \(z\) does not change sign within a fluid column, or

\[
\partial_z s \geq 0, \quad \text{and} \quad \partial_z s \neq 0.
\]
This is also a necessary condition to ensure that the inverse transformation 
\[ z = z(x', y', s, t') \]
exists as well.

Let \( \xi' \) denote any of the three independent variables \( x', y', \) and \( t' \) in the new coordinate system, and similarly let \( \xi \) denote any of the three independent variables \( x, y, \) and \( t \) in the original system. Then the transformation (7.1) gives

\[
\frac{\partial z}{\partial \xi'} = 0, \quad \frac{\partial x'}{\partial \xi} = \frac{\partial y'}{\partial \xi} = 0, \quad \text{and} \quad \frac{\partial t'}{\partial \xi} = \frac{\partial t}{\partial \xi} = 0, \quad (7.3)
\]
while

\[
\frac{\partial x'}{\partial \xi} = \frac{\partial y'}{\partial \xi} = \frac{\partial t}{\partial \xi} = 1. \quad (7.4)
\]

Similarly follows

\[
\frac{\partial \xi}{\partial \xi'} = 0, \quad \frac{\partial \xi'}{\partial \xi} = \frac{\partial \xi'}{\partial \xi} = 0, \quad \text{and} \quad \frac{\partial \xi'}{\partial \xi} = \frac{\partial \xi'}{\partial \xi} = 0, \quad (7.5)
\]
while

\[
\frac{\partial \xi'}{\partial \xi} = \frac{\partial \xi'}{\partial \xi} = \frac{\partial \xi'}{\partial \xi} = 1. \quad (7.6)
\]

We emphasize that \( s \) is monotonic with respect to \( z \), which implies that \( \frac{\partial z}{\partial s} \neq 0 \) and \( \frac{\partial s}{\partial z} \neq 0 \). We also observe that if we transform \( z \) to \( z \), that is, let \( s = z \) then \( \frac{\partial z}{\partial s} = \frac{\partial s}{\partial z} = 1 \).

Let \( \psi = \psi(x, y, z, t) = \psi(x', y', s, t') \) denote any scalar. Then the first property of the transformation is

\[
\frac{\partial \psi}{\partial \xi} = \frac{\partial z}{\partial s} \frac{\partial \psi}{\partial s}. \quad (7.7)
\]

If we take the derivative of \( \psi \) with respect to one of the independent variables in the coordinate system we transform to, say \( t' \), then we get

\[
\frac{\partial \psi}{\partial t'} = \frac{\partial \psi}{\partial t} - \frac{\partial z}{\partial s} \frac{\partial \psi}{\partial s} \frac{\partial t}{\partial t'}, \quad (7.8)
\]
where the last equal sign follows by utilizing (7.3) - (7.7). If we solve (7.8) with respect to \( \partial_t \psi \) we further get

\[
\partial_t \psi = \partial_{t'} \psi - \frac{\partial z}{\partial s} \partial_{s'} \psi \partial_t z. \quad (7.9)
\]

Similarly follows that

\[
\partial_x \psi = \partial_{x'} \psi - \frac{\partial z}{\partial s} \partial_{s'} \psi \partial_x z, \quad \text{and} \quad \partial_y \psi = \partial_{y'} \psi - \frac{\partial z}{\partial s} \partial_{s'} \psi \partial_y z. \quad (7.10)
\]

Let us define the horizontal gradient of \( \psi \) in the new coordinate system by

\[
\nabla_s \psi = i \partial_{x'} \psi + j \partial_{y'} \psi. \quad (7.11)
\]

Then making use of (7.9) and (7.10) we obtain

\[
\nabla_H \psi = \nabla_s \psi - \frac{\partial z}{\partial s} \partial_{s'} \psi \nabla_s z. \quad (7.12)
\]

Furthermore we find that the horizontal divergence of any vector, say \( \mathbf{a} \), transforms as

\[
\nabla_H \cdot \mathbf{a} = \nabla_s \cdot \mathbf{a} + \frac{\partial z}{\partial s} \partial_s \mathbf{a} \cdot \nabla_s z. \quad (7.13)
\]
We note that all vectors project onto the horizontal geopotential surface. This is also true for the gradient (7.12). Thus the metric term associated with the vertical gradient of the surface $s$ in the geopotential coordinate system is eliminated.

We note that since the individual derivative\(^1\) is independent of coordinate transformation we get

\[
\frac{D\psi}{dt} = \partial_t \psi + \mathbf{u} \cdot \nabla_H \psi + w \partial_z \psi \quad (7.14)
\]

\[
= \partial_t' \psi + \mathbf{u} \cdot \nabla_s \psi + \dot{s} \partial_s \psi, \quad (7.15)
\]

where

\[
\dot{s} = \frac{Ds}{dt} \quad (7.16)
\]

is the speed of the surface $s$ in the direction of the three-dimensional velocity. Note that (7.14) is the derivative expressed in the geopotential coordinate system while (7.15) is the material derivative expressed in the new general vertical coordinate system. We now make use of (7.9) - (7.12) to replace the appropriate terms in (7.14). Then we get

\[
\frac{D\psi}{dt} = \partial_t' \psi + \mathbf{u} \cdot \nabla_s \psi + (w - \partial_t' z - \mathbf{u} \cdot \nabla_s z) \partial_z s \partial_s \psi. \quad (7.17)
\]

Furthermore, by use of (7.15) we get

\[
\dot{s} = (w - \partial_t' z - \mathbf{u} \cdot \nabla_s z) \partial_z s \equiv \omega \partial_z s, \quad (7.18)
\]

where the identity in (7.18) defines the velocity $\omega$ by

\[
\omega = w - (\partial_t' z + \mathbf{u} \cdot \nabla_s z) = w - (\partial_t s + \mathbf{u} \cdot \nabla_H s) \partial_s z. \quad (7.19)
\]

First we observe that if $s = z$ then $\omega = w$ as expected, and the velocity $\omega$ is interpreted as the speed by which the surface $s$ travels through the fixed $z$ levels of the geopotential coordinate system. Next we observe that if $s$ is a material surface the kinematic boundary condition is $w = (\partial_t s + \mathbf{u} \cdot \nabla_H s)$. Under these circumstance it follows that $\omega = 0$. This is again to be expected since a material surface is a surface that consists of the same fluid particles for all times. If the $s$ surface is not a material surface then $\omega \neq 0$ and is associated with the speed of the fluid particles through the surface $s$. The difference between $\omega$ and $w$ is hence associated with the speed of the surface $s$ in the geopotential coordinate system. Thus $\omega$ is interpreted as that part of the vertical movement of particles that remains when moving with the surface $s$.

### 7.2 Transformation of the governing equations

To give insight into how the transformation is applied, we apply it to a non-Boussinesq, hydrostatic fluid.

---

\(^1\)Also by many authors referred to as the material derivative
The hydrostatic equation

We start by transforming the hydrostatic equation

\[ \partial_z p + \rho g = 0. \]  

(7.20)

Using the transformation formulas of the previous section we get

\[ \partial_s p + \rho g \partial_s z = 0. \]  

(7.21)

We may use this equation to determine the metric factors \( \partial_s z \) and \( \partial_z s \) as follows

\[ \partial_s z = -\frac{\partial_s p}{\rho g}, \quad \text{and} \quad \partial_z s = -\frac{\rho g}{\partial_s p}. \]  

(7.22)

Mass conservation

Next, we transform the continuity equation

\[ \partial_t \rho + \nabla \cdot (\mathbf{v} \rho) = 0. \]  

(7.23)

We first rewrite this equation to yield

\[ \frac{1}{\rho} \frac{D\rho}{dt} + \nabla \cdot \mathbf{u} + \partial_z w = 0. \]  

(7.24)

We then make use of the transformation formulas to obtain

\[ \frac{1}{\rho} \frac{D\rho}{dt} + \nabla \cdot \mathbf{u} + \partial_z w = -\rho (\partial_v \alpha + \mathbf{u} \cdot \nabla_s \alpha + \dot{s} \partial_s \alpha) + \partial_s [\partial_v (\partial_s z) + \nabla_s \cdot (\mathbf{u} \partial_s z) + \partial_s (\dot{s} \partial_s z)], \]  

(7.25)

where \( \alpha = 1/\rho \). To arrive at this result we have also solved (7.19) with respect to \( w \) to replace \( \partial_s w \). We may further develop (7.25) by making use of (7.22) to replace the metric term \( \partial_s z \). Thus we get

\[ \frac{1}{\rho} \frac{D\rho}{dt} + \nabla \cdot \mathbf{u} + \partial_z w = (\partial_s p)^{-1} [\partial_v (\partial_s p) + \nabla_s \cdot (\mathbf{u} \partial_s p) + \partial_s (\dot{s} \partial_s p)], \]  

(7.26)

and hence the transformed continuity equation reads

\[ \partial_v (\partial_s p) + \nabla_s \cdot (\mathbf{u} \partial_s p) + \partial_s (\dot{s} \partial_s p) = 0. \]  

(7.27)

Energy equation

If we apply a similar procedure to the tracer equation (1.12) we get

\[ \partial_v C + \mathbf{u} \cdot \nabla_s C + \dot{s} \partial_s C = \mathcal{F}_C + S_C \]  

(7.28)

where the right-hand side represents the transformed fluxes and source terms.
The momentum equation

We finally transform the horizontal component of the momentum equation for a non-Boussinesq, hydrostatic fluid (1.11) by first rewriting it to read

$$\frac{Du}{dt} + f k \times u = -\alpha \nabla_H p + \alpha \partial_z \tau + \nabla \cdot \mathbf{F}_M^H$$

(7.29)

where \( \tau \) is the vertical mixing or flux vector, sometimes referred to as the vertical shear stress. To transform this equation is a bit more complicated so we treat it term by term.

We first consider the pressure term, which is special. For a non-Boussinesq fluid we get

$$\alpha \nabla_H p = \alpha \nabla_s p + g \nabla_s z = \nabla_s M - p \nabla_s \alpha$$

(7.30)

is the Montgomery potential (or stream function). In the case \( s = \rho \) the last term in (7.31) vanishes since then \( \nabla_s \alpha = 0 \). Under these circumstances the Montgomery potential becomes a true potential and is a streamfunction for the geostrophic velocity. For any other choice of \( s \), however, the last term in (7.31) must be retained. We finally note that the Montgomery potential appears because all vectors are projected onto the horizontal surface (with respect to gravity), even though all gradients are evaluated in the transformed \( x', y', s', t' \) system.

Next we consider the vertical shear stress term. In this we apply (7.7) and (7.22) to get

$$\alpha \tau = \alpha \partial_z \partial_s \tau = -g \frac{\partial \tau}{\partial p} = \partial_p \tau.$$  

(7.32)

Recalling that

$$\frac{Du}{dt} = \partial_t u + u \cdot \nabla_s u + s \partial_s u$$

(7.33)

and that

$$u \cdot \nabla_s u = \nabla_s \left( \frac{1}{2} u^2 \right) + \zeta k \times u$$

(7.34)

where \( \zeta = k \cdot \nabla_s \times u \) is the relative vorticity relative to the new coordinate system. Hence the momentum equation becomes

$$\partial_t u + \nabla_s \left( \frac{1}{2} u^2 \right) + (\zeta + f) k \times u + s \partial_s u = -\nabla_s M + p \nabla_s \alpha - g \partial_p \tau + \nabla \cdot \mathbf{F}_M^H.$$  

(7.35)

We may also write this equation in flux form. We then first recombine the second and third term on the left-hand side of (7.35) using (7.34). Next we multiply (7.35) by \( \partial_p p \) and then finally make use of the continuity equation in the form (7.27). We then get

$$\partial_t (u \partial_p p) + \nabla_s \cdot (uu \partial_p p) + f k \times u \partial_p p + s \partial_s (su \partial_p p)$$

$$= -\partial_p p (\nabla_s M + p \nabla_s \alpha) - g \partial_p \tau + \partial_s p \nabla_s \cdot \mathbf{F}_M^H.$$  

(7.36)

As alluded to earlier when treating the diffusion problem, we emphasize at this point that the mixing term or “diffusion” term is mostly added to prevent our numerical model from blowing up. Hence its exact transformation is of secondary importance.
7.3 Terrain-following coordinates

As an example we apply these transformations to transform the mass conservation equation to the so called \( \sigma \)-coordinate models. This particular coordinate system is defined by

\[
s = \sigma = \frac{z - \eta}{D} \quad \text{or} \quad z = \sigma D + \eta,
\]

(7.37)

where \( D = H + \eta \) is the total depth, \( \eta \) being the deviation of the upper surface from its equilibrium position and \( H \) is the equilibrium depth of the fluid columns. The terrain following coordinate models are very popular in the oceanographic community, e.g., ROMS (Haidvogel et al., 2007), and various versions of POM (Blumberg and Mellor, 1987; Engedahl, 1995a). It is also applied in numerical weather predictions models such as the NOAA eta model.

First we note that the metric factor \( \partial_s z \) and \( \partial_s \sigma \) using (7.22) becomes

\[
\partial_s z = \partial_z \sigma = \frac{1}{D} \quad \text{and} \quad \partial_s \sigma = \partial_z z = D,
\]

(7.38)

which allows us to rewrite the hydrostatic equation to

\[
\partial_\sigma p = -\rho g D.
\]

(7.39)

Furthermore we need to know the speed \( \omega \) through the \( \sigma \) surfaces. Applying (7.37) we get

\[
\omega = w - \sigma \partial_t D - \partial_t \eta - \sigma \mathbf{u} \cdot \nabla_\sigma D - \mathbf{u} \cdot \nabla_\sigma \eta.
\]

(7.40)

Thus the mass conservation equation in the form (7.27) becomes

\[
\partial_t (\rho D) + \nabla_\sigma \cdot (\rho D \mathbf{u}) + \partial_\sigma (\dot{\sigma} \rho D) = 0,
\]

(7.41)

Separating the effect of the density we get

\[
\frac{D \rho}{dt} + \frac{\rho}{D} [\partial_t D + \nabla_\sigma \cdot (D \mathbf{u}) + \partial_\sigma (\dot{\sigma} D)] = 0,
\]

(7.42)

We observe using (7.18) that \( \dot{\sigma} = \omega \partial_z \sigma = \omega D^{-1} \). Furthermore we note that \( \partial_t D = \partial_t \eta \). Substitution of these expressions into (7.42), and invoking the Boussinesq approximation (1.15) or \( \frac{D \rho}{dt} = 0 \), the continuity equation for a Boussinesq fluid in terrain-following coordinates is

\[
\partial_t \eta + \nabla_\sigma \cdot (D \mathbf{u}) + \partial_\sigma \omega = 0.
\]

(7.43)

We note that the remaining equations may be derived from their general expressions in a similar fashion. For instance using (7.39) the momentum equation in the flux form (7.36) becomes

\[
\partial_t (D \mathbf{u}) + \nabla_\sigma \cdot (D \mathbf{u} \mathbf{u}) + f \mathbf{k} \times D \mathbf{u} + \partial_\sigma (\omega \mathbf{u})
\]

\[
= -D (\nabla_\sigma M + p \nabla_\sigma \alpha) - \frac{1}{\rho_0} \partial_\sigma \mathbf{\tau} + D \nabla_\sigma \cdot \mathbf{F}_M^H.
\]

(7.44)
Chapter 8

OPEN BOUNDARY CONDITIONS AND NESTING TECHNIQUES

As is well known, computers, however large, can only hold a finite number of numbers in their so called random access memory (RAM). Thus even the biggest computers are limited in their capacity. This is perhaps the main reason why numerical oceanography is less mature than numerical meteorology. Recall that at sub-polar latitudes the typical length scale of a synoptic low in the atmosphere is about 500 - 1000 km and the typical time scale a few days. In contrast the scales of a low in the ocean at the same latitude are about 10 - 50 km in length and a few weeks to months in time. These differences are associated with the difference in the Rossby radius of deformation that in the ocean is about two order of magnitudes smaller than in the atmosphere.

To illustrate this point let us consider a global model with a grid size of about 2 degrees (Figure 8.1 upper panel). A mesh size of 2 degrees, or about 200 km, entails that the grid size is about one fifth of the the atmospheric Rossby radius. This is a tolerable grid size for a numerical atmosphere model. If we, however, scale this to the Rossby radius of deformation in the ocean, the grid for the atmosphere model would look like the one displayed in the lower panel of Figure 8.1 which has a grid size of about 3-4 times the Rossby radius of deformation. No meteorologist in his right mind would consider it to be an adequate grid for a numerical weather prediction (NWP) model. To obtain a similar tolerable resolution in the ocean we have to employ grids of mesh sizes 2-4 km, or 1/200th of a degree. Thus the need for RAM is much higher for an ocean model than for an atmosphere model covering the same region. In addition comes the fact that with decreased mesh size the time step is much smaller as well in order to satisfy the CFL criterion. In practice it therefore takes a much larger computational effort to provide say a 24 hour “weather” forecast for the ocean for a given area on a given computer than to provide a similar weather prediction. To enable computers to provide numerical ocean weather forecasts as fast as today’s NWP models for the same area we therefore need much faster computers. To make things even worse recall that the time scale in the ocean is much longer than in the atmosphere. A weather prediction of say ten days corresponds to an ocean forecasts of at least one month.

Global weather predictions with more than adequate resolution to resolve the atmospheric weather systems are common today. Such forecasts are run by several national institutes as well
CHAPTER 8. OPEN BOUNDARY CONDITIONS

Figure 8.1: Upper panel shows the Earth’s surface covered by a 2 degree mesh. Lower panel shows a similar mesh of 30 degrees mesh size. The figure conveniently illustrates how a 2 degree mesh in the ocean would look like in the atmosphere scaled by the Rossby radius of deformation.
as at the European Centre for Medium-range Weather Forecasts (ECMWF). However, there are still local effects, notably processes associated with irregular topography, that is not yet resolved properly by the global model. Thus most national institutes providing public meteorological services run a limited area model which is “nested” into a global model. When nesting a finer mesh model (or inner model) into a coarser mesh model (or outer model), one need to transfer the model results of the outer model to the inner model via their common boundary. Thus one need a condition at these boundaries. Since fluid is allowed to pass freely through these boundaries they are commonly referred to as open boundaries. Nesting of a finer mesh model into a coarser model are also sometimes referred to as dynamical downscaling since the inner model provides a solution that is a dynamically consistent downscaling of the coarser solution to a scale that takes into account the finer scales, for instance due to a more realistic representation of the topography.

The situation for the ocean is somewhat different. First, global numerical ocean weather predictions (NOWP) are not yet feasible at all on today’s computers. Thus none of the global ocean models that exist today resolve the oceanic weather at higher latitudes. This is particularly true for the oceanic component of the coupled global climate models, or so called AOGCMs (Atmosphere - Ocean Global Climate Models). To provide synoptic NOWPs in limited areas we therefore have to make use of nesting techniques or dynamical downscaling for the time being. Such forecasts, updated daily, for Norwegian waters are for instance provided by the Norwegian Meteorological Institute to the public at the website: http://retro.met.no/kyst_og_hav/havvarsel.html.

As a consequence both ocean and atmosphere models have to deal with open boundaries. At these boundaries the governing equations are still valid, which makes them very different from natural boundaries where the governing equations are replaced by a natural boundary condition as for instance the condition of no normal flow through a solid, impermeable boundary. Nevertheless, since our computational domain ends at the open boundary we have to provide a boundary condition there. Such conditions are called open boundary conditions or OBCs for short. The mathematical challenge is to construct OBCs which ensure that a solution to the governing equations exists and is unique. From a physical point of view though we would like the solution to be as close to the “correct” solution as possible. The correct solution refers to the one we would have obtained if the model was global with only natural boundary conditions applied. It should be emphasized that this is a dilemma since the solution to the governing equations is determined not only by the equations themselves, but also by the boundary conditions as mentioned in Section 2.5. Thus when applying OBCs to determine our solution we are not ensured that the solution we obtain is the correct one. In addition, it is impossible in general to prove that the solution we obtain is unique. Only for very simplified, one-dimensional problems is this possible.

One of the first to derive a numerical solution of a NWP problem applying OBCs was Jules Charney, Ragnar Fjørtoft and John von Neumann in the late 1940s. Their attempt was published in the famous paper by Charney et al. (1950). Recall that just after the second world war the digital computers were in their infancy. They therefore employed a very simple atmospheric
model compared to today’s standards. In fact they attempted to solve the barotropic, quasi-
geostrophic equations formulated in terms of a potential vorticity equation for a computational
domain limited to a square located in the middle of the North Atlantic Ocean. Thus they had
to apply OBCs at all four sides of the domain. They opted for a set of OBCs which was to
specify the potential vorticity at the inflowing boundaries and to apply a radiation condition (see
Section 8.2 below) at outflowing boundaries. Later Platzman (1954) showed that their solution
was unstable when applying the OBC they had opted for. Thus the problem of specifying OBCs
that renders the solution unique and correct is not a new. Since these early papers on the problems
encountered in specifying OBC there has been a variety of papers treating the problem. For a
review see Chapman (1985); Røed and Cooper (1986, 1987); Palma and Matano (2000) and
most recently Blayo and Debreu (2006).

8.1 What is an open boundary?

To help in choosing an OBC that would render the solution stable, the following definition of an
open boundary was offered by Røed and Cooper (1986):

An open boundary is a computational boundary at which disturbances originating
in the interior of the computational domain are allowed to leave it without disturb-
ing deteriorating the interior solution.

Thus we demand that the conditions we impose at open boundaries satisfied certain requirements.
These requirements may later be used to construct criteria by which the success of our employed
OBC may be assessed. We emphasize that an open boundary is an artificial boundary between
what we call our computational domain (or our area of interest) and a domain exterior to our
computational domain.

One obvious requirement, which follows directly from the definition above, is that distur-
bances originating in the interior of our domain propagating toward the open boundary should
be allowed to pass through to the exterior without distorting or disturbing the interior solution.
Equally obvious is that disturbances originating in the exterior domain is free to enter our do-
main without distortions. The latter is sometimes hard to achieve since we do not always have
sufficient knowledge about the exterior solution.

To illustrate this let us consider a Kelvin wave created in the interior of our domain and
propagating towards the open boundary. The condition we impose at the open boundary should
then be able to let the wave pass through and not be reflected, that is, none of the energy con-
tained in the Kelvin wave should be allowed to be radiated back into the interior. Likewise a
Kelvin wave created in the exterior domain, for instance by a model covering a larger domain
that encompasses our computational domain, should be free to enter without being distorted or
damped.

Next we require that the chosen OBC leads to a stable solution (numerically). Moreover,
from a mathematical point of view we require that the OBC together with the governing equations
leads to a mathematical problem that is well posed or at least well-posed enough so that a solution
exists and is unique.
8.2 Radiation conditions

Many of the processes in the ocean and atmosphere are processes involving wave propagation in one way or another. The early attempts at developing OBCs therefore based their OBC formulations on simple wave equation. In its simplest form the wave equation reads,

\[ \partial_t \phi + c_\phi \partial_n \phi = 0 \]  

(8.1)

Here \( \phi \) represents the dependent variable, \( c_\phi \) is the component of the phase velocity normal the boundary associated with the variable \( \phi \), while \( \partial_n \) denotes the derivative normal to the open boundary. Imposing (8.1) as an OBC it becomes what is known as the radiation condition. When use is made of (8.1) as an OBC we fundamentally assume that the disturbances passing through the open boundary consists of waves. Note that the disturbances passing through the boundary may consist of several waves of different wavelengths, and hence that (8.1) is strictly speaking only valid for one Fourier component only.

One of the first obstacles when using (8.1) as an OBC is therefore that we do not know the phase velocity \( c_\phi \). From Section 5.6 we recall that \( c_\phi \) is the slope of the characteristics. Thus if the choice of \( c_\phi \) perfectly matches the slope of the characteristics then (8.1) is a perfect open boundary condition. However, it is only for very simple, physical problems, e.g., for a monochromatic wave problem, that we are able to determine the characteristics \textit{apriori}, and hence \( c_\phi \) is generally unknown.

We immediately recognize that (8.1) contains two special cases. The first case is for \( c_\phi = 0 \), while the second is the opposite, namely when the phase velocity \( c_\phi \to \infty \). In the former case we notice that the characteristics are straight vertical lines in \( x,t \) space, and that we may integrate (8.1) in time to give,

\[ \phi = \text{const.} \]  

(8.2)

Thus under these circumstances the dependent variable is known for all times at the OBC, and we recall from Section 2.5 that the OBC is a Dirichlet condition. Commonly this is referred to as a clamped condition since the dependent variable \( \phi \) does not change in time at the OB.

In the latter case when \( c_\phi \to \infty \) we notice that characteristics are horizontal straight lines in the \( x,t \) space. We notice from (8.1) that if \( \partial_t \phi \) should remain finite we must require that the gradient \( \partial_n \phi \) must be zero, and hence that

\[ \partial_n \phi = 0. \]  

(8.3)
Usually the condition (8.3) is referred to simply as a *gradient condition*. We recall from Section 2.5 that such a condition was referred to as a Neumann condition.

If the phase velocity is finite and differs from zero, then we have a true radiation condition. The problem is then reduced to determine the phase velocity $c_\phi$. If the solution is in the form of known waves, say a barotropic Kelvin wave\(^2\). Under these circumstances the phase velocity is known, and in the case of a Kelvin wave it is,

$$c_\phi = c_0 = \sqrt{gh}$$  \hspace{1cm} (8.4)

where $g$ is the gravitational acceleration, and $H$ is the equilibrium depth of a fluid column.

Let us consider a problem of a fluid contained in a channel of equilibrium depth $H$. Furthermore, let us consider a frictionless motion and let $h$ denote the total depth or layer thickness of a fluid column and $u$ the speed of the fluid column\(^3\). Moreover, let us consider that the motion is on a non-rotational Earth, and that the fluid has constant and uniform density. Then the governing equations may be written (cf. Section 6.2)

$$\frac{\partial u}{\partial t} = -gh\frac{\partial h}{\partial x}$$  \hspace{1cm} (8.5)

$$\frac{\partial h}{\partial t} = -H\frac{\partial u}{\partial x}$$  \hspace{1cm} (8.6)

The classic method to solve the above set is to first differentiate (8.5) with respect to $x$ and (8.6) with respect to $t$ and then add the results. The result is

$$\frac{\partial^2 h}{\partial t^2} - c_0^2 \frac{\partial^2 h}{\partial x^2} = 0,$$  \hspace{1cm} (8.7)

that is, a wave equation with a phase speed equal to $c_0$ as given in (8.4). The set (8.5) and (8.6) thus requires two boundary conditions in space. Let us assume that the channel has two open boundaries at $x = 0$ and $x = L$. The natural boundary condition at these two boundaries is then the radiation condition (8.1) with a phase speed of $\pm c_0$, respectively.

Recalling that the phase velocity is determined by the slope of the characteristics, we may also use the method of characteristics to find it. As we will show this also leads us to impose the radiation condition at the two open boundaries. We start by multiplying (8.6) by an as yet unknown function $\lambda$ and add the result to (8.5). We then get

$$\frac{\partial t}{\partial t} + \lambda H \frac{\partial_x u} + \lambda \left( \frac{\partial h}{\partial t} + \frac{g}{\lambda} \frac{\partial_x h} \right) = 0$$  \hspace{1cm} (8.8)

Next we define the operator $\frac{D^*}{dt}$ such that

$$\frac{D^*}{dt} = \frac{\partial t}{\partial t} + \frac{D^*}{dt} \frac{\partial_x u}$$  \hspace{1cm} (8.9)

that is, an individual derivative in the special direction $D^*x$ (cf. Section 5.6). We may now formulate an equation for the unknown function $\lambda$ by requiring that the characteristics satisfies the following equations

$$\frac{D^*}{dt} \frac{x}{dt} = \lambda H,$$  \hspace{1cm} (8.10)

\(^2\)A barotropic Kelvin wave is common phenomena in oceanography. It belongs to the class of planetary gravity waves. Kelvin waves are commonly filtered out in meteorology models.

\(^3\)Since we consider a frictionless motion we may safely assume that $u$ is independent of depth.
and
\[ \frac{D^* x}{dt} = \frac{g}{\lambda}. \]  
Under these circumstances we may write equation (8.8) as
\[ \frac{D^* u}{dt} + \lambda \frac{D^* h}{dt} = 0. \]  
Equating the right-hand sides of (8.10) and (8.11) and solving for the unknown function \( \lambda \) we get the two solutions
\[ \lambda_{1,2} = \pm \frac{c_0}{H}, \]  
where \( c_0 = \sqrt{gH} \). Hence (8.12) becomes
\[ \left( \frac{D^*}{dt} \right)_{1,2} \left( u \pm c_0 \frac{h}{H} \right) = 0 \]  
valid along the characteristics given by
\[ \left( \frac{D^* x}{dt} \right)_{1,2} = \pm c_0. \]  
We observe that (8.14) and (8.15) are formally two equations. By use of (8.9) and (8.15) we thus get
\[ \partial_t \left( u + c_0 \frac{h}{H} \right) + c_0 \partial_x \left( u + c_0 \frac{h}{H} \right) = 0, \]  
\[ \partial_t \left( u - c_0 \frac{h}{H} \right) - c_0 \partial_x \left( u - c_0 \frac{h}{H} \right) = 0. \]  
We recall from Section 5.5 that (8.16) and (8.17) (or eq. 8.14) are the compatibility equations, while (8.15) are the two characteristic equations. While (8.16) describes a wave propagating in the positive \( x \)-direction with phase velocity \( c_0 \), we observe that (8.17) describes a wave propagating in the opposite direction, but with the same phase velocity. In particular we notice that (8.14) expresses that the specific combinations of the dependent variables \( u \) and \( h \), namely \( u \pm c_0 \frac{h}{H} \), are conserved along the characteristics (8.15).

Let us assume that our problem is to solve (8.5) for \( 0 < x < L \) and that the two boundaries \( x = 0 \) and \( x = L \) are open. Let us in addition assume that a motion is generated in the interior of the domain, e.g., in the form of an initial deviation of the layer thickness \( h \) locally. The question then arises: what is the correct boundary condition to impose on the two open boundaries? We know from the two compatibility equations (8.14) that the information about the deviation will propagate along the two characteristics given by (8.15). Towards the right-hand boundary at \( x = L \) the information will propagate along \( \left( \frac{D^* x}{dt} \right)_1 = c_0 \), and towards the left-hand boundary \( x = 0 \) along \( \left( \frac{D^* x}{dt} \right)_2 = -c_0 \). To avoid reflection we must impose a condition that ensures that information cannot propagate back into our interior domain. Since information propagates along
the characteristics, we must ensure that no characteristics at \( x = 0, L \) slopes towards the interior. Consequently we require that

\[
\left( \frac{D^*x}{dt} \right)_2 = 0 \quad \text{at} \quad x = L \tag{8.18}
\]

and

\[
\left( \frac{D^*x}{dt} \right)_1 = 0 \quad \text{at} \quad x = 0. \tag{8.19}
\]

Substituting this into the leftmost expressions in respectively (8.16) and (8.17) we get

\[
\partial_t \left( u - c_0 \frac{h}{H} \right) = 0 \quad \text{at} \quad x = L \tag{8.20}
\]

and

\[
\partial_t \left( u + c_0 \frac{h}{H} \right) = 0 \quad \text{at} \quad x = 0. \tag{8.21}
\]

We now integrate (8.20) and (8.21) in time and get

\[
\begin{aligned}
\quad & u = c_0 \frac{h}{H} + \text{const.}, \quad \text{at} \quad x = L \tag{8.22} \\
\quad & u = -c_0 \frac{h}{H} + \text{const.} \quad \text{at} \quad x = 0. \tag{8.23}
\end{aligned}
\]

This is in fact the radiation condition. Indeed if we substitute the expression (8.6) for \( h \) into (8.22) we get (8.1) with \( \phi = u \) and \( c_\phi = c_0 \).

The advantage of using the method of characteristics to derive the non-reflective boundary condition is that it gives us insight into how to construct open boundary conditions in general. This was for instance exploited by Røed and Cooper (1987) to construct a weakly reflective boundary condition for a more general problem including the effect of Earth’s rotation based on earlier work by Hedstrøm (1979) (cf. Section 8.5).

### 8.3 Implementation of the radiation condition

We now consider the numerical implementation of the one-dimensional version of the radiation condition (8.1), and that the space variable is \( x \). In this we essentially follow the implementation given in Røed and Cooper (1987). Recall that the radiation condition then reads

\[
\partial_t \phi + c_\phi \partial_x \phi = 0. \tag{8.24}
\]

To get started let us assume that the computational domain is \( x \in < 0, L > \) and \( t \in < 0, T > \). The boundaries are then at \( x = 0, L \), where we assume that \( x = L \) is an open boundary, while \( x = 0 \) is a natural boundary. Furthermore we construct a grid in the \( x, t \) coordinates where \( x_j = (j - 1)\Delta x \) and \( t^n = n\Delta t \) (cf. Fig. 8.2).
Figure 8.2: Sketch of the mesh in the \( t, x \) plane close to the right-hand open boundary. The computational domain is then to the left of \( x = L \). The letters \( J, J-1, \) and \( J-2 \) denote grid points respectively at the open boundary, the first and second points inside the computational domain, while \( n, n-1, \) and \( n+1 \) denote the time levels.

Since (8.24) is an advection equation it is natural that we use one of the stable schemes developed in Section 5.1. We emphasize that it is important that the interior scheme and the scheme we use to solve the radiation equation has the same accuracy. Thus if the interior scheme is of second order accuracy in time and space then it is natural that we choose the leapfrog scheme. If the interior scheme is first order in time and space then it is natural that we choose a similar scheme for the radiation condition (8.24), say the upwind scheme. In the following we assume that the latter is true.

We then proceed using the upwind scheme for the radiation condition at \( x = L \), that is for \( j = J \). Assuming that \( c_\phi \geq 0 \), and following the notation of Figure 8.2, we get

\[
\frac{\phi_{j}^{n+1} - \phi_{j}^{n}}{\Delta t} + c_\phi \frac{\phi_{j}^{n} - \phi_{j-1}^{n}}{\Delta x} = 0
\]

or

\[
\phi_{j}^{n+1} = (1 - r_\phi)\phi_{j}^{n} + r_\phi\phi_{j-1}^{n}
\]

where

\[
r_\phi = c_\phi \frac{\Delta t}{\Delta x}. \tag{8.27}
\]

Equation (8.26) says that the radiation condition in essence is an interpolation of values from the interior and at previous times. The problem is that we don’t know the weighting function, that is, the phase velocity \( c_\phi \)? As suggested by Orlandski (1976) we might solve (8.20) with respect to the phase velocity (or \( r_\phi \)) and get

\[
r_\phi = \frac{\phi_{j}^{n+1} - \phi_{j}^{n}}{\phi_{j}^{n} - \phi_{j-1}^{n}}. \tag{8.28}
\]
However, since we do not know the solution at the boundary at time level \( n+1 \) this expression is useless. Our only way of determining \( c_\phi \) (or \( r_\phi \)) is to use our knowledge about the solution at previous times. We then have several options. One is to use interior points at the same time level, in which case

\[
    r_\phi = -\frac{\phi_{J-1}^{n+1} - \phi_J^n}{\phi_J^n - \phi_{J-2}^n}.
\]

(8.29)

A second is to use information at previous times at same points in space,

\[
    r_\phi = -\frac{\phi_J^n - \phi_{J-1}^{n-1}}{\phi_{J-1}^{n-1} - \phi_J^{n-1}}.
\]

(8.30)

Both of these expressions provides an expression for the phase velocity. But which is the correct one? If we interpret (8.29) and (8.30) in terms of characteristics as in the previous section (see also Section 5.5), we notice that (8.29) assumes that the slope of the characteristic through the \((x_J, t^{n+1})\) point to a first approximation equals the slope through the \((x_{J-1}, t^{n+1})\) point, while (8.30) assumes that it to a first approximation equals the slope through the point \((x_J, t^n)\). Following this argument a third option is to assume that the characteristic through \((x_J, t^{n+1})\) continues backward in times and crosses the time level \( n-1 \) between \( x_{J-1} \) and \( x_{J-2} \). This is tantamount to assume that to a first approximation the slope through \((x_J, t^{n+1})\) equals the slope through \((x_{J-1}, t^n)\). We then get the following expression for the phase velocity

\[
    r'_\phi = -\frac{\phi_{J-1}^{n-1} - \phi_J^{n-1}}{\phi_J^n - \phi_{J-1}^{n-1}}.
\]

(8.31)

On purpose we have used a prime for this expression, since it is a predictor for the phase velocity. We must require that the number returned is not negative. Hence we correct the result by defining \( r_\phi \) (no prime attached) as

\[
    r_\phi = \begin{cases} 
    r'_\phi & ; \quad 0 \leq r'_\phi \\
    0 & ; \quad r'_\phi < 0
    \end{cases}
\]

(8.32)

As argued by Røed and Cooper (1987) we think (8.32) is a better approximation. Consequently, we use the expression (8.32) to substitute for \( r_\phi \) in (8.26) when determining the new boundary value \( \phi_{J+1}^{n+1} \) at time level \( n+1 \). If our open boundary was at \( x = 0 \) the inequality sign in (8.32) must be reversed to ensure that the phase velocity then is negative.

As alluded to it is only pure wave problems where processes like non-linear interactions, friction, wind forcing and the Coriolis acceleration are neglected that satisfies the radiation condition (8.1). All realistic models employed today within oceanography or meteorology are much more complex, and includes at least the processes just mentioned, and in most cases many more. The radiation condition is therefore far from being a perfect open boundary condition4.

As alluded to it is only pure wave problems where processes like non-linear interactions, friction, wind forcing and the Coriolis acceleration are neglected that satisfies the radiation condition (8.1). All realistic models employed today within oceanography or meteorology are much more complex, and includes at least the processes just mentioned, and in most cases many more. The radiation condition is therefore far from being a perfect open boundary condition4.

Since the radiation condition in most cases is far from being perfect the meteorological and oceanographic communities has developed several other optional OBCs (e.g., Chapman, 1985; Røed and Cooper, 1986, 1987; Palma and Matano, 2000; Blayo and Debreu, 2006). In the following sections we study some of the more popular ones.

---

4In fact it may be shown that there is no such thing as a perfect boundary condition mathematically speaking, since the problem in a geophysical context is ill posed.
8.4 The sponge

One of the most popular ones is the so called *sponge condition*. In essence the method is to extend the computational domain outside of the area of interest (interior domain) to include an area where the energy contained in that part of the solution leaving the interior domain is gradually decreased. In practice we achieve this by gradually increasing the relative importance of those terms associated with diffusive or frictional processes as the solution is advected or propagated into the exterior or extended domain (sometimes referred to as the sponge domain).

As an example let us study the problem formulated in association with (8.5), but with some friction added. Thus the governing equations become

\[
\begin{align*}
\partial_t u &= -g \partial_x h - \gamma u, \quad (8.33) \\
\partial_t h &= -H \partial_x u, \quad (8.34)
\end{align*}
\]

where \( \gamma \) is a constant in the interior domain, say \( \gamma = \gamma_0 \). As in the former problem we solve (8.33) and (8.34) within the interior domain \( x \in (0, L) \) where \( x = 0 \) and \( x = L \) are open boundaries. Outside of the interior domain we extend the computational domain to include areas within which we let the frictional parameter \( \gamma \) increase gradually. For instance at the right-hand open boundary \( x = L \) we add a buffer from \( x = L \) to \( x = LL \) in which we let the value of \( \gamma \) increase exponentially, viz.,

\[
\gamma = \begin{cases} 
\gamma_0 & ; \quad 0 \leq x \leq L \\
\gamma_0 e^{\lambda(x-L)} & ; \quad L < x \leq LL
\end{cases}, \quad (8.35)
\]

where the parameter \( \lambda \) determines how fast or quickly the frictional effect increases.

We can derive an analytic solution to (8.33) and (8.34). We start by differentiating (8.34) with respect to time, and then substitute for \( \partial_t u \) from (8.33). We then get

\[
\partial_t^2 h + \gamma \partial_t h = gH \partial_x^2 h, \quad (8.36)
\]

Searching for wave like solution we let

\[
h = h_0 e^{\omega t} e^{i\alpha x}. \quad (8.37)
\]

Substituting this expression into (8.36) we get the dispersion relation

\[
\omega^2 + \gamma \omega + gH \alpha^2 = 0, \quad (8.38)
\]

\[
h = h_0 e^{-\frac{1}{2} \gamma t} e^{i\alpha(x-ct)}, \quad (8.39)
\]

where

\[
c = \sqrt{gH - \left(\frac{\gamma}{2\alpha}\right)^2}, \quad (8.40)
\]

\( \alpha \) is the wavelength and \( h_0 \) is a constant. We observe that as time evolves the amplitude of the solution within the sponge domain decreases. Furthermore, we notice that phase velocity (8.40)
decreases with increasing $\gamma$, that is, decreases as the wave propagates deeper into the sponge area. Thus as the solution to the system (8.33) and (8.34) propagates into the sponge where the frictional parameter $\gamma$ increases, it will be suppressed while decelerated. If we want to retain these properties it is important that $\gamma$ increases slowly within the sponge. To achieve this the exterior, sponge layer (or zone) added to the interior domain must be of a certain finite length. If $\gamma$ increases too quickly then the wave solution will not behave like given by (8.39) and possible reflection of wave energy might be generated.

Since application of the sponge condition as an open boundary condition requires that the sponge zone is of a certain extension, it adds a large amount of computer time to our problem, and hence slows down the wall clock time. Another problem with the sponge condition is that if the solution consists of forced waves (cf. Røed and Cooper, 1986), for instance is governed by equations like

$$\frac{\partial}{\partial t} u = -g \frac{\partial}{\partial x} h - \gamma u + \tau \quad (8.41)$$

$$\frac{\partial}{\partial t} h = -H \frac{\partial}{\partial x} u \quad (8.42)$$

where $\tau$ represents the forcing, then the solution in the sponge layer of the former wave solution and a solution dominated by a balance between the forcing term and the frictional term, that is, $u = \tau / \gamma$, which implies that as $\gamma$ increases $u$ decreases so that mass (volume) accumulates within the sponge zone. For longer term integrations this accumulation of volume changes the pressure forcing and sooner or later this will have an impact on the interior solution as well.

### 8.5 A weakly reflective OBC

As alluded to in Section 8.2 we may use the method of characteristics to construct a weakly reflective OBC also for problems including non-linearity’s, Coriolis affects and forcing (cf. Røed and Cooper, 1987).

As an example let us again study the problem governed by (8.5), where we add non-linear terms as well the Coriolis term and forcing,

$$\frac{\partial}{\partial t} u + u \frac{\partial}{\partial x} u - f v = -g \frac{\partial}{\partial x} h + F^x, \quad (8.43)$$

$$\frac{\partial}{\partial t} v + u \frac{\partial}{\partial x} v + f u = F^y, \quad (8.44)$$

$$\frac{\partial}{\partial t} h + \frac{\partial}{\partial x} (hu) = 0, \quad (8.45)$$

where $F^x, F^y$ are the forcing terms, and $f$ is the Coriolis parameter. We immediately recognize the system (8.43) - (8.45) as the shallow water equations for a barotropic fluid. Multiplying (8.45) by a yet unknown function $\lambda$, and adding to (8.43), and defining a common operator by

$$\frac{D^*}{dt} = \frac{D^* x}{dt} \frac{D^*}{\partial x}, \quad (8.46)$$

we get that $\lambda$ must satisfy the equation

$$\frac{D^* x}{dt} = u + \lambda h = u + \frac{g}{\lambda}. \quad (8.47)$$
Thus
\[ \lambda_{1,2} = \pm \frac{c}{h}, \quad c = \sqrt{gh}, \]  
and hence the compatibility equations associated with (8.43) and (8.45) and the appropriate characteristic equations are
\[ \left( \frac{D^x}{dt} \right)_{1,2} (u \pm 2c) = fv + F^x \quad \text{along} \quad \left( \frac{D^x}{dt} \right)_{1,2} = u \pm c. \]  

To avoid reflections at, say \( x = L \), we require
\[ \left. \left( \frac{D^x}{dt} \right)_{2} \right|_{x=L} = 0. \]  

By substitution of this expression in (8.49) the weakly reflective open boundary condition becomes,
\[ \partial_t(u + 2c) + (u + c)\partial_x(u + 2c) = fv + F^x, \quad \partial_t(u - 2c) = fv + F^x, \]
valid at \( x = L \). A finite difference approximation to these equation then gives a weakly reflective OBC.

### 8.6 The Flow Relaxation Scheme

We now construct an OBC that was first suggested by Davies (1976). As shown below it is somewhat similar to the sponge OBC in two respects. First it requires us to extend the computational domain to include an exterior domain or buffer zone. Second it is in essence a sponge in which the solution is suppressed as it progresses into the buffer zone. The method is called the Flow Relaxation Scheme commonly abbreviated to FRS (Davies, 1976; Martinsen and Engedahl, 1987; Cooper and Thompson, 1989; Engedahl, 1995a).

A description of the FRS nicely giving insight into its use as an OBC may be find in Shi et al. (1999) and Shi et al. (2001). One of the advantages of the FRS compared to for instance the sponge is that it allows us to specify an exterior solution. The FRS can therefore also be used as a one way nesting condition, in which an exterior solution is specified by, e.g., a courser grid model covering a much larger area. The FRS as a nesting technique is for instance used as the main method whereby information from global and semi-global model are transferred to regional models at Meteorologisk institutt. This is true for both their numerical weather prediction (NWP) models as well as their numerical ocean weather prediction (NOWP) models\(^5\).

In essence the method just modifies the numerical solution in a buffer zone, the so called FRS zone, for each time step, based on the interior solution and a specified exterior solution. The FRS zone commonly consists of a small number of grid points, say 5-20. We emphasize that

---

\(^5\)cf. http://met.no/
the FRS zone is an extension of the interior domain and thus extends the computational domain (cf. Figure 8.3). Within the FRS zone the solution is relaxed towards an apriori specified exterior solution, often referred to as the outer solution. The relaxation is performed by specifying a weighing function that for each grid point in the FRS zone computes a weighted mean between the specified outer solution and the interior solution computed from the governing equations.

Let \( \phi(x,t) \) be the dependent variable in our problem and let the interior domain be \( x \in \langle -M, L \rangle \), where \( x = 0 \) is an open boundary. As displayed in Figure 8.3 the FRS zone extends the interior domain so that the computational domain is increased to the left by adding an FRS zone starting at \( x = -M \) and ending at \( x = 0 \). As usual we define the grid points by \( x_j = (j - 1)\Delta x \), where the index \( j = 1(1)J \) covers all grid points of the computational domain, that is, both the FRS zone and the interior domain. Then \( j = J \) specifies the right-hand boundary of the computational domain, that is, \( x_J = L \), while \( j = 1 \) specifies the left-hand boundary of the FRS zone, that is, \( x_1 = -M \). Let the index of grid point where the FRS zone is attached to the interior domain (\( x = 0 \)) be \( j = JM \). Then \( x_{JM} = 0 \) defines the point which is at the left-hand open boundary. As alluded to the FRS allows us to specify an outer solution, which can be the result of another numerical model covering a larger domain than our interior domain. We denote this exterior solution by \((\phi_e)_j^n\) which emphasize that the outer solution is a function of space and time. Let us now assume that we have computed all the dependent variable \( \phi_j^n \) at all points, including the FRS zone at time level \( n \). Furthermore using the governing equation of our model we can derive a solution at the time level \( n + 1 \) including within the FRS zone, except of course at \( j = 1 \). We denote this predictor by \( \phi_j^* \). The next step is to correct the solution in the FRS zone by computing our dependent variable as a weighted mean between our predicted
solution $\phi_j^*$ and the specified outer solution $(\phi_e)_j^n$ to derive the final or corrected solution at time level $n + 1$. We do this by employing the formula

$$\phi_j^{n+1} = (1 - \alpha_j)\phi_j^* + \alpha_j(\phi_e)_j^{n+1}; \quad j = 1(1)JM, \quad (8.53)$$

where $\alpha_j$ denotes the weights. By letting $\alpha_j$ vary from $\alpha_1 = 1$ at the left-hand side of the FRS zone, or more stringent, at the outer edge of the FRS zone, ($j = 1$) to $\alpha_{JM} = 0$ at the open boundary at $j = JM$, we notice from (8.53) that at the outer edge of the FRS zone the numerical solution at time level $n + 1$ equals the specified outer solution, or $\phi_{1}^{n+1} = (\phi_e)^{n+1}$. Similarly we notice that at the open boundary the solution equals the interior solution, or $\phi_{JM}^{n+1} = \phi_{JM}^*$.  

Experiments employing the FRS shows that the solution is sensitive to the the distribution of the specified weighting function $\alpha$ throughout the FRS zone. It is also sensitive to the width of the FRS zone, e.g., Martinsen and Engedahl (1987), Engedahl (1995a). They found that distributing $\alpha$ applying a hyperbolic tangent function, that is,

$$\alpha_j = 1 - \tanh \frac{j - 1}{2}; \quad j = 1(1)JM, \quad (8.54)$$

is a good choice. Furthermore they concluded that for oceanic application the width of the FRS zone should be at least seven grid points, that is, $JM \geq 7$.

One of the disadvantages of employing FRS as an OBC (or nesting technique) is that the computational domain is increased, and hence that the computational burden is increased. This disadvantage is somewhat suppressed by the fact that the FRS allows us to specify an outer solution, which as shown below can be used to effectively minimize the error. Another disadvantage is that solution when applying the FRS as an OBC does not conserve fundamental properties such as volume (or mass).

As an example let us study the numerical solution of the continuous problem

$$\partial_t \phi = \mathcal{L}[\phi] \quad ; \quad x \in < 0, L >, \quad (8.55)$$

where $\mathcal{L}$ is a spatial differential operator. Furthermore, let us assume that the open boundary is at $x = 0$ and that $x = L$ is a natural boundary. As above we let $(\phi_e)_j^n; \quad j = 1(1)JM$ denote the specified exterior solution. If we solve (8.55) applying a forward in time finite difference scheme then we get

$$\phi_j^n = \phi_j^n + \Delta t \mathcal{L}_j^n \quad ; \quad j = 2(1)J - 1, \quad (8.56)$$

where $\phi_j^n$ is the predictor. We then correct the predictor by applying the relaxation formula (8.53). We then get

$$\phi_j^{n+1} = (1 - \alpha_j)\phi_j^* + \alpha_j(\phi_e)_j^{n+1} \quad ; \quad j = 1(1)J, \quad (8.57)$$

where we without loss of generality have extended (8.53) to the entire computational domain. To ensure that we do no corrections to the predictor within the interior domain we redefine the weighting function $\alpha_j$ as

$$\alpha_j = \begin{cases} 
1 - \tanh \frac{j - 1}{2} & ; \quad j = 1(1)JM - 1 \\
0 & ; \quad j = JM(1)J 
\end{cases} \quad (8.58)$$
We may now use the expression on the right-hand side of (8.56) to substitute for \( \phi_j^* \) in (8.57). If we in addition add the zero \( \alpha_j \phi_j^{n+1} \) we get

\[
\frac{\phi_j^{n+1} - \phi_j^n}{\Delta t} = L_j^n + \gamma_j \left[ (\phi_e)_j^{n+1} - \phi_j^{n+1} \right]
\]  

(8.59)

where the coefficient \( \gamma_j \) is defined by,

\[
\gamma_j = \frac{\alpha_j}{1 - \alpha_j}.
\]  

(8.60)

If we now let \( \Delta t \) and \( \Delta x \) tend to zero, we notice that (8.59) is a forward in time, finite difference approximation to the continuous equation

\[
\partial_t \phi = L[\phi] + \gamma (\phi_e - \phi) ; \quad x \in <0, L> .
\]  

(8.61)

We observe that except for the additional “frictional” term \( \gamma (\phi_e - \phi) \) equation (8.61) equals (8.55). We also notice that the additional term is proportional to the difference between the interior solution and the exterior solution, and that the proportionality factor (8.60) varies from zero at the open boundary \( (x = 0) \) to infinity at the edge of the FRS zone. Thus the relative importance of the frictional term increases as we progress into the FRS zone. If we now specify an exterior solution as constant and zero, then (8.61) is turned into

\[
\partial_t \phi = L[\phi] - \gamma \phi .
\]  

(8.62)

Under these circumstances the FRS acts like a sponge with a frictional parameter \( \gamma \) which gradually and monotonically increases towards infinity as we progress into the sponge, not unlike the exponential function specified in Section 8.4.

To illustrate the non-conservative properties of the FRS we use the example problem governed by (8.5) and (8.6). To get started we notice first that as \( \Delta x \) and \( \Delta t \) tend to zero (8.57) takes on the form

\[
\phi = (1 - \alpha)\phi^* + \alpha \phi_e .
\]  

(8.63)

We now have two dependent variables, namely \( h \) and \( u \), hence

\[
h = (1 - \alpha)h^* + \alpha h_e ,
\]  

(8.64)

\[
u = (1 - \alpha)u^* + \alpha u_e .
\]  

(8.65)

Substituting for \( u \) from (8.65) into (8.6) we get

\[
\partial_t h = -H \partial_x u = (1 - \alpha)\partial_x u^* - H \partial_x u_e + H u^* \partial_x \alpha
\]  

(8.66)

The two first terms on the right hand-side of (8.66) are the terms we would have obtained if we relax (8.5) directly assuming that (8.6) is valid for the interior as well as the exterior solution. However, since the relaxation parameter is a function of \( x \) also a term appears containing the gradient of the relaxation parameter \( \alpha \). Thus the volume (or mass) conservation expressed by
(8.6) is invalidated. To avoid the non-conservation of mass to impact our interior solution we must ensure that the relaxation parameter \( \alpha \) is a slowly varying function. In turn this implies that the width of the FRS zone must be long enough for this to be realized.

Finally we notice that if the exterior solution is equal to or close to the true solution, then the friction term in (8.62) disappear as do the false divergence in (8.66). Under these circumstances the FRS is close to being a perfect open boundary or nesting condition. Thus the usefulness of the FRS depends to a certain extent on how good we are to “guess” the exterior solution. This is the reason why the FRS is mostly used as a one way nesting condition. When specifying the exterior solution to be the solution of the same governing equation, albeit for a coarser mesh, we ensure that the exterior solution is indeed close to the interior solution of the fine mesh model embedded in the coarser grid.

Exercises

1. Show that a one-sided, finite difference scheme in time and space of the radiation condition (8.1) can be written

\[
\phi_B^{n+1} = \begin{cases} 
\phi_B^n & ; \ c_\phi > 0 \\
\frac{\phi_B^n}{(1 + c_\phi \Delta t/\Delta x)} \phi_B^n - c_\phi \frac{\Delta t}{\Delta x} \phi_{B+1}^n & ; \ c_\phi \leq 0
\end{cases}
\]

The open boundary is to the left so that subscript \( B \) denotes the values of the variables on the open boundary while subscript \( B + 1 \) indicates the values to the right of the open boundary.

2. Show by use of (8.67) that the radiation condition is “simply” an interpolation of values on the inside of the computational domain.
Chapter 9
ADVANCED TOPICS

9.1 Higher order advection schemes

As alluded to in Section 2.6 schemes with higher order accuracy may be constructed using Taylor series expansion. As an example we show how to construct a fourth order accurate scheme for the advection equation

$$\partial_t \theta + u \partial_x \theta = 0.$$ (9.1)

First we recall from the section on Taylor expansions (Section 2.6) that (2.21) and (2.22) in our notation becomes

$$\theta^n_{j \pm 1} = \theta^n_j \pm [\partial_x \theta]^n_j \Delta x + \frac{1}{2} [\partial^2_x \theta]^n_j \Delta x^2 \pm \frac{1}{6} [\partial^3_x \theta]^n_j \Delta x^3 + \mathcal{O}(\Delta x^4).$$ (9.2)

Thus follows that

$$\frac{\theta^n_{j+1} - \theta^n_{j-1}}{2\Delta x} = [\partial_x \theta]^n_j + \frac{1}{6} [\partial^3_x \theta]^n_j \Delta x^2 + \mathcal{O}(\Delta x^4),$$ (9.3)

which we then used to provide a second order approximation to the spatial derivative in (9.1) by simply dropping terms $\mathcal{O}(\Delta x^2)$ and higher. In (9.3) we have used the points adjacent to $\theta^n_j$ to construct the series, that is, the points $\pm \Delta x$ away. Suppose we used points located $\pm 2\Delta x$ away instead. Then the Taylor series (9.2) becomes

$$\theta^n_{j \pm 2} = \theta^n_j \pm [\partial_x \theta]^n_j 2\Delta x + \frac{1}{2} [\partial^2_x \theta]^n_j (2\Delta x)^2 \pm \frac{1}{6} [\partial^3_x \theta]^n_j (2\Delta x)^3 + \mathcal{O}(\Delta x^4)$$ (9.4)

and hence that

$$\frac{\theta^n_{j+2} - \theta^n_{j-2}}{4\Delta x} = [\partial_x \theta]^n_j + \frac{2}{3} [\partial^3_x \theta]^n_j \Delta x^2 + \mathcal{O}(\Delta x^4).$$ (9.5)

Use of (9.5) to construct a second order finite difference approximation to $[\partial_x \theta]^n_j$ by neglecting all terms $\mathcal{O}(\Delta x^2)$ and higher is as valid as using (9.3). In the limit $\Delta x \to 0$ they both tend to $\partial_x \theta$, that is, both are numerically consistent. We may therefore combine them linearly to give

$$a \frac{\theta^n_{j+1} - \theta^n_{j-1}}{2\Delta x} + b \frac{\theta^n_{j+2} - \theta^n_{j-2}}{4\Delta x} = (a + b) [\partial_x \theta]^n_j + \frac{1}{6} (a + 4b) [\partial^3_x \theta] \Delta x^2 + \mathcal{O}(\Delta x^4),$$ (9.6)
where \(a\) and \(b\) are linear weights yet to be found. Solving (9.6) with respect to \([\partial_x \theta]^n_j\) then gives

\[
[\partial_x \theta]^n_j = \left(\frac{a}{a + b}\right) \frac{\theta_{j+1}^n - \theta_{j-1}^n}{2\Delta x} + \left(\frac{b}{a + b}\right) \frac{\theta_{j+2}^n - \theta_{j-2}^n}{4\Delta x} - \frac{1}{6} \left(\frac{a + 4b}{a + b}\right) [\partial_x^2 \theta] \Delta x^2 + \mathcal{O}(\Delta x^4).
\]

(9.7)

We may now use (9.7) to construct a fourth order finite difference approximation to \(\partial_x \theta\) by choosing the numbers \(a\) and \(b\) so that \(a + 4b = 0\) which eliminates the second term on the right-hand side of (9.7). The leading truncation error then becomes \(\mathcal{O}(\Delta x^4)\). Another requirement is that the scheme is consistent. To ensure consistency the right-hand side of (9.7) must tend to \(\partial_x \theta\) when \(\Delta x \to 0\), which requires that \(a + b = 1\). Thus we have two equations to solve for the two unknown weights \(a\) and \(b\) which gives \(a = \frac{1}{3}\) and \(b = -\frac{1}{3}\). Hence a numerically consistent, fourth order in space and second order in time scheme for the advection equation is

\[
\frac{\theta_{j+1}^n - \theta_{j-1}^n}{2\Delta t} + u \left\{ \frac{4}{3} \frac{\theta_{j+1}^n - \theta_{j-1}^n}{2\Delta x} - \frac{1}{3} \frac{\theta_{j+2}^n - \theta_{j-2}^n}{4\Delta x} \right\} = 0.
\]

(9.8)

or

\[
\theta_{j+1}^n = \theta_{j-1}^n - \frac{4}{3} \frac{\Delta t}{\Delta x} \left[ \theta_{j+1}^n - \theta_{j-1}^n - \frac{1}{8} (\theta_{j+2}^n - \theta_{j-2}^n) \right].
\]

(9.9)

As we did in Section 5.9 we may analyze this scheme in terms of its dispersion properties. We start by decomposing \(\theta_{j}^n\) into its Fourier components in time and space, that is, we write

\[
\theta_{j}^n = \Theta_{0} e^{i \alpha (j \Delta x - cn \Delta t)}.
\]

(9.10)

We then substitute (9.10) into (9.8) to give

\[
c = \frac{1}{\alpha \Delta t} \arcsin \left\{u \alpha \Delta t \left[ \frac{4}{3} \left(\frac{\sin \alpha \Delta x}{\alpha \Delta x}\right) - \frac{1}{3} \left(\frac{\sin 2 \alpha \Delta x}{2 \alpha \Delta x}\right) \right]\right\}.
\]

(9.11)

To leading order in \(\alpha \Delta x\) we then obtain

\[
c \approx u \left\{1 - \frac{4}{5!} (\alpha \Delta x)^4 + \cdots \right\}.
\]

(9.12)

Recalling (see Section 5.9 on page 66) that the second order in space leapfrog scheme phase speed was

\[
c = \frac{1}{\alpha \Delta t} \arcsin \left[u \alpha \Delta t \left(\frac{\sin \alpha \Delta x}{\alpha \Delta x}\right) \right] \approx u \left\{1 - \frac{1}{3!} (\alpha \Delta x)^2 + \cdots \right\},
\]

(9.13)

we observe that (remember that \(0 \leq \alpha \Delta x \leq \pi\)) the fourth order scheme is nearly always superior to the second order scheme. This process of constructing higher order finite difference approximations may be continued. For example we note that the scheme

\[
\frac{\theta_{j+1}^n - \theta_{j-1}^n}{2\Delta t} + u \left\{ \frac{3}{2} \frac{\theta_{j+1}^n - \theta_{j-1}^n}{2\Delta x} - \frac{3}{5} \frac{\theta_{j+2}^n - \theta_{j-2}^n}{4\Delta x} + \frac{1}{10} \frac{\theta_{j+3}^n - \theta_{j-3}^n}{4\Delta x} \right\} = 0.
\]

(9.14)
is good to $O(\Delta x^6)$. The dispersion relation becomes

$$c = \frac{1}{\alpha \Delta t} \arcsin\left\{ u \alpha \Delta t \left[ \frac{4 \sin \alpha \Delta x}{3} - \frac{3 \sin 2\alpha \Delta x}{5} + \frac{1 \sin 3\alpha \Delta x}{10} \right] \right\}. \quad (9.15)$$

which to leading order gives

$$c \approx u \left\{ 1 - \frac{36}{7!} (\alpha \Delta x)^6 + \cdots \right\}. \quad (9.16)$$

Comparing (9.16) with (9.12) and (9.13) shows that the sixth order scheme is superior to the fourth order scheme and so on. Potential complications, however, can arise from these higher-order spatial treatments. These include the aforementioned computational modes (Section 5.7), troubles at boundaries, and more stringent conditions for numerical stability.

As an example of the latter let us consider the stability of the second order in time, fourth order in space accurate CTCS scheme (9.9). Using von Neumann’s method the equation for the growth factor becomes

$$G^2 + 2i\lambda G - 1 = 0; \quad \lambda = \frac{1}{3} C \sin \alpha \Delta x (4 - \cos \alpha \Delta x), \quad (9.17)$$

where $C = u \Delta t / \Delta x$ is the Courant number. Thus the solutions are

$$G_{1,2} = -i \lambda \pm \sqrt{1 - \lambda^2}. \quad (9.18)$$

As we experience for all CTCS schemes the two solutions for the complex growth factor have absolute values equal to one. Thus the scheme is neutrally stable under the condition that the radical in (9.18) is a positive definite quantity. Thus we get

$$|C| \leq \frac{3}{5} \quad \text{or} \quad \Delta t \leq \frac{3 \Delta x}{5|u|}, \quad (9.19)$$

which indeed is more restrictive than for the second order in space CTCS (leapfrog) scheme.

### 9.2 Non-linear instability

Towards the end of Section 3.2 we mentioned that every non-linear solution of a problem of hyperbolic nature in which friction is neglected will eventually become numerically unstable. This is independent of the time step chosen and is associated with the energy cascade towards smaller and smaller scales that is the nature of non-linear problems. Hence it is not sufficient, for instance in the case of solving the non-linear advection equation (5.1), to satisfy the linear CFL criterion.

To satisfy ourselves that this is indeed true it is enough to solve a simple non-linear advection problem like (5.1). Sooner or later disturbances of wavelengths in the range $2 \Delta x$ to $4 \Delta x$ crops up. These disturbances are at first small in amplitude but growing. At some stage into the calculation the solution falls short of satisfying the linear CFL condition and the solution blows up, that
9.2. NON-LINEAR INSTABILITY  

is, becomes linearly, numerically unstable. The solution is then useless. It is common to credit Phillips (1959) to be the first to demonstrate this phenomenon by analytic means. Richtmyer (1963) provided another example which is reproduced below. Robert et al. (1970) generalized the previous example.

Before entering into details we notice:

1. All good functions may be expanded in terms of a discrete set of waves or exponentials
2. In a linear system waves of different wavelengths exist independent from each other
3. In a non-linear system the latter is no longer true and waves of different wave numbers will interact and sometime generate waves of new periods
4. Given a finite grid of size \( \Delta x \) we have a band-limited wavenumber space, that is, only a finite number of discrete waves can exist

The first point is well known. It simply tells us that all good functions \( \Psi(x) \) of period \( 2L \) may be formulated into a Fourier series, that is, for \( x \in [-L, L] \) the function \( \Psi(x) \) is written as

\[
\Psi(x) = a_0 + \sum_{m=1}^{\infty} a_m \sin(\alpha_m x) + b_m \cos(\alpha_m x)
\]  \hspace{1cm} (9.20)

where \( \alpha_m = m\pi / L \) is the discrete wavenumber, \( a_m \) and \( b_m \) is the amplitude or energy associated with the wavenumber \( \alpha_m \) and \( a_0 \) is the mean or average value of \( \Psi \) for \( x \in [-L, L] \).

The second point tells us that if the system is linear there is no exchange of energy between them. Thus two wavetrains of different amplitude, wavelength and direction will pass each other without changing neither of them.

The third point emphasizes the fact that it is the non-linearity that causes exchange to happen. To illustrate this suppose we have a solution, say wind or current \( u(x, t) \), given by

\[
u(x, t) = \sum_n u_n(t) \sin(\alpha_n x).
\]  \hspace{1cm} (9.21)

Then nonlinear products will give rise to terms having wavenumbers which are the sum of and difference of the two original wavenumbers, e.g.,

\[
\sin(\alpha_1 x) \sin(\alpha_2 x) = \frac{1}{2} [\cos(\alpha_1 - \alpha_2)x - \cos(\alpha_1 + \alpha_2)x]
\]  \hspace{1cm} (9.22)

Thus in a non-linear case the two wavetrains will be different after the passage, that is, they will experience a change in either wavelength, as illustrated by (9.22), amplitude or direction.

The fourth and last point tells us that when we formulate the function \( \Psi(x) \) as a sum of discrete waves on a grid of size \( \Delta x \) we have a band-limited wavenumber space in which the shortest wave that can possibly be resolved is \( 2\Delta x \). Thus our wavenumber space is limited to wavenumbers \( \alpha_m \in [0, \pi / \Delta x] \). For a non-linear problem in which the various waves interact to produce waves of wavenumber \( \alpha > \pi / \Delta x \), that is waves of wavelengths shorter than \( 2\Delta x \),
they are unresolved by our grid. Unfortunately these unresolved waves are folded into some low wavenumber. In fact as displayed in Figure 9.1 a wave of wavelength $\frac{3}{4}\Delta x$ is indistinguishable from a wave of wavelength $4\Delta x$. Let us arbitrarily call $\alpha < \pi/2\Delta x$ low wavenumbers and $\pi/\Delta x < \alpha < \pi/2\Delta x$ high wavenumbers. The latter are then waves of wavelengths between $2\Delta x$ and $4\Delta x$ and corresponds to the shortest waves that are resolved by our grid of size $\Delta x$.

Figure 9.1: Displayed are the two waves of wavelength $4\Delta x$ (solid curve) and $\frac{3}{4}\Delta x$ (dashed curve), in a grid of grid size $\Delta x$. Note that our grid cannot distinguish between the unresolved wave of wavelength $\frac{3}{4}\Delta x$ and the resolved wave of wavelength $4\Delta x$. Thus the energy contained in the unresolved wave will be folded into the low wavenumber space represented by the $4\Delta x$ wave.

Points three and four above make us expect \textit{a priori} that even though all initial energy is low wavenumber (long waves), non-linear interactions will eventually provide variance (or energy) at high wavenumbers (short waves). This is easily verified by investigating the model problem we use below, which is a simple non-linear advection equation in one dimension, that is,\footnote{Note that (9.23) is the acceleration term in the momentum equation for a one-dimensional problem. Hence non-linearity is ubiquitous in all realistic atmospheric and oceanographic models.}

$$\partial_t u + u \partial_x u = 0. \tag{9.23}$$

The difference between (9.23) and the earlier advection equation, e.g., (5.1) on page 49, is the appearance of the nonlinear term $u \partial_x u$. Suppose we have a solution at a particular time level $n$ that is a monochromatic wave of wavelength $2\pi/\alpha$ where $\alpha$ is the wavenumber, that is, $u^n(x) = u_0 \sin \alpha x$. Then from (9.23), using a scheme that is centered in time, we get

$$u^{n+1} - u^{n-1} = -u_0^2 \sin \alpha x \partial_x \sin \alpha x = -u_0^2 \sin \alpha x \cos \alpha x = -\frac{1}{2} u_0^2 \sin 2\alpha x. \tag{9.24}$$

Hence the solution at the next time level is a wave of wavelength $2\pi/2\alpha$, that is, a wavelength half of that of the original wavelength at time level $n$. Thus, we observe, as expected, that all
the energy originally contained at low wavenumbers (long waves) end up at high wavenumbers unresolved by our grid. Due to the folding of the energy contained in the unresolved waves, the energy contained in the shortest wave resolved by our grid, that is, waves of wavelength $4\Delta x$, accumulates. Thus after a sufficient time period the numerical model blows up due to ordinary numerical, linear instability, even though the linear problem is numerically stable.

To inspect the non-linear instability in some more detail we use the example of Richtmyer (1963). We start with the assumption the problem has variance at wavelengths $\infty$ (zero wavenumber), $4\Delta x$ and $2\Delta x$, and that the model problem is the simple non-linear advection equation in one dimension (9.23). Let us as Richtmyer (1963) approximate (9.23) using the classic leapfrog (CTCS) scheme, that is

$$u_{j+1} = u_{j-1} - \frac{\lambda}{2} \left[ (u^n_{j+1})^2 - (u^n_{j-1})^2 \right]$$  \hspace{1cm} (9.25)

where $\lambda = \Delta t / \Delta x$. We will not concern ourselves with boundary conditions, but as in the von Neumann analysis of Section 4.3 on page 34 perform a local analysis. Then making use of (9.20) with $\Psi = u$ and $a_0 = V$ we can show that an exact, formal solution to (9.25) is

$$u^n_j = C^n_n \cos\left(\frac{\pi j}{2}\right) + S^n_n \sin\left(\frac{\pi j}{2}\right) + U^n_n \cos(\pi j) + V.$$  \hspace{1cm} (9.26)

We can identify the amplitudes $C^n_n, S^n_n$ as the amplitudes of a wave with length $4\Delta x$, $U^n_n$ as the amplitude of a wave of length $2\Delta x$, and $V$ as a wave of low wavenumber ($\alpha < \pi / 2\Delta x$) with zero wavenumber (infinite wavelength). When we substitute (9.26) into (9.25) and notice that $(u^n_{j+1})^2 - (u^n_{j-1})^2 = (u^n_{j+1} - u^n_{j-1})(u^n_{j+1} + u^n_{j-1})$ we obtain relationships among the amplitudes,

$$C^n_{n+1} - C^n_{n-1} = 2\lambda S^n_n(U^n_n - V)$$
$$S^n_{n+1} - S^n_{n-1} = 2\lambda C^n_n(U^n_n + V)$$
$$U^n_{n+1} = U^n_{n-1}.$$  \hspace{1cm} (9.27)

The last equation in (9.27) says that $U^n_n$ may take on different values for the odd and even time steps, say $A$ for odd time steps ($n = 1, 3, 5, \ldots$) and $B$ for the even time steps ($n = 2, 4, 6, \ldots$), that is, $U^2m = A$ and $U^2m-1 = B$ for $m = 1, 2, 3, \ldots$. By eliminating $S^n_n$ from the first equation in (9.27), we obtain

$$C^n_{n+2} - 2C^n_n + C^n_{n-2} = 4\lambda^2(A + V)(B - V)C^n_n.$$  \hspace{1cm} (9.28)

The question then arise. Is this solution stable in the von Neumann sense? As in the simple linear case, using the von Neumann method, we define a growth factor associated with the $4\Delta x$ wave. For the $4\Delta x$ wave to be stable the growth factor has to be less than or equal to one. Thus we first define the growth factor by letting $G \equiv C^n_{n+2} / C^n_n$. Substituting this into (9.28) we derive

$$G^2 - 2\gamma G + 1 = 0,$$  \hspace{1cm} (9.29)

where $\gamma$ is a real number given by

$$\gamma = 1 + 2\lambda^2(A + V)(B - V).$$  \hspace{1cm} (9.30)
The roots of (9.29) are

\[ G_{1,2} = \gamma \pm i\sqrt{1 - \gamma^2}. \]  

(9.31)

We notice that as long as the radical is real then

\[ |G_{1,2}| = \sqrt{\gamma^2 + 1 - \gamma^2} \equiv 1 \]

(9.32)

The 4\(\Delta x\) wave is therefore neutrally stable provided

\[ 1 - \gamma^2 \geq 0 \quad \text{or} \quad -1 \leq \gamma \leq 1. \]  

(9.33)

As is obvious it is only possible to satisfy (9.30) if the amplitude of the 2\(\Delta x\) wave is such that \(|A| < V\) and/or \(|B| < V\). This is violated when the amplitude of the 2\(\Delta x\) wave is large in comparison with the energy contained in the longer waves (low wavenumbers). In this case the 4\(\Delta x\) will grow exponentially and the scheme is unstable.

In numerical models of the atmosphere or ocean, it is important to damp out the smallest space scales to control non-linear instability. We notice that this may be done by adding explicit viscosity as described in Section 3.2 and Section 4.6 or with a dissipative finite difference method in which you control the dissipation.

9.3 Combined advection-diffusion

In Chapter 4 we learned that the diffusion equation was unstable when applying a centered in time, centered in space scheme (leapfrog scheme), while we in Chapter 5 learned that a forward in time, centered in space scheme (Euler scheme) was unstable for the advection equation. As mentioned earlier in Chapter 3 will most atmospheric and oceanographic problems contain both advection and diffusion in one and the same equation. The question is therefore what scheme should we employ when we have a problem that combines the two processes?

We will investigate this by seeking finite difference approximations to the continuous combined advection-diffusion equation (3.1). We start by using the parametrization given by () for the advective flux and () for the diffusive flux. Moreover, we simplify the problem by assuming that the problem is one-dimensional in space and that the velocity is constant, that is, \(v = u_0 i\) where \(u_0\) is a constant. Thus the advection-diffusion equation becomes

\[ \partial_t \theta + u_0 \partial_x \theta = \kappa \partial_x^2 \theta. \]  

(9.34)

To obtain a stable scheme we must ensure that the diffusive part is forward in time and the advective part is centered in time. We may for instance make use of the approximation

\[ \theta_j^{n+1} = \theta_j^{n-1} - \frac{u_0 \Delta t}{\Delta x} (\theta_{j+1}^n - \theta_{j-1}^n) + \frac{2\kappa \Delta t}{\Delta x^2} (\theta_{j+1}^{n-1} - 2\theta_j^{n-1} + \theta_{j-1}^{n-1}). \]  

(9.35)

We take note that the diffusive part is taken at time level \(n-1\) and thus becomes forward in time with a time step of \(2\Delta t\). In contrast the advective part is evaluated at time step \(n\) and is thus centered in time with a time step of \(\Delta t\). Hence each part is stable in itself. If \(\kappa = 0\) then the
advective part is stable if the Courant number \( C \equiv \frac{u_0 \Delta t}{\Delta x} \leq 1 \). If \( u_0 = 0 \) the diffusive part is stable under the condition \( \kappa \frac{\Delta t}{\Delta x^2} \leq \frac{1}{4} \). The factor \( \frac{1}{4} \) arises because of the \( 2\Delta t \) time step used for the diffusive part, and replaces the \( \frac{1}{2} \) factor in (4.32) on page 35.

In the general case with \( u_0 \neq 0 \) and \( \kappa \neq 0 \) we get a modified condition. We again use von Neumann’s method and substitute the single Fourier component into (9.35) to get an equation for the growth factor. The algebra is left to the reader (cf. Exercise 1 at the end of this Chapter) and gives

\[
G^2 + 2i\lambda G - \lambda_2 = 0
\]  

(9.36)

where

\[
\lambda = \frac{u_0 \Delta t}{\Delta x} \sin \alpha \Delta x \quad \text{and} \quad \lambda_2 = 1 - 4\left( \frac{\kappa \Delta t}{\Delta x^2} (1 - \cos \alpha \Delta x) \right)
\]  

(9.37)

are two real numbers. This equation has two roots given by

\[
G_{1,2} = -i\lambda \pm \sqrt{\lambda_2 - \lambda^2}.
\]

(9.38)

If the radical is positive the two roots are complex conjugates, but if the radical is negative the two roots become imaginary conjugate numbers. In the first case, which arises if \( \lambda_2 \geq \lambda^2 \geq 0 \) it follows that

\[
|G| = \sqrt{GG^*} = \sqrt{\lambda^2 + \lambda_2 - \lambda^2} = \sqrt{\lambda_2}
\]

(9.39)

and hence that the solution is stable because in this case \( 0 \leq \lambda_2 \leq 1 \). In the second case \( \lambda_2 \leq \lambda^2 \) in which case it follows that

We conclude that the sufficient condition for stability of the combined advection-diffusion scheme (9.35) is

\[
\frac{(u_0 \Delta t)^2 + 4\kappa \Delta t}{\Delta x^2} \leq 1.
\]

(9.40)

We note that that for either \( u = 0 \) or \( \kappa = 0 \), the stability condition for the individual advective and diffusive schemes are recovered. We also note that imposing teach condition is not a sufficient condition. We therefore obtain the somewhat surprising results that adding explicit diffusion in the advection equation actually reduces the maximum time step allowed for advection. What (9.40) says is that by adding diffusion we arrive at a more restrictive condition. This is visualized in Fig. (9.2).

For most cases in oceanography and meteorology this is not a serious problem since commonly

\[
\frac{\kappa \Delta t}{\Delta x^2} \ll \frac{u \Delta t}{\Delta x}, \quad u > 0
\]

(9.41)

We mentioned in Sections 4.6 and 9.2 that it is common to add a diffusion term to avoid non-linear problems to blow up due to non-linear instability (cf. also Section ). In this regard we also mentioned that it might be useful to use the inconsistent Dufort-Frankel scheme to approximate the diffusion term. In this case (9.34) would be approximated by

\[
\theta_j^{n+1} = \theta_j^{n-1} - \frac{u \Delta t}{\Delta x} (\theta_{j+1}^{n} - \theta_j^{n}) + \frac{2\kappa \Delta t}{\Delta x^2} (\theta_{j+1}^{n} - \theta_{j-1}^{n}) + \theta_j^{n} - \theta_j^{n-1} + \theta_j^{n+1}. \quad \theta_j^{n+1}
\]

(9.42)
We have then combined a consistent conditionally stable scheme for advection with an unconditionally stable, inconsistent scheme for diffusion. It can be shown that for the one-dimensional case it is sufficient to satisfy the CFL condition. In the more general case for instance for a two-dimensional case a more stringent condition has to be applied (Cushman-Roisin, 1984).

Many authors (cf. for instance Clancy, 1981) suggest to use the unstable forward in time, centered in space (FTCS) scheme when combining advection and diffusion. The approximation to (9.34) then becomes

\[
\theta_j^{n+1} = \theta_j^n - \frac{u_0 \Delta t}{2 \Delta x} (\theta_{j+1}^n - \theta_{j-1}^n) + \frac{\kappa \Delta t}{\Delta x^2} (\theta_{j+1}^n - 2 \theta_j^n + \theta_{j-1}^n). \quad (9.43)
\]

The amplification factor then follows the equation

\[
G = 1 - \frac{i u_0 \Delta t}{2 \Delta x} \sin \alpha \Delta x + \frac{2 \kappa \Delta t}{\Delta x^2} (\cos \alpha \Delta x - 1). \quad (9.44)
\]

As shown by Clancy (1981) the scheme is stable provided the two conditions

\[
\frac{\kappa \Delta t}{\Delta x^2} \leq \frac{1}{2}, \quad \text{and} \quad \frac{|u_0| \Delta t}{\kappa} \leq 1 \quad (9.45)
\]

are both satisfied at the same time. Despite the enthusiasm of several authors we do not recommend the use of the FTCS scheme. Rather we advocate to use the more conservative schemes (9.35) and (9.42).
9.4 The spectral method

When we applied the various approximation to the advection equation above, we only consider grid-point values of the dependent variables. We did not make any assumption about how the variables behaved between grid points other than assuming that they are good functions.

An alternative approach is to expand the dependent variables in terms of finite series of orthogonal functions (cf. Section 2.9 on page 20). The problem is then reduced to solving a set of ordinary differential equations which determine the behavior in time of the expansion coefficients. Following this approach is known as the spectral method.

The spectral method is particularly suitable for global atmospheric models where the dependent variables are zonally cyclic function and hence easy to expand. The method is therefore commonly applied in modern global atmospheric models, for instance in the global model used at the European Centre for Medium range Weather Forecast (ECMWF). We note that the method is a bit more cumbersome to apply in non-global models, and also in global oceanographic models since the latter have to deal with the continental land boundaries.

Application to the one-dimensional linear advection equation

We will demonstrate the spectral method for the one-dimensional advection equation on the globe, i.e., along a latitude. Under these circumstances the natural boundary condition is the periodic or cyclic boundary condition (cf. Section 2.5 on page 14).

We recall from Section 5.1 on page 50 that the one-dimensional advection equation is

\[ \frac{\partial \phi}{\partial t} = -u_0 \frac{\partial \phi}{\partial x}, \quad \text{for} \quad x \in [0, L] \quad \text{and} \quad t > 0 \quad (9.46) \]

where \( L \) is the length of the circumference at a particular latitude. We recall from Section 2.9 that (9.46) is just a special case of the general equation (2.48) on page 20. Hence the linear operator of Section 2.9 is \( \mathcal{H} = -u_0 \partial_x \). Since we will solve (9.46) along a latitude we first conveniently transform to a coordinate system in which

\[ 2\pi x = \xi L \quad \text{or} \quad \xi = \frac{2\pi x}{L} \quad (9.47) \]

where \( \xi \in <0, 2\pi> \) is the new (dimensionless) zonal coordinate. Since

\[ \partial_x \phi = \partial_\xi \phi \partial_x \xi \quad (9.48) \]

(9.46) then transforms to

\[ \frac{\partial \phi}{\partial t} = -\gamma \partial_\xi \phi \quad (9.49) \]

where

\[ \gamma = \frac{2\pi u_0}{L} \quad (9.50) \]

is the angular velocity (in units one per second). The cyclic boundary condition is written

\[ \phi(\xi, t) = \phi(\xi + 2\pi m) \quad m = 1, 2, 3, \ldots \quad (9.51) \]
where $m$ describes the number of times you have travelled around the world at that latitude. We further let the initial condition be described by the good function $f(\xi)$, and hence

$$\phi(\xi, 0) = f(\xi).$$

(9.52)

As outlined at the beginning of Section 5.1 on page 49 the true solution to (9.49) is then

$$\phi = f(\xi - \gamma t).$$

(9.53)

Solving (9.49) using expansions in terms of orthogonal functions requires us to choose a suitable set of expansion functions. The obvious choice in our case following (9.20) on page 114 above is to choose complex exponentials (sine and cosine functions), since these are eigenfunctions of the differential operator $\mathcal{H} = -u_0 \partial_x$. For a continuous function we get

$$\phi(\xi,t) = \sum_{\alpha} \phi_{\alpha}(t) e^{i\alpha \xi},$$

(9.54)

where $\alpha$ is the wave number and the summation is for all possible wavenumbers from $-\infty$ to $+\infty$. Solving (9.49) using numerical methods implies that we are band-limited in wavenumber space and hence we must use a truncated version of (9.54), that is,

$$\phi(\xi,t) = \sum_{l=-l_{\text{max}}}^{l_{\text{max}}} \phi_l(t) e^{i\alpha_l \xi}. $$

(9.55)

where $l = l_{\text{max}}$ gives the maximum wave number $\alpha_{l_{\text{max}}}$ resolved on the grid, that is the shortest wavelength resolved by our choice of grid size (here $2\Delta \xi$). Since $\phi_{-l} = \phi^*_l$, we need only be concerned with $0 \leq l \leq l_{\text{max}}$, rather than the full set of expansion coefficients.

We now substitute (9.55) into (9.49) and equate coefficients of the expansion functions. Thus

$$\partial_t \phi_l = -i \alpha_l \gamma \phi_l \quad \text{for} \quad l = 0(1)l_{\text{max}}$$

(9.56)

giving $2l_{\text{max}} + 1$ equations for the expansion coefficients $\phi_l$’s. For this particular case (9.56) can be integrated exactly for each wavenumber $\alpha_l$ separately to give

$$\phi_l(t, \alpha_l) = \phi_l(0, \alpha_l) e^{-i\alpha_l \gamma t}$$

(9.57)

where $\phi_l(0, \alpha_l)$ is the initial condition associated with the wavenumber $\alpha_l$. If we expand the good function $f(\xi)$ in terms of a truncated Fourier series, that is,

$$f(\xi) = \sum_l a_l e^{i\alpha_l \xi} = \sum_l \phi_l(0, \alpha_l) e^{i\alpha_l \xi},$$

(9.58)

we get that $\phi_l(0, \alpha_l) = a_l$, and hence that the complete solution to (9.49) is

$$\phi(\xi,t) = \sum_{l=0}^{l_{\text{max}}} a_l e^{i\alpha_l(\xi - \gamma t)},$$

(9.59)
which is the same as the true solution. Hence there is no dispersion due to the space discretization, unlike in the finite difference approximation above. This fact is due to the space derivatives being computed analytically while they were approximated in the finite difference method. We recall due to the orthogonality property of the expansion functions, in our case $e^{i\alpha \xi}$, that

$$\phi_l = \int_0^{2\pi} \sum_m \phi_m e^{i\alpha_m \xi} e^{-i\alpha_l \xi} d\xi. \quad (9.60)$$

Thus by multiplying (9.55) by the complex conjugate of the expansion functions and integrating in space we get

$$\phi_l(t; \alpha_l) = A_l \int_0^{2\pi} \phi(\xi, t) e^{-i\alpha_l \xi} d\xi, \quad (9.61)$$

where $A_l$ are the normalization factors. Note that (9.61) is the so called direct Fourier transform. The normalization coefficients are determined from the initial condition, or by use of (9.61), that

$$A_l = \frac{a_l}{\int_0^{2\pi} \phi(\xi, 0) e^{-i\alpha_l \xi} d\xi}. \quad (9.62)$$

In practise we have at our disposal the grid points values of $\xi$ rather than a continuous function in space. Thus we know $\xi$ at $J+1$ points $\Delta \xi$ apart such that $\xi_j = j\Delta \xi$ where $j = 0, 1, 2, \ldots, J-1, J$ and where $\xi_J = 2\pi$. In this case we think of the truncated Fourier series of $\phi$ as given in (9.55) as representing an interpolating function which exactly fits the values of $\phi$ at the $J+1$ grid points. We then write (9.61) as a discrete direct Fourier transform,

$$\phi_l(t, \alpha_l) = A'_l \sum_{j=1}^J \phi(\xi_j, t) e^{-i\alpha_l \xi_j}, \quad (9.63)$$

where the normalization coefficients are found by discretizing (9.62),

$$A'_l = \frac{a_l}{\sum_{j=1}^J \phi(\xi_j, 0) e^{-i\alpha_l \xi_j}}. \quad (9.64)$$

The corresponding discrete inverse Fourier transform is then

$$\phi(\xi_j, t) = \sum_{l=-l_{max}}^{l_{max}} \phi_l(t, \alpha_l) e^{i\alpha_l \xi_j}. \quad (9.65)$$

Both (9.63) and (9.65) can be computed with the Fast Fourier Transform (FFT) algorithm. It can be shown that starting from the set $\phi_l(t, \alpha_l)$ going to the set $\phi(\xi_j, t)$ with $j = 0, 1, 2, \ldots, J-1, J$ and returning to the set $\phi_l(t, \alpha_l)$ we recover exactly the original values provided the number of grid points $J$ are such that $J > 2l_{max} + 1$. Recall that $l_{max}$ is the number of waves used to compute the direct Fourier transform in (9.65). In addition we must require that the points $\xi_j$ are equally spaced or that $\Delta \xi$ is a constant.
It remains to find the expansion coefficients $\phi_l(t, \alpha_l)$ at an arbitrary time given their initial values $\phi_0$. We do this by a time stepping procedure, for instance applying a centered in time scheme to (9.56),
\[
\phi^{n+1}_l = \phi^{n-1}_l + 2i\alpha_l \gamma \Delta t \phi^{n}_l \quad l = 1, 2, 3, \ldots, LM
\]
for each wavenumber $\alpha_l$. Using von Neumann’s method we show that numerical stability is ensured provided
\[
|\alpha_l \gamma \Delta t| \leq 1; \quad \forall l.
\]
Since the maximum wavenumber is $\alpha_{LM}$ we require $|\alpha_{LM} \gamma \Delta t| \leq 1$. Moreover, since the maximum dimensionless wavenumber\(^2\) is $\alpha_{LM} = L/2\Delta x$ it follows that the stability condition in terms of the Courant number $C = u_0 \Delta t / \Delta x$ is
\[
C \leq \frac{1}{\pi},
\]
which is actually more stringent than the one derived for the finite difference approximation. Although being more restrictive the spectral method and scheme has the great advantage that it is nearly non-dispersive, and that the dispersivity is very small even for the shortest waves of two grid lengths.

### 9.5 Two dimensional problems

To investigate the effect of including more than one-dimension in space we study below the effect on the numerical stability criterion when including two dimensions in space. Further expansion into three dimensions is then straightforward.

**Advection equation**

We start by expanding the simple one-dimensional advection equation to two dimensions in space. We then get
\[
\partial_t \theta + u_0 \partial_x \theta + v_0 \partial_y \theta = 0,
\]
where $u_0$ and $v_0$ are constant speeds in the $x$ and $y$ direction respectively. To solve (9.69) numerically let us employ the well known second order accurate CTCS (leapfrog) scheme. Thus
\[
\frac{\theta^{n+1}_{jk} - \theta^{n-1}_{jk}}{2\Delta t} + u_0 \frac{\theta^{n}_{j+1k} - \theta^{n}_{j-1k}}{2\Delta x} + v_0 \frac{\theta^{n}_{jk+1} - \theta^{n}_{jk-1}}{2\Delta y} = 0,
\]
where $\Delta y$ is the space increment along the $y$ axis. To investigate the numerical stability we employ von Neumann’s method. Hence we start by defining the discrete Fourier component in two dimensions, that is,
\[
\theta^n_j = \Theta_n e^{\alpha_j \Delta x} e^{\beta k \Delta y},
\]
\(^2\)The maximum dimensional wavenumber is $2\pi/2\Delta x$. 

123
where $\alpha$ and $\beta$ are wavenumbers in the $x$ and $y$ directions, respectively. Next we insert the discrete Fourier component into (9.70). We then get
\[
\Theta_{n+1} = \Theta_{n-1} - 2i\Theta_n \left( u_0 \frac{\Delta t}{\Delta x} \sin \alpha \Delta x + v_0 \frac{\Delta t}{\Delta y} \sin \beta \Delta y \right)
\] (9.72)
Thus the equation for the growth factor becomes
\[
G^2 + 2i\lambda G - 1 = 0 ; \quad \lambda = u_0 \frac{\Delta t}{\Delta x} \sin \alpha \Delta x + v_0 \frac{\Delta t}{\Delta y} \sin \beta \Delta y,
\] (9.73)
giving the two solutions
\[
G_{1,2} = i\lambda \pm \sqrt{1 - \lambda^2}
\] (9.74)
If we compare with the one-dimensional problem we observe that the only difference is in the definition of $\lambda$. As earlier we must require that the radical is a positive definite quantity and hence that
\[
|u_0 \frac{\Delta t}{\Delta x} \sin \alpha \Delta x + v_0 \frac{\Delta t}{\Delta y} \sin \beta \Delta y| \leq 1
\] (9.75)
For this to be valid for all possible choice of wavenumbers $\alpha$ and $\beta$ we must require
\[
|u_0| \frac{\Delta t}{\Delta x} + |v_0| \frac{\Delta t}{\Delta y} \leq 1 \quad \text{or} \quad \Delta t \leq \frac{\Delta x \Delta y}{|u_0| \Delta y + |v_0| \Delta x}.
\] (9.76)
If we let the grid be regular, that is, let $\Delta x = \Delta y = \Delta s$ and $u_0 = v_0 = c_0$ we get
\[
\Delta t \leq \frac{\Delta s}{2|c_0|}.
\] (9.77)
Thus we observe that increasing the dimension leads to a more stringent CFL condition. This is to be expected since the physical interpretation of the CFL condition says that the characteristic must be within the cone of influence from time level $n$.

**Diffusion equation**

We now consider the diffusion problem in two dimensions, that is,
\[
\partial_t \theta = \kappa (\partial_x^2 \theta + \partial_y^2 \theta).
\] (9.78)
As in the on-dimensional case (cf. Chapter 4) we employ the forward in time, centered in space scheme (FTCS). Thus we get
\[
\frac{\theta_{j+1}^{n+1}}{\Delta s} = \frac{\theta_{j}^{n+1}}{\Delta s} + \kappa \frac{\Delta t}{\Delta x^2} \left( \theta_{j-1}^{n} - 2\theta_{j}^{n} + \theta_{j+1}^{n} \right) + \kappa \frac{\Delta t}{\Delta y^2} \left( \theta_{j}^{n-1} - 2\theta_{j}^{n} + \theta_{j+1}^{n} \right).
\] (9.79)
To investigate the numerical stability of this scheme we use again von Neumann’s method. As in the advection case we have to use a discrete Fourier component that includes waves propagating
in the $x$ direction as well as the $y$ direction, that is, we have to employ (9.71). Inserting (9.71) into (9.79) then gives a growth factor

$$G = 1 + 2\kappa \frac{\Delta t}{\Delta x^2} (\cos \alpha \Delta x - 1) + 2\kappa \frac{\Delta t}{\Delta y^2} (\cos \beta \Delta y - 1).$$  \quad (9.80)

We observe that this expression is comparable to the one derived for the one-dimensional case, that is (4.29) on page 35, except that we have an additional term. Applying von Neumann’s criterion for numerical stability (4.26) we must require

$$-1 \leq 1 + 2\kappa \frac{\Delta t}{\Delta x^2} (\cos \alpha \Delta x - 1) + 2\kappa \frac{\Delta t}{\Delta y^2} (\cos \beta \Delta y - 1) \leq 1.$$

(9.81)

As in the one-dimensional case we note that the right-hand inequality is trivially satisfied. To satisfy the left-hand inequality we must require

$$\kappa \frac{\Delta t}{\Delta x^2} (1 - \cos \alpha \Delta x) + \kappa \frac{\Delta t}{\Delta y^2} (1 - \cos \beta \Delta y) \leq 1 \quad (9.82)$$

Since the left-hand side of (9.82) is maximum when $\cos \alpha \Delta x = \cos \beta \Delta y = -1$ we note that if

$$\kappa \frac{\Delta t}{\Delta x^2} + \kappa \frac{\Delta t}{\Delta y^2} \leq \frac{1}{2}$$

(9.83)

then the stability criterion is satisfied for all possible values of the wavenumbers $\alpha$ and $\beta$. Thus the time step is limited according to

$$\Delta t \leq \frac{1}{2\kappa} \frac{\Delta x^2 \Delta y^2}{\Delta x^2 + \Delta y^2}.$$  \quad (9.84)

In the special case when the grid is regular (a square grid), that is $\Delta x = \Delta y = \Delta s$, we get

$$\Delta t \leq \frac{\Delta s^2}{4\kappa}.$$  \quad (9.85)

If we compare (9.85) with the one dimensional case (4.32) we again notice that the allowed time step is reduced by a factor of two. Thus as in the advection case the inclusion of more than one dimension implies that the criterion for numerical stability becomes more stringent. In fact this is a general result which applies to all problems.

**Shallow water equations**

We now consider what happens to the shallow water problem when we include more than one dimension. As we did for the one dimensional case it is worthwhile to analyze the various motions supported by the two-dimensional problem first, that is, (6.16) - (6.18) on page 76. To this end we again use the fact that the solution consists of a sum of Fourier components and analyze the solution for one Fourier component. To take into account that we now have two
dimensions we now use the frequency rather than phase speed in the analysis. Hence we write the Fourier component as,

$$ x = x_0 e^{-i\omega t} e^{i(\alpha x + \beta y)}, $$  

(9.86)

where \( \omega \) is the frequency while \( \alpha \) and \( \beta \) are wave numbers in the \( x \)- and \( y \)-direction, respectively, all of which are assumed to be real quantities. The vector \( x \) consists of the three dependent variables, that is,

$$ x = \begin{bmatrix} u \\ v \\ \phi \end{bmatrix} \quad \text{and} \quad x_0 = \begin{bmatrix} u_0 \\ v_0 \\ \phi_0 \end{bmatrix}, $$  

(9.87)

where \( x_0 \) is the amplitude.

Inserting (9.86) into the linearized equations (6.16) - (6.18) we get

$$ i(\alpha \bar{u} + \beta \bar{v} - \omega) u - f v + i\alpha \phi = 0, $$  

(9.88)

$$ f u + i(\alpha \bar{u} + \beta \bar{v} - \omega) v + i\beta \phi = 0, $$  

(9.89)

$$ i\bar{\phi} \alpha u + i\bar{\phi} \beta v + i(\alpha \bar{u} + \beta \bar{v} - \omega) \phi = 0, $$  

(9.90)

which in turn may be formulated as the homogeneous linear equation,

$$ \mathcal{A} \cdot x = 0, $$  

(9.91)

where the tensor \( \mathcal{A} \) is

$$ \mathcal{A} = \begin{bmatrix} i(\alpha \bar{u} + \beta \bar{v} - \omega) & -f & i\alpha \\ f & i(\alpha \bar{u} + \beta \bar{v} - \omega) & i\beta \\ i\bar{\phi} \alpha & i\bar{\phi} \beta & i(\alpha \bar{u} + \beta \bar{v} - \omega) \end{bmatrix}. $$  

(9.92)

For non-trivial solutions to exist, the determinant of the tensor \( \mathcal{A} \) must be zero, which gives

$$ i(\alpha \bar{u} + \beta \bar{v} - \omega) \left[ i(\alpha \bar{u} + \beta \bar{v} - \omega)^2 - \bar{\phi}(\alpha^2 + \beta^2) + f^2 \right] = 0. $$  

(9.93)

As in the on-dimensional case we find that we get three solutions for the frequency \( \omega \), namely

$$ \omega_1 = \bar{u} \alpha + \bar{v} \beta, $$  

(9.94)

$$ \omega_2 = \bar{u} \alpha + \bar{v} \beta + \sqrt{c_0^2(\alpha^2 + \beta^2) + f^2}, $$  

(9.95)

$$ \omega_3 = \bar{u} \alpha + \bar{v} \beta - \sqrt{c_0^2(\alpha^2 + \beta^2) + f^2}, $$  

(9.96)

where

$$ c_0 = \sqrt{\bar{\phi}} = \sqrt{gH} $$  

(9.97)

is the wave phase speed. The first solution is simply the geostrophic balance in (6.14) and (6.15). We easily derive this interpretation by substituting \( \omega_1 \) from (9.94) into (9.88) and (9.89), respectively, that is, that is,

$$ 0 - f v + i\alpha \phi = 0, $$  

(9.98)

$$ f u + x_0 + i\alpha \phi = 0, $$  

(9.99)
which gives

\[ u = -\frac{1}{f}i\beta\phi \quad \text{and} \quad v = \frac{1}{f}i\alpha\phi \quad \Rightarrow \quad u = -\frac{1}{f}\partial_y\phi \quad \text{and} \quad v = \frac{1}{f}\partial_x\phi. \quad (9.100) \]

The last implication follows by using the Fourier solution backwards and shows that the geostrophic balance (6.14) is recovered.

The two other solutions represented by \( \pm\sqrt{\phi(\alpha^2 + \beta^2)} + f \) are combined inertia and gravity waves. The inertia part is associated with frequencies \( \omega \) proportional to \( f \), so called inertial oscillation, while gravity waves are associated with frequencies \( \omega = \sqrt{c_0^2(\alpha^2 + \beta^2)} \). Recalling that the dispersion relation for gravity waves is \( \omega = c_0\sqrt{\alpha^2 + \beta^2} \) which shows that \( c_0 \) is the phase speed of gravity waves. Note that the full solution to the linear problem is a sum over all possible solutions for all wave numbers \( \alpha \) and \( \beta \).

To solve (6.16) - (6.18) by numerical means we employ the CTCS (leapfrog) scheme. Hence

\[
\begin{align*}
\frac{u_{jk}^{n+1} - u_{jk}^{n-1}}{2\Delta t} - f v_{jk}^{n} + \frac{\phi_{j+1k}^{n} - \phi_{j-1k}^{n}}{2\Delta x} &= 0, \\
\frac{v_{jk}^{n+1} - v_{jk}^{n-1}}{2\Delta t} + f u_{jk}^{n} + \frac{\phi_{jk+1}^{n} - \phi_{jk-1}^{n}}{2\Delta y} &= 0, \\
\frac{\phi_{jk}^{n+1} - \phi_{jk}^{n-1}}{2\Delta t} + c_0^2 \frac{u_{jk+1}^{n} - u_{jk-1}^{n}}{2\Delta x} + i c_0^2 \frac{v_{jk+1}^{n} - v_{jk-1}^{n}}{2\Delta y} &= 0.
\end{align*}
\]

To investigate the numerical stability of the two-dimensional shallow water problem we again make use of von Neumann’s method. The difference is that the discrete Fourier component is

\[ x_{jk}^n = X_n e^{i(\alpha j \Delta x + \beta k \Delta y)} \]

where the transpose of the vector \( x_{jk}^n \) is \( x_{jk}^n = [u_{jk}^n, v_{jk}^n, \phi_{jk}^n] \) and the transpose of the vector \( X_n \) is \( X_n^T = [U_n, V_n, \Phi_n] \). Insertion into (9.101) - (9.103) then gives

\[
\begin{align*}
U_{n+1} - U_{n-1} &= 2f\Delta tV_n - 2i\Phi_n \frac{\Delta t}{\Delta x} \sin \alpha \Delta x, \\
V_{n+1} - V_{n-1} &= -2f\Delta tU_n - 2i\Phi_n \frac{\Delta t}{\Delta y} \sin \beta \Delta y, \\
\Phi_{n+1} - \Phi_{n-1} &= -2i c_0^2 U_n \frac{\Delta t}{\Delta x} \sin \alpha \Delta x - 2i c_0^2 V_n \frac{\Delta t}{\Delta y} \sin \beta \Delta y.
\end{align*}
\]

To find an equation for the growth factor we first eliminate \( V_n \) and \( U_n \). We do this by first replacing \( n \) by \( n + 1 \) in (9.105) and (9.106) followed by a replacement of \( n \) by \( n - 1 \). By subtracting the results we get

\[
\begin{align*}
U_{n+2} - 2U_n + U_{n+2} &= 2f\Delta t(V_{n+1} - V_{n-1}) \\
&- 2i(\Phi_{n+1} - \Phi_{n-1}) \frac{\Delta t}{\Delta x} \sin \alpha \Delta x.
\end{align*}
\]

127
\[ V_{n+2} - 2V_n + V_{n+2} = -2f \Delta t(U_{n+1} - U_{n-1}) \]
\[ - 2i(\Phi_{n+1} - \Phi_{n-1}) \frac{\Delta \varphi}{\Delta y} \sin \beta \Delta y . \]  

(9.109)

Substituting for \( V_{n+1} - V_{n-1} \) from (9.106) in the first equation and \( U_{n+1} - U_{n-1} \) from (9.105) in the second equation we get

\[ U_{n+2} - 2(1 - 2f^2 \Delta t^2)U_n + U_{n+2} = -4i f \Phi_n \frac{\Delta t^2}{\Delta y} \sin \beta \Delta y \]
\[ - 2i(\Phi_{n+1} - \Phi_{n-1}) \frac{\Delta \varphi}{\Delta x} \sin \alpha \Delta x , \]  

(9.110)

\[ V_{n+2} - 2(1 - 2f^2 \Delta t^2)V_n + V_{n+2} = 4i f \Phi_n \frac{\Delta t^2}{\Delta x} \sin \alpha \Delta x \]
\[ - 2i(\Phi_{n+1} - \Phi_{n-1}) \frac{\Delta \varphi}{\Delta y} \sin \beta \Delta y . \]  

(9.111)

We are now in a position to eliminate \( U_n \) and \( V_n \) from (9.107). To this end we first replace \( n \) by \( n + 2 \) in (9.107), and then replace \( n \) by \( n - 2 \). Adding the results and subtracting (9.107) multiplied by \( 2(1 - 2f^2 \Delta t) \) we get

\[ \Phi_{n+3} - \Phi_{n+1} - 2(1 - 2f^2 \Delta t^2)(\Phi_{n+1} - \Phi_{n-1}) + \Phi_{n+1} - \Phi_{n-3} = \]
\[ - 2i c_0^2 [U_{n+2} - 2(1 - 2f^2 \Delta t^2)U_n + U_{n-2}] \frac{\Delta t}{\Delta x} \sin \alpha \Delta x \]
\[ - 2i c_0^2 [V_{n+2} - 2(1 - f^2 \Delta t^2)V_n + V_{n-2}] \frac{\Delta t}{\Delta y} \sin \beta \Delta y . \]  

(9.112)

Thus substituting from (9.110) and (9.111) we finally get

\[ \Phi_{n+3} = A \Phi_{n+1} + A \Phi_{n-1} - \Phi_{n-3} = 0 \]  

(9.113)

where

\[ A = 1 + 2 \left\{ 1 - 2f^2 \Delta t^2 - 2 \left( c_0 \frac{\Delta t}{\Delta x} \right)^2 \sin^2 \alpha \Delta x - 2 \left( c_0 \frac{\Delta t}{\Delta y} \right)^2 \sin^2 \beta \Delta y \right\} \]  

(9.114)

Defining the growth factor by \( \Phi_{n+2} = G \Phi_n \) we get a third order equation for the growth factor, that is,

\[ G^3 - AG^2 + AG - 1 = 0 . \]  

(9.115)

One solution is \( G_1 = 1 \), while the two other solutions are given by the

\[ G_{2,3} = \frac{1}{2} (A - 1) \pm i \sqrt{1 - \frac{1}{4} (A - 1)^2} . \]  

(9.116)

As so many times before we have to require that the radical is real in which case \( |G_{1,2,3}| = 1 \) and the scheme is neutrally stable. The condition for this to be true is hence that

\[ \frac{1}{4} (A - 1)^2 \leq 1 \]  

(9.117)
or
\[-1 \leq 1 - 2f^2 \Delta t^2 - 2 \left( c_0 \frac{\Delta t}{\Delta x} \right)^2 \sin^2 \alpha \Delta x - 2 \left( c_0 \frac{\Delta t}{\Delta y} \right)^2 \sin^2 \beta \Delta y \leq 1 \] (9.118)

The right-hand inequality is no problem. The left-hand inequality gives
\[\left( c_0 \frac{\Delta t}{\Delta x} \right)^2 \sin^2 \alpha \Delta x + \left( c_0 \frac{\Delta t}{\Delta y} \right)^2 \sin^2 \beta \Delta y \leq 1 - f^2 \Delta t^2 \] (9.119)

and hence that
\[\Delta t \leq \frac{\Delta x}{c_0} \sqrt{\frac{\Delta y^2}{\Delta x^2 + \Delta y^2} (1 - f^2 \Delta t^2)}. \] (9.120)

In practice \(f^2 \Delta t^2 \ll 1\). Thus
\[\Delta t < \frac{\Delta x}{c_0} \sqrt{\frac{\Delta y^2}{\Delta x^2 + \Delta y^2}}. \] (9.121)

If we let \(\Delta x = \Delta y = \Delta s\) then we get
\[\Delta t < \frac{\Delta s}{c_0 \sqrt{2}}. \] (9.122)

We observe in comparison with the one-dimensional case as displayed in (6.49) that including more than one dimension in space results in a more stringent condition because of the appearance of the factor \(\sqrt{2}\) in the denominator. This result is general. As shown we got exactly the same result when expanding from one to two dimensions for the simple advection problem and the simple diffusion problem.

**Exercises**

1. Use von Neumann’s method (Chapter 4.3) to show that the expression (9.36) is indeed the correct expression for the growth factor when using the scheme given in (9.35).

2. Show that the condition (9.40) is a sufficient condition for numerical stability.


Bibliography


130


Mesinger, F., and A. Arakawa (1976), Numerical methods used in atmospheric models, GARP Publication Series No. 17, 64 p., World Meteorological Organization, Geneva, Switzerland.


