Fundamentals of Atmospheres and Oceans on Computers

Lars Petter Røed[†,⋆] and Thor Erik Nordeng[†,⋆]

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

PREFACE

These notes are directed towards Master students in Meteorology and Oceanography at the University of Oslo, Oslo, Norway following the course GEF4500 “Fundamentals of Atmospheres and Oceans on Computers”. We also hope they are of interest to others with an interest in the basic methods we use to put the atmosphere and the ocean on computers.

The notes are based on earlier notes used by the authors to taught the two independent courses GEF4230 “Numerical models of the atmosphere” and GEF4620 “Numerical methods to solve oceanographic problems”. Many of the basic methods we use in numerical oceanography and meteorology are originally developed within the meteorological community. Most of these are now adopted and widely used also in the oceanographic community. Thus we found that it was time to melt the two courses into one. A further rationale is the fact that the atmosphere and the ocean is inherently a coupled system, perhaps most evident within the field of climate modeling. We therefore feel its important that the students, whether they want to become meteorologists or oceanographers, obtain a basic knowledge of the numerical methods used within both communities. In particular this is important for future scientists who want to study climate and climate change. Be that as it may, atmospheres and oceans on computers is one of the most important tools in contemporary meteorology and oceanography. It is therefore essential for everyone who ventures into these fields and those who aspire to become a meteorologist or an oceanographer to have an insight into the fundamental methods used to develop numerical oceanographic and meteorological models.

Although the numerical methods used are to a large extent the same, there are methods and techniques that are unique to each field. It is therefore necessary to offer additional courses for those who will use a numerical model as their essential tool. Such courses are however directed towards Master and Ph.D. students who need to know the advenced methods used in either Meteorology or Oceanography.

These notes were written during the fall semester of 2007, at which time this course was offered for the first time. Although it is being compiled by one of us (LPR) there are also important contributions made by second author (TEN), most notable sections covering topic peculiar to meteorology. One of the auhors (LPR) would also like to extend his appreciation to Prof. James J. O’Brien, now distinguished professor emiritus at the Florida State University, USA and former director of the Center for Atmospher-Ocean Predictions (COAPS) for introducing him to numerical methods to solve oceanographic problems. A lot of the material covered in these lecture notes is extracted from his lectures at Florida State. The authors would also like to extend their appreciation to Dr. Arne Bratseth, former professor at the University of Oslo now deseased, for using material covered in his lecture notes on “Numerical Atmosphere Models” (previously GF322). Finally, we would like to extend our thanks to the many students who has pointed out misprints and errors in earlier versions and this version of the lecture notes.

Blindern, September 20, 2007

Lars Petter Røed and Thor Erik Nordeng (sign.)

Innhold

Figurer

Kapittel 1

INTRODUCTION

1.1 The governing equations

In the atmosphere and ocean the most prominent dependent variables are the three components u, v, w of the three-dimensional velocity[1] $\mathbf{v}$[2], pressure p, density ρ, (potential) and temperature θ. For the atmosphere also humidity q and cloud liquid water content q_L must be included, and regarding the ocean the salinity S must be included. Except for the velocity $\mathbf{v}$ it is common to refer to the remaining unknowns as state variables. To determine these unknowns we need an equal number of equations, normally referred to as the equations that governs the motion of the two spheres atmosphere and ocean, or simply the governing equations. Note that the state variables are scalars while the velocity is a vector. The state variables are therefore examples of what is usually referred to as tracers. Some of them, like density, pressure, salinity, temperature and humidity, are active tracers in the sense that they influence the motion, while others are passive. An example of the latter is any dissolved chemical component or substance that passivly follows the motion.

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 (or tracer concentration). For the atmosphere and ocean the governing equations in their non-Boussinesq form are[3]

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{1.1}$$

$$\partial_t (\rho \mathbf{v}) + \nabla \cdot (\rho \mathbf{v}\mathbf{v}) = -2\rho \boldsymbol{\Omega} \times \mathbf{v} - \nabla p + \rho \mathbf{g} - \nabla \cdot (\rho \boldsymbol{\mathcal{F}}_M) \tag{1.2}$$

$$\partial_t (\rho C_i) + \nabla \cdot (\rho C_i \mathbf{v}) = -\nabla \cdot (\rho \boldsymbol{\mathcal{F}}_i) + \rho S_i \quad i = 1, 2, \cdots \tag{1.3}$$

$$\rho = \rho(p, C_1, C_2) \tag{1.4}$$

where the tracers C_i represent any of the above state variables, $\boldsymbol{\mathcal{F}}_M, \boldsymbol{\mathcal{F}}_i$ are flux tensors represen-

[1]The velocity is a vector which is normally referred to as the wind vector in the atmosphere and the current in the ocean

[2]In the following bold upright fonts, e.g., $\mathbf{v}$, are used to denote a vector, while bold special italic fonts, e.g., $\boldsymbol{\mathcal{F}}$, are used to denote tensors

[3]See for example *Gill* (1982) or *Griffies* (2004)

ting the processes of turbulent mixing of momentum and tracers, respectively, $\mathbf{\Omega}$ is the Earth's rotation rate, $\mathbf{g}$ is the gravitational acceleration and S_i is a tracer source. The operator $\frac{D}{dt}$ is the material derivative[4], that is,

$$\frac{D}{dt} = \partial_t + \mathbf{v} \cdot \nabla. \tag{1.5}$$

The first of these equations (1.1) is the conservation of mass. The second (1.2) constitutes the conservation of momentum, while the third (1.3) is the tracer conservation equation. The fourth and last (1.4) is the equation of state. We note that except for the equation of state these equations are all partial differential equations or PDEs.

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

$$p = \rho R \theta \tag{1.6}$$

where R is the gas constant. In the ocean, however, no such simple relation exists. For the ocean it is a highly non-linear relation between the state variables pressure, density, temperature and salinity, which in general cannot be expressed in a formal analytic closed form. It is common to visualize the equation of state for the ocean in a so called T-S diagram where the salinity S is drawn along the horizontal axis while the (potential) temperature θ is drawn along the vertical axis as shown in Figure

1.2 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 accelaration, Dw/dt, is small when compared to the, e.g., the gravitational acceleration ρg. Thus the vertical momentum eqution is replaced by the hydrostatic equation in which the gravitational acceleration is balanced by the vertical pressure gradient only. 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 this we first write down the vertical component of the momentum equation using (1.2), that is,

$$\partial_t(\rho w) + \nabla \cdot (\rho w \mathbf{v}) = -\partial_z p - \rho g - \nabla \cdot (\rho \mathbf{F}_M^V), \tag{1.7}$$

where $\mathbf{F}_M^V$ is the vertical vector component of the mixing tensor. In many cases, but not all, the vertical velocity and its associates acceleration terms $\partial_t(\rho w)$ and $\nabla \cdot (\rho w \mathbf{v})$ are small and the latter can safely be neglected compared to the gravitational acceleration. 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

$$\partial_z p = -\rho g, \tag{1.8}$$

which is the hydrostatic equation[5]. The horizontal component becomes

$$\partial_t(\rho \mathbf{u}) + \nabla_H \cdot (\rho \mathbf{u}\mathbf{u}) + \partial_z(\rho w \mathbf{u}) = -\rho f \mathbf{k} \times \mathbf{u} - \nabla_H p - \nabla_H \cdot (\rho \boldsymbol{\mathcal{F}}_M^H) \tag{1.9}$$

[4]Also referred to as the individual derivative

[5]The name is based on the fact that in a fluid left at rest, i.e., static, in the gravitational field also satisfies this equation

where $\nabla_H = \mathbf{i}\partial_x + \mathbf{j}\partial_y$ is the horizontal component of the three-dimension del-operator, $f = 2\Omega \sin\phi$ is the Coriolis parameter where ϕ is the latitude and Ω is the Earth's rotation rate, and $\boldsymbol{\mathcal{F}}_M^H$ is the horizontal component of the three-dimensional flux tensor due to turbulent mixing. The tracer equation is left unchanged, but as in the momentum equation we may single out the vertical acceleration term and thus write

$$\partial_t(\rho C_i) + \nabla_H \cdot (\rho C_i \mathbf{u}) + \partial_z(\rho C_i w) = -\nabla \cdot (\rho \boldsymbol{\mathcal{F}}_i) + \rho S_i \quad i = 1, 2, \cdots . \tag{1.10}$$

1.3 The Boussinesq approximation

One common approximation employed in ocean models is the so called *Boussinesq approximation*. One fundamental basis for this approximation is the fact that the ocean, in contrast to the atmosphere, is nearly incompressible. This means that for any parcel of fluid the change in density is small with respect to the density itself, that is,

$$\frac{1}{\rho}\frac{D\rho}{dt} = \frac{D \ln \rho}{dt} \approx 0 \tag{1.11}$$

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

$$\nabla \cdot \mathbf{v} = 0. \tag{1.12}$$

As a consequence the Boussinesq ocean is not allowed to change its volume, but is free to change its density and mass. This is in contrast to a non-Boussineq ocean which is not allowed to change its mass, but is free to change its density and volume. This has an important impact on the Boussinesq ocean's ability to expand due to heating. Heating a parcel of fluid in a Boussinesq ocean entails that its density is decreased while its volume is maintained. Thus no expansion is allowed, but to maintain its volume when heated the parcel loose mass. This is of 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.

Under these circumstances the momentum equation (1.2) becomes

$$\partial_t \mathbf{u} + \nabla \cdot (\mathbf{v}\mathbf{u}) = -f\mathbf{k} \times \mathbf{u} - \rho_0^{-1}\nabla_H p - \nabla_H \cdot \boldsymbol{\mathcal{F}}_M^H, \tag{1.13}$$

where ρ_0 is a reference density. Similarly the tracer conservation equation (1.3) reduces to

$$\partial_t C_i + \nabla \cdot (C_i \mathbf{v}) = -\nabla \cdot \mathbf{F}_i + S_i \tag{1.14}$$

for $i = 1, 2, \cdots$. Finally we note that for particular phenomena in the atmosphere and oceans even further approximations may be invoked so that the governing equations (1.1) - (1.4) reduces to the fairly "simple" shallow water equations as given in Section 1.6.

Finally we note that the introduction of more and more simplifications sometimes complicates the numerical problem. For instance the fairly popular rigid lid approximation implies that the equations must be solved globally rather than locally since the solution at one point not only depends on its nearest neighboring points, but in fact depends on all the points within the computational domain. We will come back to this fact in Section 1.5 below when treating elliptic problems, since the rigid lif approximations requires the solution to an elliptic equation for each time step.

1.4 Partial differential equations

Thus we observe that when solving the governing equations for the atmosphere and ocean we are fundamentally concerned with solving a set of equations referred to as partial differential equations (PDEs). In general a PDE may be written

$$a\partial_{x'}^2\theta + 2b\partial_{x'}\partial_{y'}\theta + c\partial_{y'}^2\theta + 2d\partial_{x'}\theta + 2e\partial_{y'}\theta + f\theta = g. \tag{1.15}$$

Here $\partial_{x'}, \partial_{y'}$ denotes differentiation with respect to the independent variables ξ, η, while $\theta = \theta(\xi, \eta)$ denotes the dependent variable. The coefficients $a, b, .., g$ are in general functions of the independent variables, that is, $a = a(\xi, \eta)$, etc. Note that there is no physical interpretation associated with either the independent or the dependent variables. Thus ξ og η represents any independent variable, for instance time or one of the spatial variables, while θ represent any dependent variable, for instance velocity, pressure, density, salinity, humidity or a general tracer.

1.5 Elliptic, parabolic and hyperbolic equations

If $b^2 - ac > 0$ then the roots of (1.15) are real and distinct. The corresponding PDE is then *hyperbolic*. The classic example is the wave equation,

$$\partial_t^2\phi - c_0^2\partial_x^2\phi = 0. \tag{1.16}$$

To derive (1.16) from (1.15) we let $u = \theta$, $\xi = t$, $\eta = x$, and let $a = 1$, $b = 0$, $c = -c_0^2$, $d = e = f = g = 0$. Then follows that $b^2 - ac = 0 - (-c_0^2) = c_0^2$ which indeed is positive.

By defining

$$\Phi = \partial_t\phi - c_0\partial_x\phi \tag{1.17}$$

we note that the wave equation (1.16) becomes

$$\partial_t\Phi + c_0\partial_x\Phi = 0. \tag{1.18}$$

Since c_0 is a constant (1.18) may be written

$$\partial_t\Phi + \partial_x(\Phi c_0) = 0. \tag{1.19}$$

This equation, which is commonly referred to as the *advection equation*, is a actually a one-dimensional version of (1.14) with $\mathbf{v}$ replaced by $c_0\mathbf{i}$ and a zero on the right-hand side. Thus the

advection equation is of fundamental importance in the modelling of atmospheric and oceanographic motions. It also indicates that the equations governing atmospheric and oceanographic motions are inherently hyperbolic.

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

$$\partial_t C = \kappa \partial_x^2 C, \tag{1.20}$$

where κ is the diffusion coefficient (heat capacity). To arrive at (1.20) from (1.15) we let $\xi = x, \eta = t, a = 1, b = c = d = f = g = 0$, and $e = 1/2$. Equation (1.20) is a simplification of the full three-dimensional diffusion equation

$$\partial_t C = \nabla \cdot (\mathcal{K} \cdot \nabla C) = -\nabla \cdot \mathbf{F}_C \tag{1.21}$$

where $\mathbf{F}_C = -\mathcal{K} \cdot \nabla C$ is the diffusive flux vector and $\mathcal{K}$ is a matrix (dyade) describing the conductive efficiency of the medium with regard to the tracer C. Thus $\mathcal{K} = \kappa_{mn}\mathbf{i}_m\mathbf{j}_n$, $m, n = 1, 2, 3$. To retrive (1.20) we simply let $\kappa_{11} = \kappa$ and $\kappa_{mn} = 0$ for $m \neq 1$ og $n \neq 1$ and assume that κ is constant. Let us for a moment assume that the atmosphere or ocean is at rest, that is, $\mathbf{v} = 0$ and that there is no sources for the tracer ($S = 0$). Then (1.14) reduces to (1.21), which implies that also the diffusion balance is of fundamental importance when solving atmospheric and oceanographic problems. Hence atmospheric and oceanographic motions are inherently both parabolic and hyperbolic.

Finally, if $b^2 - ac < 0$ then the roots of (1.15) are imaginary, distinct, and complex conjugated. The corresponding PDE is then *elliptic*. The classic example is Poisson's equation,

$$\nabla_H^2 \phi = \partial_x^2 \phi + \partial_y^2 \phi = g(x, y), \tag{1.22}$$

where again ∇_H is the two-dimensional part of the three-dimensional del operator. We arrive at this equation by letting $\xi = x$ and $\eta = y$ in (1.15), and by letting $a = c = 1$ and $b = d = e = f = 0$. Other examples are the Helmholtz equation and the Laplace equation, that is,

$$\nabla_H^2 \phi + f(x, y)\phi = g(x, y), \tag{1.23}$$

$$\nabla_H^2 \phi = 0, \tag{1.24}$$

respectively.

1.6 The shallow water equations

A very common set of equations in meteorology and oceanography are the so called shallow water equations. We may derive these equations from the governing equation (1.1) - (1.4) by making some assumptions. First we assume that the density ρ is uniform in time and space. Next we assume that the hydrostatic approximation is valid, and then we integrate the equations from

bottom to top. Then letting the depth of a fluid column be denoted h the governing equations become

$$\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla_H \mathbf{u} + f\mathbf{k} \times \mathbf{u} = -g\nabla_H h, \tag{1.25}$$

$$\partial_t h + \nabla_H \cdot (h\mathbf{u}) = 0, \tag{1.26}$$

which is commonly referred to as the *shallow water equations*. Here u, v are the two components of the horizontal velocity vector $\mathbf{u}$ along, respectively, the x and y axis, $f = 2\Omega \sin\phi$ is the Coriolis parameter where Ω is the Earth's rotation rate and ϕ is latitude, and g is the gravitational acceleration. We may linearize these equations by assuming the the deviation of height of a fluid column is small compared to its equilibrium depth H, that is, $(h - H)/H \ll 1$. Then follows

$$\partial_t u - fv = -g\partial_x h, \tag{1.27}$$

$$\partial_t v + fu = -g\partial_y h, \tag{1.28}$$

$$\partial_t h + H\partial_x u + H\partial_y v = 0. \tag{1.29}$$

By first using (1.27) and (1.28) to find u, v as functions of h, and then substituting the results into (1.29) we obtain

$$(\partial_t^2 + f^2 - gH\nabla_H^2)\partial_t h = 0. \tag{1.30}$$

Integration in time t then yields

$$(\partial_t^2 + f^2 - gH\nabla_H^2)h' = 0. \tag{1.31}$$

Here we have used $h = H + h'$ and have assumed that $h' = 0$ at time $t = 0$. If we in addition assume that the motion is independent of one of the dependent variableses, say y, we get

$$(\partial_t^2 + f^2 - gH\partial_x^2)h' = 0. \tag{1.32}$$

We note that (1.31) is hyperbolic in t, x (and in t, y), but elliptic in x, y. Thus, we note that although the steady state solution to (1.32) is elliptic, the time marching problem is inherently hyperbolic.

The governing equations that describe the time evolution of atmosperic and oceanographic motion is fundamentally hyperbolic. It is important that we keep this in mind when developing numerical methods to solve atmospheric and oceanographic problems.

1.7 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 set (1.27) - (1.29) (or eq. 1.30) 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. This is where the *boundary conditions* comes in. 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 we overspecify the system, and if we specify too few results in 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 wrong. 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 (1.22) we need four boundary conditions, two in x and two in y. To determine the solution to the diffusion equation (1.20) 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

1. Dirichlet conditions
2. Neumann conditions.

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 impermable wall. Formulated mathematically this condition imples that

$$\mathbf{n} \cdot \mathbf{v} = 0 \tag{1.33}$$

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

$$\mathbf{n} \cdot \mathbf{F}_\theta = 0, \tag{1.34}$$

where $\mathbf{F}_\theta = -\kappa \nabla \theta$ is the diffusive heat flux vector. Thus (1.34) is the same as specifying the *gradient* (in this case a zero gradient) at the boundary, which is the classic example of a Neumann 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

$$\nu \partial_z \mathbf{u} = C_D \mathbf{u} \quad ; \quad z = -H, \tag{1.35}$$

where ν 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 bounary 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

$$C(x,t) = C(x+X,t), \tag{1.36}$$

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.

1.8 Taylor expansion

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 are continuous[6]. 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 ininite sum of orthonogal functions such as for instance trigonometric function (Section 1.11).

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

$$\theta(x+\Delta x,t) = \theta(x,t) + \partial_x\theta(x,t)\Delta x + \frac{1}{2}\partial_x^2\theta(x,t)\Delta x^2 + \frac{1}{6}\partial_x^3\theta(x,t)\Delta x^3 + \mathcal{O}(\Delta x^4) \tag{1.37}$$

$$\theta(x-\Delta x,t) = \theta(x,t) - \partial_x\theta(x,t)\Delta x + \frac{1}{2}\partial_x^2\theta(x,t)\Delta x^2 - \frac{1}{6}\partial_x^3\theta(x,t)\Delta x^3 + \mathcal{O}(\Delta x^4) \tag{1.38}$$

By subtracting (1.38) from (1.37), and then do some suitable manipulation the first derivative of θ at the point (x,t) in time and space may be written

$$\partial_x\theta(x,t) = \frac{\theta(x+\Delta x,t) - \theta(x-\Delta x,t)}{2\Delta x} + \mathcal{O}(\Delta x^2) \tag{1.39}$$

We may also choose to solve (1.37) directly with respect to the first derivative. Then we obtain

$$\partial_x\theta(x,t) = \frac{\theta(x+\Delta x,t) - \theta(x,t)}{\Delta x} + \mathcal{O}(\Delta x) \tag{1.40}$$

Expression (1.39) and expression (1.40) above may also be used to formulate possible *finite difference approximations* of the first derivative of θ with respect to x, that is,

$$[\partial_x\theta]_{x,t} \approx \frac{\theta(x+\Delta x,t) - \theta(x-\Delta x,t)}{2\Delta x} \tag{1.41}$$

[6]This definition is somewhat different from the one offered in the little known but enlightning book by M. J Lighthill entitled "Good functions"

$$[\partial_x \theta]_{x,t} \approx \frac{\theta(x+\Delta x,t)-\theta(x,t)}{\Delta x} \tag{1.42}$$

Instead of (1.37) we may also use (1.38) to formulate an approximation, that is,

$$[\partial_x \theta]_{x,t} \approx \frac{\theta(x,t)-\theta(x-\Delta x,t)}{\Delta x} \tag{1.43}$$

We note that while (1.41) is centered on the spatial point x (1.42) and (1.43) are one-sided. The approximation (1.41) is therefore denoted a *centered approximation*, while (1.42) and (1.43) 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 θ in time we get

$$\theta(x,t+\Delta t) = \theta(x,t) + \partial_x\theta(x,t)\Delta t + \frac{1}{2}\partial_t^2\theta(x,t)\Delta t^2 + \frac{1}{6}\partial_t^3\theta(x,t)\Delta t^3 + \mathcal{O}(\Delta t^4), \tag{1.44}$$

$$\theta(x,t-\Delta t) = \theta(x,t) - \partial_x\theta(x,t)\Delta t + \frac{1}{2}\partial_t^2\theta(x,t)\Delta t^2 - \frac{1}{6}\partial_t^3\theta(x,t)\Delta t^3 + \mathcal{O}(\Delta t^4). \tag{1.45}$$

To construct a centered finite difference approximation to the time rate of change of θ we simply subtract (1.45) from (1.44) to obtain

$$[\partial_t \theta]_{x,t} \approx \frac{\theta(x,t+\Delta t)-\theta(x,t-\Delta t)}{2\Delta t} \tag{1.46}$$

Similarly we may construct approximations to higher order derivatives. For instance to find a centered finite difference approximation to the second order derivative of θ with respect to x we first simply add the two Taylor expansion (1.37) and (1.38) to give

$$\partial_x^2\theta(x,t) = \frac{\theta(x+\Delta x,t)-2\theta(x,t)+\theta(x-\Delta x,t)}{\Delta x^2} + \mathcal{O}(\Delta x^2). \tag{1.47}$$

Then by neglecting terms of higher order in (1.47) a finite difference approximation to the second order derivative is

$$[\partial_x^2 \theta]_{x,t} \approx \frac{\theta(x+\Delta x,t)-2\theta(x,t)+\theta(x-\Delta x,t)}{\Delta x^2}. \tag{1.48}$$

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 (1.41) we note that the neglected terms are of $\mathcal{O}\Delta x^2)$. This is in contrast to the forward and backward approximations in which the negelected terms where of $\mathcal{O}(\Delta x)$. Thus the centered approximation appears to share the fact that the neglected terms are of higher order than the one-sided approximations.

As displayed in (1.46) we may in a similar fashion formulate finite difference approximations to any higher order derivative with respect to t, x and other spatial independent variables. As is becoming more common in modern codes is also that we may formulate higher order approximations where the terms neglected in the Taylor series expansion are $\mathcal{O}(\Delta x^n)$ where $n \geq 3$.

1.9 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 $\mathcal{O}(\Delta x^2)$ when using the centered finite difference approximation, the terms we neglected when using the one-sided approximation was $\mathcal{O}(\Delta x)$. 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 approximation are usually called *higher order schemes* or higher order finite difference approximations. We note from (1.58) 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.

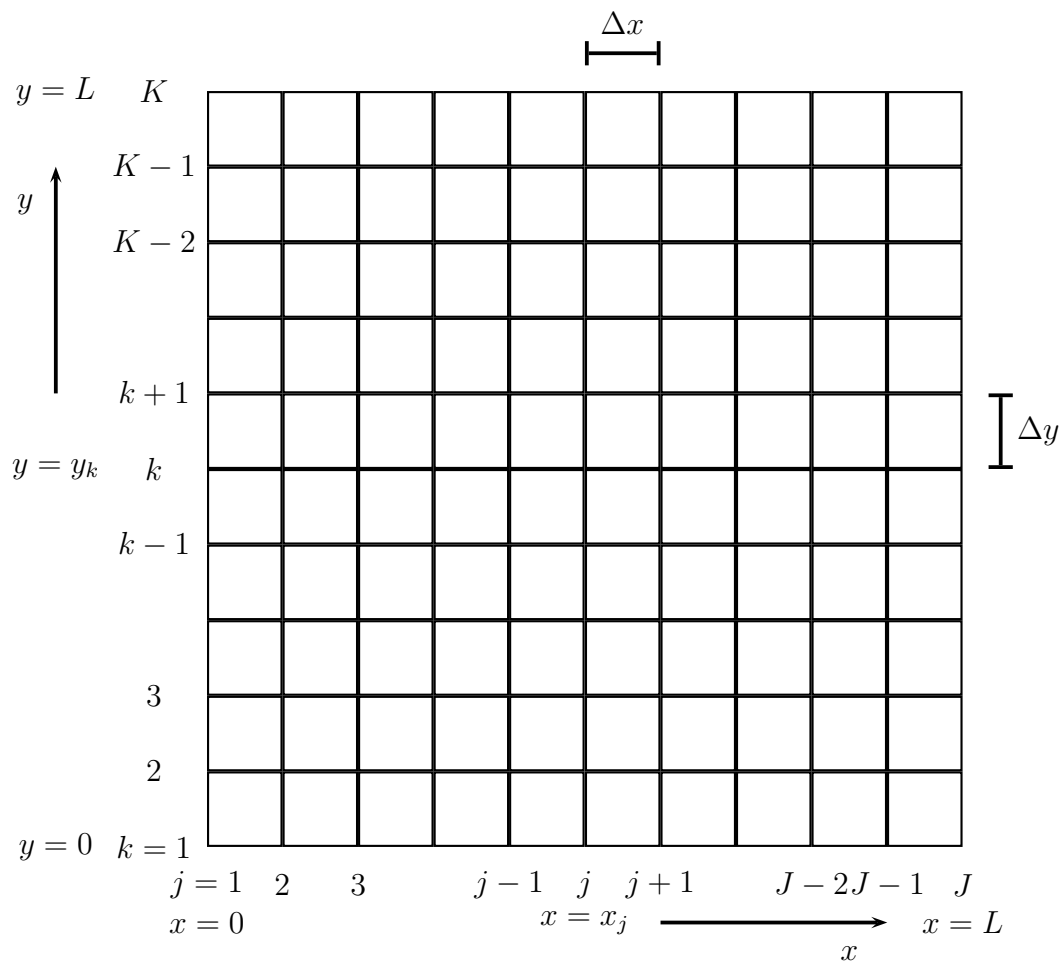
Finally, we emphasize that when the problem is multi-dimensional it is important that the finite difference approximation in all spatial directions are truncated to the same order. As an example consider a line wave propagating in space. The only way to ensure that the numerical solution then has the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

1.10 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 the Poisson problem and let us assume that we seek a solution within a quadratic domain where x, y both starts at 0 and ends at L. As is common we start by covering the domain by a quadratic mesh as displayed in Figure 1.1. We keep track of the grid points in the mesh by counting along the x-axis and the y-axis, respectively. 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 the $j = J$. Similarly is the point $y = 0$ associated with $k = 1$ and the point $y = L$ with $k = K$[7]. The jth point along the x-axis is then $x = x_j$ where the subscript refers to the value for x at the jth point

[7]FORTRAN 90/95 tillater også at dummytellerene j og k kan starte på 0. Dersom dettte brukes vil $x_j = j\Delta x$ og $y_k = k\Delta y$ slik at $x_0 = y_0 = 0$ mens $x_J = y_K = L$ som før. Derav følger at $\Delta x = L/J$, og $\Delta y = L/K$

along the x-axis. Similarly we let $y = y_k$ be associated with the kth point along the y-axis. The coordinates of the grid is then given by x_j, y_k.



Figur 1.1: 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.

The distances between the points along the x-axis and y-axis is Δx and Δy, respectively. Then the jth point along the x-axis and kth point along the y-axis are

$$x_j = (j-1)\Delta x, \quad y_k = (k-1)\Delta y, \tag{1.49}$$

respectively, from which follows $x_1 = y_1 = 0$ and $x_J = y_K = L$. Note that the latter gives

$$\Delta x = L/(J-1), \quad \Delta y = L/(K-1), \tag{1.50}$$

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

$$\theta_{jk} = \theta(x_j, y_k) = \theta[(j-1)\Delta x, (k-1)\Delta y]. \tag{1.51}$$

Furthermore follows that

$$\theta_{jk} = \theta(x_j, y_k) = \theta[(j-1)\Delta x, (k-1)\Delta y] \tag{1.52}$$

$$\theta_{j-1k} = \theta(x_{j-1}, y_k) = \theta(j-2)\Delta x, (k-1)\Delta y] \tag{1.53}$$

$$\theta_{jk+1} = \theta(x_j, y_{k+1}) = \theta[(j-1)\Delta x, k\Delta y] \tag{1.54}$$

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

$$t^n = (n-1)\Delta t, \quad n = 1, 2, \cdots \tag{1.55}$$

when counting time, where Δt 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

$$\theta_j^n = \theta(x_j, t^n) = \theta[(j-1)\Delta x, (n-1)\Delta t] \tag{1.56}$$

Finally we note that when the variable is four-dimensional the notion we use is

$$\theta_{jkl}^n = \theta(x_j, y_k, z_l, t^n) \tag{1.57}$$

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

1.11 Fourier series

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 $\mathcal{O}(\Delta x^4)$ is

$$[\partial_x \theta]_j \approx \frac{4}{3}\frac{\theta_{j+1} - \theta_{j-1}}{2\Delta x} - \frac{1}{3}\frac{\theta_{j+2} - \theta_{j-2}}{4\Delta x}. \tag{1.58}$$

   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.

Kapittel 2

TIME DEPENDENT PROBLEMS

2.1 The advection-diffusion equation

We will now focus on the tracer conservation equation in the form (1.14). Neglecting possible sources we may rewrite (1.14) to yield

$$\partial_t \theta + \nabla \cdot \mathbf{F}_A + \nabla \cdot \mathbf{F}_D = 0, \tag{2.1}$$

where θ is the dependent variable (or the state variable), for instance potential temperature, $\mathbf{F}_A$ is the advective flux vector to be associated with second term on the left-hand side of (1.14), and $\mathbf{F}_D$ is the flux due to the turbulent mixing commonly referred to as the diffusive flux vector, since the nature of the turbulence is to smooth out any differences in the tracer and is thus akin to a diffusive process. If θ is the potential temperature then (2.1) is the conservation equation for internal energy or heat content except for the neglect of source terms.

While the advecive flux vector is a mathematical formulation of those physical processes that is responsible for transporting the property of the tracer from one place to another via the motion, the diffusive, turbulent flux vector represents quite another physical process, namely one in which the property θ is transferred from one location to another by turbulent mixing, that is, small scale, inherently chaotic processes that causes properties to be exchanged between two locations without invoking any motion. Thus it is a process similar to conduction. Although their mathematical expression in (2.1) are similar, we therefore refer to the second term on the left-hand side of (2.1) as being the advective term, while we refer to the third term on the left-hand side as being the diffusive term.

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

$$\mathbf{F}_A = \mathbf{v}\theta, \tag{2.2}$$

as in (1.14), the formulation of the diffusive flux vector may take various forms. The most common one, called “down the gradient” diffusion, is

$$\mathbf{F}_D = -\kappa \nabla \theta, \tag{2.3}$$

where κ is the diffusion coefficient or conductive capacity[1] Equation (2.3) 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 θ over small distances.

If we for a moment neglect the diffusive part of (2.1) 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" (1.18). Solving this equation is often referred to a solving the *advection problem*. If we next consider a balance between the time rate of change of the tracer concentration and the diffusive part of (2.1), using the parametrization (2.3), we are left with a typical parabolic problem (cf. 1.20), and the resulting equation is called the diffusion equation. Solving the diffusion equation is often referred to as solving the *diffusion problem*.

In the following Chapters 3 and 4 we will give insight into solving respectively the diffusion problem and the advection problem by use of numerical methods. Later we will also give insight into how to solve numerically the combined advection-diffusion problem, in which both the advective part and the diffusive part are kept in (2.1). Both of these problems are central when solving the full governing equations (1.1) - (1.4). Thus we maintain that to know how to treat these terms correctly is of fundamental importance when solving any problem in meteorology and oceanography numerically. At the same time it conveniently gives us the opportunity to introduce basic concepts within numerical solution to atmospheric and oceanographic problems in a simple setting.

Before doing this we will first study some physical properties peculiar to each of the two processes. These are important properties that 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 a part of what is often referred to as the "debugging" process[2].

2.2 The diffusion problem

As alluded to above one property of the diffusive processes is that it tends to damp out differences in the fields. This is also true for unwanted noise, if any, in our numerical solutions. The most convenient way to show that the diffusion equation indeed has this property is to analyze the variance of the tracer concentration. The variance, which is the square of the tracer concentration itself, is a measure of how noisy a field is. Thus we first multiply the diffusion equation

$$\partial_t \theta = -\nabla \cdot \mathbf{F}_D \tag{2.4}$$

by the tracer concentration θ itself to obtain

$$\partial_t \theta^2 = -2\theta \nabla \cdot \mathbf{F}_D. \tag{2.5}$$

The left hand side of (2.5) is then the time rate of change of the variance at one point in time and space. Next we integrate over the total volume V for which we seek a solution to (2.4), and let

[1] Its original formulation is due to a Dr. Adolf Eugen Fick who in 1855 formulated the parametrization (2.3) which is now referred to as Fickian diffusion.

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

Ω denote the bounding surface of that volume. Then the time rate of change of the total variance within the volume is

$$\partial_t \int_V \theta^2 dV = -2 \int_\Omega \theta \mathbf{F}_D \cdot \delta \boldsymbol{\sigma} + 2 \int_V \mathbf{F}_D \cdot \nabla \theta dV. \tag{2.6}$$

Here the vector $\delta \boldsymbol{\sigma} = \mathbf{n} \delta \sigma$ wherer $\mathbf{n}$ is a unit vector directed along the outward normal to the surface Ω and $\delta \sigma$ is an infinitesimal surface element. Let us now assume that the at the boundary Ω either the Dirichlet condition $\theta = 0$ or the Neuman condition $\mathbf{n} \cdot \mathbf{F}_D = 0$ prevails. Then the first term on the right-hand side of (2.6) is zero, and (2.6) becomes

$$\partial_t \int_V \theta^2 dV = 2 \int_V \mathbf{F}_D \cdot \nabla \theta dV. \tag{2.7}$$

Thus if $\mathbf{F}_D \cdot \nabla \theta \leq 0$ then the right-hand side of (2.7) is negative and hence that

$$\partial_t \int_\tau \theta^2 d\tau \leq 0. \tag{2.8}$$

Thus if the flux vector $\mathbf{F}_D$ is down the gradient in the sense that it is directed opposite to $\nabla \theta$ then it always tends to decrease the variance. Recalling from (2.3) that in our case with Fickian diffusion, that is, $\mathbf{F}_D = -\kappa \nabla \theta$ we indeed obtain

$$\mathbf{F}_D \cdot \nabla \theta = -\kappa (\nabla \theta)^2 \leq 0. \tag{2.9}$$

Thus Fickian diffusion is indeed down the gradient and will always damp 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 3.2) 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[3]:

> *"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."*

[3]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).

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 5.2).

2.3 The advection problem

As alluded to in Chapter 2.1 is the advection problem a balance between the first and second term on the left hand-side of (2.1), that is,

$$\partial_t \theta + \nabla \cdot \mathbf{F}_A = 0, \tag{2.10}$$

where θ is any state variable and $\mathbf{F}_A$ is the adcevtive flux vector. As for the diffusion problem we are looking for solutions within a limited volume V in space and for all times from $t = 0$ to $t = \infty$[4]. On the boundary of the volume V, given by the surface Ω, the equations are replaced by the boundary conditions, while the initial condition replaces the equations at time $t = 0$.

Let the advective flux be parameterized by the common parameterization $\mathbf{F}_A = \mathbf{u}\theta$, and let the boundary condition at the surface Ω be such that $\mathbf{F}_A \cdot \delta\boldsymbol{\sigma} = 0$[5]. Then by performing the same operation as in Section 2.2 we find that the total variance becomes

$$\partial_t \int_V \theta^2 dV = 2\int_V \mathbf{F}_A \cdot \nabla\theta dV = \int_V \mathbf{u} \cdot \nabla\theta^2 dV = -\int_V \theta^2 \nabla \cdot \mathbf{u} dV. \tag{2.11}$$

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 $\nabla_H \cdot \mathbf{u} = 0$ is special. In this case the right han-side of (2.11) is zero and hence any disturbances creating a variance in θ will just prevail, that is, the total variance is conserved.

As mentioned in Section 1.3 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 ($\nabla_H \cdot \mathbf{u} > 0$), 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 $\nabla_H \cdot \mathbf{u} < 0$.

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

[4] In practice we have to limit the computation to a finite time span

[5] This is achieved by assuming $\mathbf{u} = 0$ or $\mathbf{u} \cdot \delta\boldsymbol{\sigma} = 0$, that is, no flow across the boundary.

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 te diffusion problem where all down the gradient diffusive fluxes give a decrease in the total tracer variance.

Kapittel 3

THE DIFFUSION PROBLEM

3.1 Finite difference approximations

To make the diffusion problem attractive without loosing its essence, we simply assume that our problem is one-dimensional in space. When formulated in analytic or continuous form the one-dimensional diffusion equation is given by (1.20), that is,

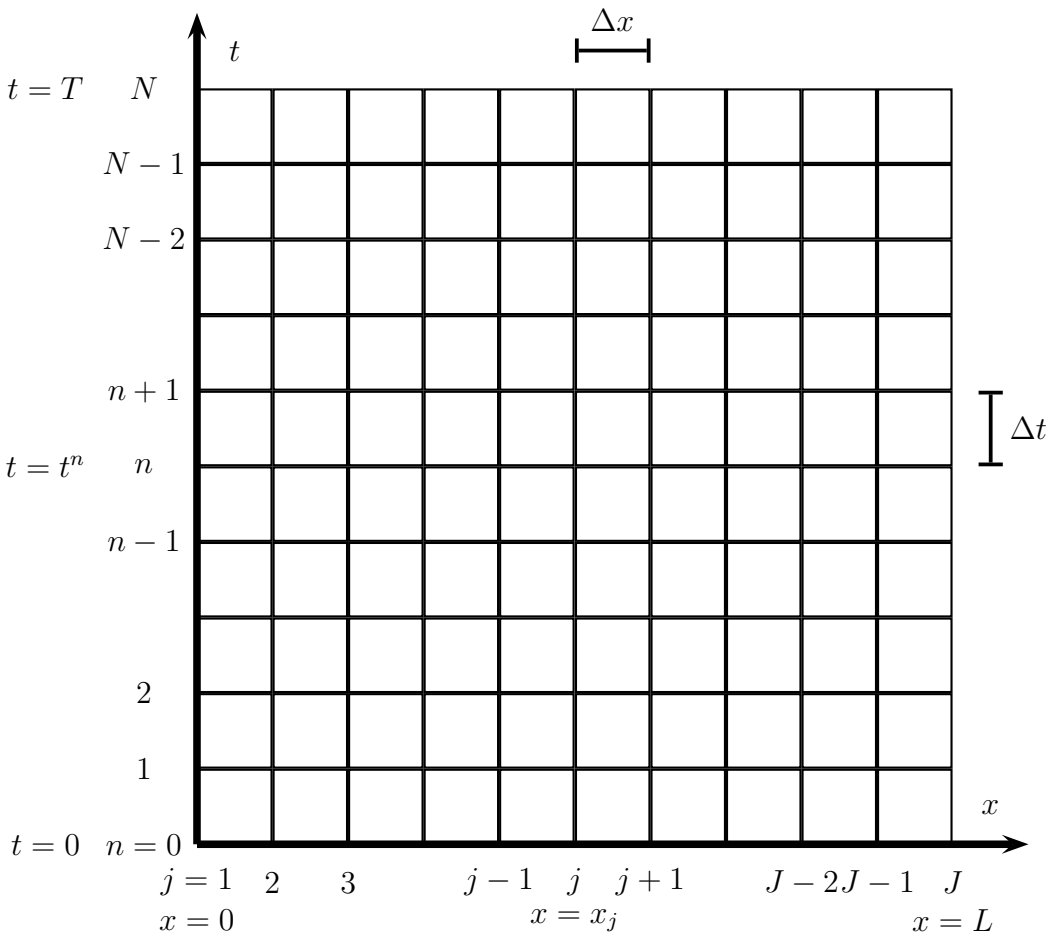
$$\partial_t \theta = \kappa \partial_x^2 \theta, \tag{3.1}$$

where θ can be any variable, say potential temperature, density, velocity components, etc. As alluded to in Chapter 1.4, we note that this equation is parabolic in nature. Thus the physical characteristic of the problem is that properties (or energy) are transferred from one location to the next by conduction. We emphasize that this is very different from hyperbolic type problems, like the advection problem, which physical characteristic is that properties are transferred from one location to the next by propagating waves.

An obvious example is heat conduction in both atmospheres and oceans. Then θ 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.

Our concern is to find a numerical solution to (3.1) for all times and for a given computational domain or area, say for instance $x \in \langle 0, L \rangle$, with given boundary conditions for θ at $x = 0$ and $x = L$ and for a given initial condition at $t = 0$. Let us for instance consider heat conduction in the atmosphere and ocean. Let x be the height (or depth) coordinate, and let θ describe the potential temperature anomaly away from a given mean temperature profile. We then assume that we know the anomaly at time $t = 0$. Our task is then to find, by numerically solving (3.1), how the anomaly profile evolves in time between the two heights (or depths) $x = 0$ and $x = L$ given how the anomaly evolves at these two heights. By considering that $\theta = 0$ at $x = 0$ and $x = L$ we imply that the boundary condition is a Dirichlet condtion, that is, that the value of the dependent variable is fixed for all times. We also assume that the initial profile is different from the trivial solution $\theta = 0$, that is, we assume that $\theta = \theta_0(x)$ at time $t = 0$.

To find a numerical solution to (3.1) we follow the notation in Section 1.10. Thus we first divide the intervals $x \in \langle 0, L\rangle$ and $t \in \langle 0, T\rangle$, where T is some finite time, into respectively $J-1$ and $N-1$ sections of width Δx and Δt. 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 $j \in [1, J]$ and $n \in [0, N]$ are counters such that $x_J = L$ and $t^N = T$ (see Figure 3.1).



Figur 3.1: Displayed is the employed grid we use to solve (3.1) by numerical means. The grid points in the x, t directions are incremented by $\Delta x, \Delta t$, respectively, so that there are a total of J points along the x-axis and N points along the t-axis. The points are counted by using the dummy counters j, n.

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 1.8 that

$$[\partial_t\theta]_j^n = \frac{\theta_j^{n+1}-\theta_j^n}{\Delta t} + \mathcal{O}(\Delta t), \quad [\partial_x^2\theta]_j^n = \frac{\theta_{j+1}^n - 2\theta_j^n + \theta_{j-1}^n}{\Delta x^2} + \mathcal{O}(\Delta x^2). \tag{3.2}$$

Thus we write the finite difference approximation of (3.1) as

$$\frac{\theta_j^{n+1}-\theta_j^n}{\Delta t} = \kappa\frac{\theta_{j-1}^n - 2\theta_j^n + \theta_{j+1}^n}{\Delta x^2}, \tag{3.3}$$

which when solved with respect to θ_j^{n+1} gives

$$\theta_j^{n+1} = \theta_j^n + \frac{\kappa\Delta t}{\Delta x^2}\left(\theta_{j-1}^n - 2\theta_j^n + \theta_{j+1}^n\right). \tag{3.4}$$

Note that (3.4) is only valid for $j = 2(1)J-1$ and for $n = 0(1)N$. At the boundaries $j = 1$ and $j = J$ and for $n = 0$ the boundary and initial conditions prevail.

To find θ at the first time level $n = 1$, that is, θ_j^1, we simply substitute $n = 0$ into (3.4) which gives

$$\theta_j^1 = \theta_j^0 + \frac{\kappa\Delta t}{\Delta x^2}\left(\theta_{j-1}^0 - 2\theta_j^0 + \theta_{j+1}^0\right). \tag{3.5}$$

Since all the θ_j^0s are known from the initial condition, that is, $\theta_j^0 = \theta(x_j, 0)$ for all $x = x_j$ where $j = 1(1)J$, we may proceed to calculate θ at the first time level for all $j = 2(1)J-1$. Let us start with $j = 2$. Then (3.5) gives

$$\theta_2^1 = \theta_2^0 + \frac{\kappa\Delta t}{\Delta x^2}\left(\theta_1^0 - 2\theta_2^0 + \theta_3^0\right). \tag{3.6}$$

We may then proceed to evaluate $\theta_3^1, \theta_4^1, \cdots, \theta_{J-1}^1$ using (3.5). Note that we end with $j = J-1$. This procedure then gives us θ at all the grid points except at the two boundaries $j = 1$ and $j = J$, that is, we miss θ_1^1 and θ_J^1. These are however given by the boundary conditions and need not be evaluated. This reflects the well known analytic property of differential equations, namely yhat they are valid only in the interior of the domain. At the boundaries (whether in time or space) the equations are replaced by the boundary condition. Thus (3.5) together with the boundary conditions gives us θ for all $j = 1(1)J$ at time level $n = 1$. We may then proceed to compute θ at time level $n = 2$ by substituing $n = 1$ into (3.4), that is,

$$\theta_j^2 = \theta_j^1 + \frac{\kappa\Delta t}{\Delta x^2}\left(\theta_{j-1}^1 - 2\theta_j^1 + \theta_{j+1}^1\right), \tag{3.7}$$

and so on for all time levels n up to and including $n = N$.

We note that since $x_J = L = (J-1)\Delta x$ we cannot choose J, L and Δx independently. Once two of them are chosen the third is given by the formula

$$J = 1 + \frac{L}{\Delta x}. \tag{3.8}$$

Likewise follows that

$$N = 1 + \frac{T}{\Delta t} \tag{3.9}$$

showing that N, Δt and T also depend on each other.

The use of a forward, one-sided approximation in time, as used in (3.4), is the natural choice since we know θ at time $t = 0$ (or time level $n = 0$). It has, however, a low accuracy in that the truncation error is $\mathcal{O}(\Delta t)$. To increase the accuracy we may for instance use a centered in time scheme for the time derivative, that is,

$$[\partial_t \theta]_j^n = \frac{\theta_j^{n+1} - 2\theta_j^n + \theta_j^{n-1}}{\Delta t^2} + \mathcal{O}(\Delta t^2), \tag{3.10}$$

in which case the finite difference approximation to (3.1) becomes,

$$\frac{\theta_j^{n+1} - \theta_j^{n-1}}{2\Delta t} = \kappa \frac{\theta_{j-1}^n - 2\theta_j^n + \theta_{j+1}^n}{\Delta x^2}, \quad j = 2(1)J-1, n = 0(1)N, \tag{3.11}$$

or

$$\theta_j^{n+1} = \theta_j^{n-1} + \kappa \frac{2\Delta t}{\Delta x^2} \left(\theta_{j-1}^n - 2\theta_j^n + \theta_{j+1}^n \right), \tag{3.12}$$

for $j = 2(1)J-1$ and $n = 0(1)N$. To obtain the solution at the first time level $n = 1$, that is, to obtain θ_j^1 we again substitute $n = 0$ into (3.12) which gives

$$\theta_j^1 = \theta_j^{-1} + \kappa \frac{2\Delta t}{\Delta x^2} \left(\theta_{j-1}^0 - 2\theta_j^0 + \theta_{j+1}^0 \right). \tag{3.13}$$

Thus we need to know θ_{j-1}^0, which corresponds to knowing the potential temperature profile at a time $t < 0$, that is, one time level *below* the first time level. at Dette krever imidlertid mer informasjon om initialbetingelsen. By using the one-sided forward scheme we avoid this problem.

3.2 Numerical stability

Another major problem with the scheme (3.12) is that it is *numerically unstable*. This means that the numerical solution, instead of following the analytic solution, steadily deviates from the analytic solion. 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 intabilities that we would actually like to simulate using our numerical model. For our numerical solution to have any legitimacy it must be 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 analyse the numerical scheme with respect to its numerical stability, let us first consider the analytic solution to (3.1). To this end we note that any good function θ may be written as a sum of cosines and sines or even more compact as a sum of exponentials (see for instance *Lighthill*, 1970, page 3)

$$\theta(x,t) = \sum_{\alpha=-\infty}^{\infty} \Theta_\alpha(t) e^{i\alpha x} = \sum_{\alpha=-\infty}^{\infty} \theta_\alpha \tag{3.14}$$

where α is the wave number. Each component in (3.14), that is,

$$\theta_\alpha = \Theta(t) e^{i\alpha x}, \tag{3.15}$$

is called a Fourier component, where $\Theta(t)$ is an unknown time dependent amplitude. By substituting this component into (3.1) we obtain

$$\partial_t \Theta = -\kappa\alpha^2 \Theta. \tag{3.16}$$

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

$$\Theta = \Theta_0 e^{-\kappa\alpha^2 t} \tag{3.17}$$

where Θ_0 is the initial amplitude at time $t = 0$. Thus we find that the analytic solution to (3.1) is

$$\theta(x,t) = \sum_{-\infty}^{\infty} \Theta_0 e^{-\kappa\alpha^2 t} e^{i\alpha x}. \tag{3.18}$$

We note that the amplitude of all the Fourier components decreases monotonically and exponentially as time increases. Moreover, we note that that part of the solution associated with the shortest waves (highest wave numbers) decreases faster than that part of the solution associated with the longer waves (low wave numbers). This is in accord with Section 2.2 where we concluded, based on (2.8), that diffusion acts to smooth out disturbances. What we additional learn from (3.18) 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 3.3 below.

In mathemathical 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

$$\left| \frac{\theta}{\theta_0} \right| \leq B, \tag{3.19}$$

where θ_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.

3.3 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 (3.15), that is,

$$\theta_j^n = \Theta_n e^{i\alpha j\Delta x}, \tag{3.20}$$

where Θ_n is the discrete amplitude at time level n, that is, at time $t = n\Delta t$, and α is the wave-number as above. The next step is to define the the *growth factor* G as the amplification of the amplitude Θ as we proceed from one time level to the next, that is,

$$G \equiv \frac{\Theta_{n+1}}{\Theta_n} \quad \Rightarrow \quad \Theta_{n+1} = G\Theta_n \quad \text{and} \quad \Theta_{n-1} = G^{-1}\Theta_n. \tag{3.21}$$

We observe that (3.21) is similar to (3.19), 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 (3.19) is the ratio between the value at a random time level and the initial value. Letting $n = 0$ in (3.21) then gives

$$\Theta_1 = G\Theta_0 \tag{3.22}$$

where Θ_0 is the initial amplitude. By letting $n = 1$ in (3.21) and making use of (3.22) we obtain

$$\Theta_2 = G\Theta_1 = G^2\Theta_0 \tag{3.23}$$

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

$$\Theta_m = G^m\Theta_0 \tag{3.24}$$

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 (3.19) is satisfied if

$$|G| \leq 1 \tag{3.25}$$

since then G^m decreases as the time level or time increases[1]. The criterion (3.25) is called *von Neumann's condition for stability*. Note that it is a sufficient condition, a fact that we will come back to shortly (Section 3.4).

As an example we will use von Neumanns method to analyse the forward in time, centered in space scheme for the diffusion as given by (3.4). Thus we first substitute (3.20) into (3.4) to obtain

$$\Theta_{n+1} = \Theta_n + \frac{\kappa\Delta t}{\Delta x^2}\left(e^{i\alpha\Delta x} - 2 + e^{-i\alpha\Delta x}\right)\Theta_n \tag{3.26}$$

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

$$\Theta_{n+1} = \left[1 - 2\frac{\kappa\Delta t}{\Delta x^2}(1 - \cos\alpha\Delta x)\right]\Theta_n. \tag{3.27}$$

[1] Confer Computer Problem No. 1

We thus find the growth factor by simply dividing (3.27) by Θ_n, that is, the amplitude at time level n, which gives

$$G = 1 - 2\frac{\kappa\Delta t}{\Delta x^2}(1 - \cos\alpha\Delta x). \tag{3.28}$$

To satisfy (3.25) we then observe that

$$-1 \leq 1 - 2\frac{\kappa\Delta t}{\Delta x^2}(1 - \cos\alpha\Delta x) \leq 1. \tag{3.29}$$

Since $0 \leq (1 - \cos\alpha\Delta x) \leq 2$ follows that the right-hand side unequality is always satisfied for all Δt and Δx. The unequality on the left-hand side, however, is satisfied if and only if

$$\frac{\kappa\Delta t}{\Delta x^2}(1 - \cos\alpha\Delta x) \leq 1. \tag{3.30}$$

Recall that $0 \leq (1 - \cos\alpha\Delta x) \leq 2$, and hence if

$$\frac{\kappa\Delta t}{\Delta x^2} \leq \frac{1}{2} \quad \text{eller} \quad \Delta t \leq \frac{\Delta x^2}{2\kappa}. \tag{3.31}$$

then follows that (3.30) is satisfied for all wavenumbers α. This condition also ensures that (3.29) is satisfied, and hence that von Neumann's condition (3.25) is satisfied as well. Furthermore (3.31) tells us that we cannot choose Δx and Δt independently. Once Δx is chosen the time step Δt must be chosen in accord with (3.31). We therefore say that the forward in time, centered in space scheme (3.4) is *conditionally stable* under the condition (3.31).

We also observe from (3.30) that the waves that first violate the unequality are waves with wavenumbers given by

$$\cos\alpha\Delta x = -1 \tag{3.32}$$

corresponding to those waves that maximizes $1 - \cos\alpha\Delta x$. The solutions to (3.32) are wavenumbers α_m given by

$$\alpha_m\Delta x = (2m - 1)\pi; \quad m = 1, 2, \ldots. \tag{3.33}$$

with corresponding wavelengths

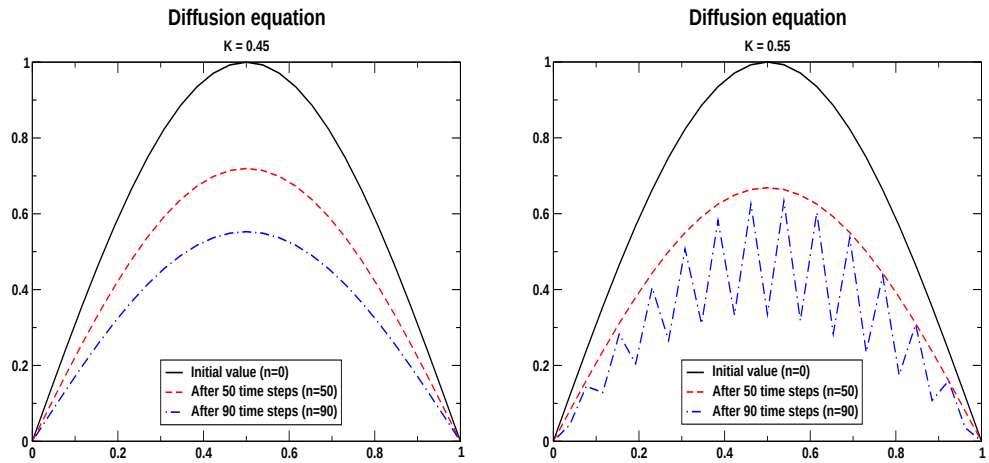
$$\lambda_m = \frac{2\pi}{\alpha_m} = \frac{2\Delta x}{2m - 1}. \tag{3.34}$$

The most dominant of these waves is the wave corresponding to $m = 1$. Thus the most unstable wave has wavelength

$$\lambda_1 = 2\Delta x. \tag{3.35}$$

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 3.2. (cf. Computer Problem No. 2).

We mentioned briefly that the we call the forward in time, centered in space scheme (3.4) for the diffusion equation a conditionally stable scheme. If no such condition can be found which satisfies the von Neumann condition (3.25) then we say that the scheme is *unconditionally unstable*.



Figur 3.2: Displayed are solutions of the diffusion eqaution using the scheme (3.4) 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 (3.31) is violated for $K = 0.55$, but not for $K = 0.45$.

If von Neumann's condition is satisfied whatever is chosen regarding the time step Δt and grid size Δx we say that the scheme is *unconditionally stable*. If the special case $|G| = 1$ is true then we in addition say that the scheme is *neutrally stable*.

It is worthwhile mentioning that when $|G| < 1$ it follows from (3.21) 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 dissipiation depends on the scheme and not the least the absolute value of the growth factor. It is therefore of importance to ensure that the numerical dissipation is as small as possible. For physical problems that do include energy dissipation it is important to ensure that the numerical dissipation is small with respect to the physical dissipation. We therefore always favor neutral schemes 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 increment so as to minimize the numerical energy dissipation. This is the same as ensuring that G is 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 Δt such that $\Delta t \lesssim \Delta x^2/2\kappa$.

3.4 The necessary stability condition

We mentioned above that von Neumann's condition is a sufficient condition. This entails that if (3.25) is satisfied then the scheme is definitively stable. The question is if its too strict, that is, if

[2]In this context energy dissipation means that the amplitude of the solution decreases in time

it is also the *necessary condition*?

To determine this we return the original requirement as formulated in (3.19). Substituting the discrete Fourier component (3.20) into (3.19) then gives

$$|\frac{\Theta_{n+1}}{\Theta_0}| \leq B \quad \Rightarrow |G|^n \leq B. \tag{3.36}$$

Taking the natural logarithmic on both sides then gives

$$n \ln |G| \leq \ln B \equiv B'. \tag{3.37}$$

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 ϵ 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 (3.37) we obtain

$$\epsilon \leq \frac{B'\Delta t}{t^n} = O(\Delta t). \tag{3.38}$$

Thus the necessary condition that satisfies the numerical stability requirement is

$$|G| \leq 1 + O(\Delta t), \tag{3.39}$$

which shows that the von Neumann condition (3.25) is indeed too strict. However, most physical problems, even those containing physical instabilites, always involves 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.

For the one-dimensional diffusion equation the growth factor G, as displayed by (3.28) is a scalar. For multivariable and multi-dimensional problems the growth factor will commonly be a tensor or matrix $\boldsymbol{\mathcal{G}}$. The sufficient condition is then that its spectral radius is less than or equla to one. This is tantamount to require that the largest eigenvalue of the matrix $\boldsymbol{\mathcal{G}}$ is less than or equal to one.

3.5 Explicit and implicit schemes

The schemes (3.4) and (3.12) both contains 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 multivariable 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 (3.4) and (3.12), 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 (3.1). 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 (3.12). 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

$$\theta_j^{n+1} = \theta_j^{n-1} + \kappa \frac{2\Delta t}{\Delta x^2}\left(\theta_{j-1}^{n+1} - 2\theta_j^{n+1} + \theta_{j+1}^{n+1}\right), \quad \begin{cases} j = 2(1)J-1 \\ n = 0(1)\ldots \end{cases}. \tag{3.40}$$

Let us first analyze the stability of the explicit scheme using von Neumann's method. Substitution of the Fourier component (3.20) into (3.12), removing the common factor $e^{i\alpha j\Delta x}$ and using (3.21) then gives

$$G = G^{-1} - \kappa \frac{4\Delta t}{\Delta x^2}(1 - \cos\alpha\Delta x). \tag{3.41}$$

Multiplying by G and rearranging terms yields the equation

$$G^2 + 2\lambda G - 1 = 0 \quad \lambda = \kappa \frac{2\Delta t}{\Delta x^2}(1 - \cos\alpha\Delta x) \geq 0 \tag{3.42}$$

to determine the growth factor. It has the two solutions

$$G_{1,2} = -\lambda \pm \sqrt{1+\lambda^2}. \tag{3.43}$$

We note that in order to be stable both solutions must satisfy von Neumann's condition. We observe that

$$|G_2| = \lambda + \sqrt{1+\lambda^2} \geq 1, \tag{3.44}$$

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 diffusin problem. It is always unconditionally unstable.

Then what about the implicit scheme (3.40)? To analyse the its stability we first rearrange the terms in (3.40) to give

$$\theta_{j-1}^{n+1} - \left(2 + \frac{\Delta x^2}{2\kappa\Delta t}\right)\theta_j^{n+1} + \theta_{j+1}^{n+1} = -\frac{\Delta x^2}{2\kappa\Delta t}\theta_j^{n-1}, \quad \begin{cases} j = 2(1)J-1 \\ n = 0(1)\ldots \end{cases}. \tag{3.45}$$

We note in passing that the implicit formulation require us to solve for θ at time level $n+1$ at the three grid points $j-1$, j, and $j+1$ simultanously. It may however be efficiently solved employing an elliptic solver as shown below in Section 3.8. We also note in passing that, curiously enough, that the implicit formulation of the analytic parabolic diffusion equation turns it into an ellitic numerical equation.

To determine the growth factor we again employ von Neumann's method and substitute the Fourier component into (3.45) to give

$$G\left[2\cos\alpha\Delta x-(2+\frac{\Delta x^2}{2\kappa\Delta t})\right]=-\frac{\Delta x^2}{2\kappa\Delta t}G^{-1}. \tag{3.46}$$

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

$$|G_1|=|G_2|=\frac{1}{\sqrt{1+\frac{4\kappa\Delta t}{\Delta x^2}(1-\cos\alpha\Delta x)}} \tag{3.47}$$

Note that $|G_{1,2}|\leq 1$ for all Δt and Δx. Hence the implicit formulation of the centered in time centered in space scheme is unconditionally *stable*. This a property 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 (3.40) does not contain any constraint on the time step Δt, and we may choose Δt 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 growth factor decreases with increasing time step. Thus the implicit formulation always contains numerical dissipitation, and that this unphsyical dissipation increases with increasing time steps. Again this is a property shared by all implicit schemes. Although a long time step does not influence the stability of the implicit scheme it is nevertheless adviceable to choose a time step that brings the growth factor as close to one as possible to minimize the numerical dissipation inherent in all implicit schemes.

3.6 Consistent schemes: DuFort-Frankel

One important aspect about the finite difference approximations to our governing equations is that they should mimic their analytic countepart. We therefore require that they should converge toward the analytic equations in the limit when the time step Δt and the spatial increments Δx as well as Δy and Δz go to zero. 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 1.8 satisfies the consistency requirement. Since both of the schemes (3.4) and (3.12) are based on a Taylor series expansion, they are both prime examples of consistent schemes. Despite of this it is possible to construct numerical schemes without using Taylor series expansions. One example elaborated below is the Dufort-Frankel scheme. It is especially in cases when the scheme is not constructed based on Taylor series expansions that we need to analyse its consistency. If the scheme turns out to be inconsisten we require as a minimum that the dominant physical processes that the governing equations simulates are consistently represented in the finite difference analogue.

The Dufort-Frankel scheme is one such scheme. To construct the scheme we start with the consistent centered in time, centered in space scheme (3.12). As shown above (Section 3.5) this scheme is unconditionally unstable when applied to the diffusion equation, and hence useless. However, in many cases the diffusion term is added to the equations in order to smooth 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 our consistence requirement. To construct the Dufort-Frankel scheme we observe that the value of θ at the grid point j, n in space and time can be thought of as a linear interpolation in time of the two adjacent grid points $j, n+1$ and $j, n-1$, or

$$\theta_j^n = \frac{1}{2}\left(\theta_j^{n+1} + \theta_j^{n-1}\right). \tag{3.48}$$

Using this as a substitute for θ_j^n on the right-hand side of (3.12) we obtain

$$\theta_j^{n+1} = \theta_j^{n-1} + 2\frac{\kappa\Delta t}{\Delta x^2}\left(\theta_{j-1}^n - \theta_j^{n-1} - \theta_j^{n+1} + \theta_{j+1}^n\right), \quad \begin{cases} j = 2(1)J-1 \\ n = 0(1)N \end{cases}. \tag{3.49}$$

We first note that the introduction of the term θ_j^{n+1} makes the scheme implicit. As all implicit scheme we therefore expect the scheme to be stable. Curiously enough, by simply adding some implicity to the centered in time, centered in space unconditionally *unstable* scheme we have turned it into an unconditionally *stable* scheme (see Exercise 4 on page 36). However, in contrast to the implicit sample scheme (3.40), the implicity is limited to the one term θ_j^{n+1} which only involved the spatial grid point j. We may therefore move this term from the right-hand side of (3.49) to its left-hand side. After some rewriting we then obtain

$$\theta_j^{n+1} = \left[\theta_j^{n-1} + \chi(\theta_{j-1}^n - \theta_j^n + \theta_j^n)\right](1+\chi)^{-1}. \tag{3.50}$$

where

$$\chi = \frac{2\kappa\Delta t}{\Delta x^2}. \tag{3.51}$$

Thus the numerical algorithm associated with the implicit DuFort-Frankel scheme (3.49) 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, it is unconditionally stable.

It remains to analyse the consistency of the scheme. To do this we use the Taylor series expansions of Section 1.8. Thus we have the series

$$\theta_{j\pm1}^n = \theta_j^n \pm [\partial_x\theta]_j^n\,\Delta x + \frac{1}{2}\left[\partial_x^2\theta\right]_j^n\Delta x^2 \pm \frac{1}{6}\left[\partial_x^3\theta\right]_j^n\Delta x^3 + \mathcal{O}(\Delta x^4), \tag{3.52}$$

and

$$\theta_j^{n\pm1} = \theta_j^n \pm [\partial_t\theta]_j^n\,\Delta t + \frac{1}{2}\left[\partial_t^2\theta\right]_j^n\Delta t^2 \pm \frac{1}{6}\left[\partial_t^3\theta\right]_j^n\Delta t^3 + \mathcal{O}(\Delta t^4). \tag{3.53}$$

Substitution of these series in (3.49) then gives

$$\left[\partial_t^2\theta\right]_j^n - \kappa\left[\partial_x^2\theta\right]_j^n = -\varkappa\left[\partial_t^2\theta\right]_j^n + O(\Delta t^2) + O(\Delta x^2). \tag{3.54}$$

where

$$\varkappa = \kappa \left(\frac{\Delta t}{\Delta x} \right)^2 . \tag{3.55}$$

To be consistent all the terms on the right-hand side of (3.54) 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 Δt tends to zero faster than Δx. We therefore note that (3.54) only converges to the analytic equation if $\varkappa \to 0$ when $\Delta x \to 0$ and $\Delta t \to 0$. Thus 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 $\varkappa \to 0$ when $\Delta x \to 0$ and $\Delta t \to 0$.

As already mentioned at the end of Section 2.2 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-linerar interaction among the various wavelengths which in turn is responsible for a more or less continuous cascade of energy towards smaller and smaller scales. Neglect of dissipating or smoothing out the energy contained in the upper part of the energy spectrum leads to 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 just blows up.

When the diffusion term is used for this purpose it does not represent any of the physical process that we want to resolve. Rather it is introduced as a numerical method to avoid our model to blow up. Nevertheless it represent the impact of physics acting on a smaller scale than our model resolves. It is therefore a parameterization. Since this parameterization and/or the parameters it contains may change in accord with the models resolution (i.e. grid size) we refer to it as *subgrid scale parameterization* (SGS). Note that the SGSs are smaller than our grid resolution. ote also that for a given grid size Δx the resolution is determined by the Nuquist wavelength $2\Delta x$.

3.7 The Crank-Nicholson scheme

We will now look at 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 that its truncation error is second order in both time and space.

To develop this scheme it is first convenient to define

$$\delta_x^2 \psi_j^n = \psi_{j-1}^n - 2\psi_j^n + \psi_{j+1}^n . \tag{3.56}$$

Thus the forward in time centered in space scheme (3.4) may be written

$$\theta_j^{n+1} = \theta_j^n + \frac{\kappa \Delta t}{\Delta x^2} \delta_x^2 \theta_j^n . \tag{3.57}$$

Recall that this scheme is conditionally stable under the condition

$$\chi \equiv \frac{2\kappa \Delta t}{\Delta x^2} \leq 1 . \tag{3.58}$$

From Section 3.5 we recall that an implicit scheme is always stable. Thus we expect that the consistent and implicit scheme

$$\theta_j^{n+1} = \theta_j^n + \frac{\kappa \Delta t}{\Delta x^2} \delta_x^2 \theta_j^{n+1}. \tag{3.59}$$

is unconditionally stable. We note that the only difference between the schemes (3.57) and (3.59) is that the former is explicit while the latter is implicit. We also note that (3.59) is similar to (3.40) except that (3.59) is forward in time while (3.40) is centered in time.

We now combine the two schemes to obtain

$$\theta_j^{n+1} = \theta_j^n + \kappa \frac{\Delta t}{\Delta x^2} \left[\gamma \delta_x^2 \theta_j^{n+1} + (1-\gamma) \delta_x^2 \theta_j^n \right], \tag{3.60}$$

where γ is a number so that $0 \leq \gamma \leq 1$. If $\gamma = 0$ then (3.60) reduces to the the scheme (3.57), that is, a pure explicit scheme. If $\gamma = 1$ then (3.60) reduces to the pure implicit scheme (3.59).

If γ is between 0 and 1 then the scheme is an implicit/explicit scheme which is sometimes refererred to as a *semi-implicit scheme*. To analyse the stability of the scheme we use von Neumann's method. The growth factor G then becomes

$$G = \frac{1 - (1-\gamma)\chi(1 - \cos \alpha \Delta x)}{1 + \gamma \chi (1 - \cos \alpha \Delta x)}. \tag{3.61}$$

Recalling that the condition $|G| \leq 1$ or that $-1 \leq G \leq 1$ is a sufficent condition for numerical stability we find that $G \leq 1$ is always satisfied while $-1 \leq G$ is satisfied if

$$\chi(1 - 2\gamma) \leq 1. \tag{3.62}$$

Thus the scheme (3.60) 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. We also note that this shows that the pure implicit scheme (3.59) 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 only *conditionally stable* under the condition given in (3.62). We also note that for $\gamma = 0$ we retrieve the condition (3.31) of Section 3.3, that is, $\chi \leq 1$.

The value $\gamma = \frac{1}{2}$ is special. It constitutes the critical value at which the scheme (3.60) is still unconditionally stable. For this special value the scheme becomes

$$\theta_j^{n+1} = \theta_j^n + \frac{1}{2} \kappa \Delta t \left[\frac{\theta_{j-1}^{n+1} - 2\theta_j^{n+1} + \theta_{j+1}^{n+1}}{\Delta x^2} + \frac{\theta_{j-1}^n - 2\theta_j^n + \theta_{j+1}^n}{\Delta x^2} \right], \tag{3.63}$$

which is the scheme named the Crank-Nicholson scheme. Using The Taylor series expansion (**??**) to substitute for the centered differences on the right-hand side of (3.63) we obtain

$$\frac{\theta_j^{n+1} - \theta_j^n}{\Delta t} = \frac{1}{2} \kappa \left(\left[\partial_x^2 \theta \right]_j^{n+1} + \left[\partial_x^2 \theta \right]_j^n \right) + \mathcal{O}(\Delta x^2). \tag{3.64}$$

Expanding θ_j^{n+1} and $[\partial_x^2\theta]_j^{n+1}$ using Taylor series gives

$$\frac{\theta_j^{n+1}-\theta_j^n}{\Delta t} = [\partial_t\theta]_j^n + \frac{1}{2}\left[\partial_t^2\theta\right]_j^n \Delta t + \mathcal{O}(\Delta t^2), \tag{3.65}$$

$$\left[\partial_x^2\theta\right]_j^{n+1} = \left[\partial_x^2\theta\right]_j^n + \left[\partial_t(\partial_x^2\theta)\right]_j^n \Delta t + \mathcal{O}(\Delta t^2). \tag{3.66}$$

Substituting these series in (3.64) and rearranging terms then gives

$$[\partial_t\theta]_j^n = \kappa\left[\partial_x^2\theta\right]_j^n - \frac{1}{2}\left\{[\partial_t^2\theta]_j^n - \kappa\left[\partial_t(\partial_x^2\theta)\right]_j^n\right\}\Delta t + \mathcal{O}(\Delta t^2) + \mathcal{O}(\Delta x^2), \tag{3.67}$$

Furthermore, from the analytic diffusion equation (3.1) we obtain

$$\partial_t^2\theta = \kappa\partial_t(\partial_x^2\theta). \tag{3.68}$$

Thus the second term on the right-hand side of (3.67) vanishes and (3.67) becomes

$$[\partial_t\theta]_j^n = \kappa\left[\partial_x^2\theta\right]_j^n + \mathcal{O}(\Delta t^2) + \mathcal{O}(\Delta x^2), \tag{3.69}$$

which shows that the Crank-Nicholson scheme, besides being unconditionally stable, is also of second order in time and space even though it is forward in time. (se Oppgave 5 på side 36).

3.8 A direct elliptic solver

One should, however, beware that using the Crank-Nicholson scheme (3.63) turns the original parabolic equation into an apparent elliptic equation. By including the implicit terms on the right-hand side of (3.63) the original local algorithm is in fact turned into a non-local or global algorithm. Due to the inclusion of these terms the solution at time level $n+1$, i.e., θ_j^{n+1}, in addition to depend on the solution at the previous time level n also depends on the solution at the adjacent space points $\pm\Delta x$ away at the same time level. This is best illustrated by rearranging the terms in (3.63) as follows

$$-\frac{\kappa\Delta t}{2\Delta x^2}\theta_{j-1}^{n+1} + \left(1+2\frac{\kappa\Delta t}{2\Delta x^2}\right)\theta_j^{n+1} - \frac{\kappa\Delta t}{2\Delta x^2}\theta_{j+1}^{n+1} = \theta_j^n + \frac{\kappa\Delta t}{2\Delta x^2}\left[\theta_{j-1}^n - 2\theta_j^n + \theta_{j+1}^n\right]. \tag{3.70}$$

Thus we cannot solve for θ_j^{n+1} without knowing θ_{j-1}^{n+1} and θ_{j+1}^{n+1}. Imagine we are solving (3.70) for increasing values of j. Then for any arbitrary j we do know θ_{j-1}^{n+1}, but we havent yet solved for $j+1$, and thus we do not know θ_{j+1}^{n+1}. To solve such global systems we must revert to what is commonly referred to as an elliptic solver. Since the diffusion equation is a parabolic equation, and thus constitutes a time marching problem, the ellipric solver we have to apply the elliptic solver for each time step. We are therefore in need of a method whereby an elliptic PDE can be solved efficiently on the computer. The most efficient elliptic solvers are those referred to as direct elliptic solvers[3] One such method is the so called *Gauss elimination* which we will us as

[3] Also so called iterative or indirect elliptic solvers may be used, but they are slower.

an example. It consists of two steps. The first is called a *forward sweep*. Next we find the actual solution by performing a *backward substitution*.

To get started, we first rewrite (3.70) into a more general form,

$$a_j\theta_{j-1}^{n+1} + b_j\theta_j^{n+1} + c_j\theta_{j+1}^{n+1} = h_j^n, \tag{3.71}$$

where a_j, b_j, and c_j represents the coefficents in (3.70). The use of the subscript j attached to these coefficients is to acknowledge that they in general are functions of space. Likewise h_j 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 (3.71) within a finite domain. Thus at $j = 1$ and at $j = J$ θ is determined by the boundary conditions. For conveneince we will here assume that these are simple Dirichlet conditions, that is, θ_1^n and θ_J^n are known functions of time.

For convenience we will also drop the superscript $n + 1$. Thus we are required to solve,

$$a_j\theta_{j-1} + b_j\theta_j + c_j\theta_{j+1} = f_j, \quad j = 2(1)J - 1, \tag{3.72}$$

under the conditions

$$\theta_1 = \hat{\theta}_0, \quad \text{and} \quad \theta_J = \hat{\theta}_L. \tag{3.73}$$

We observe that (3.72) may be written

$$\boldsymbol{\mathcal{A}} \cdot \boldsymbol{\theta} = \mathbf{h}', \tag{3.74}$$

where the tensor $\boldsymbol{\mathcal{A}}$ is the tridiagonal matrix

$$\boldsymbol{\mathcal{A}} = \begin{bmatrix} b_2 & c_2 & 0 & \dots & 0 & 0 \\ a_3 & b_3 & c_3 & \dots & 0 & 0 \\ 0 & a_4 & b_4 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & b_{J-2} & c_{J-2} \\ 0 & 0 & 0 & \dots & a_{J-1} & b_{J-1} \end{bmatrix}. \tag{3.75}$$

and the vectors $\boldsymbol{\theta}$ and $\mathbf{h}'$ are, respectively,

$$\boldsymbol{\theta} = \begin{bmatrix} \theta_2 \\ \theta_3 \\ \vdots \\ \theta_{J-1} \end{bmatrix}, \tag{3.76}$$

and

$$\mathbf{h}' = \begin{bmatrix} h_2 - a_2\theta_0 \\ h_3 \\ \vdots \\ h_{J-1} - c_{J-1}\theta_L \end{bmatrix}. \tag{3.77}$$

Forward sweep

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

$$b_2\theta_2 + c_2\theta_3 = h_2'. \tag{3.78}$$

We then normalize by dividing by b_2, i.e.,

$$\theta_2 + d_2\theta_3 = w_2, \tag{3.79}$$

where

$$d_2 = \frac{c_2}{b_2}, \quad w_2 = \frac{h_2'}{b_2}. \tag{3.80}$$

For $j = 3$ we obtain

$$a_3\theta_2 + b_3\theta_3 + c_3\theta_4 = h_3' \tag{3.81}$$

Substituting for θ_2 from (3.79) and normalizing gives

$$\theta_3 + d_3\theta_4 = w_3 \tag{3.82}$$

where

$$d_3 = \frac{c_3}{b_3 - d_2a_3}, \quad w_3 = \frac{h_3' - a_3w_2}{b_3 - d_2a_3}. \tag{3.83}$$

We now define the recursion formulae

$$d_j = \begin{cases} \frac{c_2}{b_2} & ; \quad j = 2 \\ \frac{c_j}{b_j - d_{j-1}a_j} & ; \quad j = 3(1)J-2 \\ 0 & ; \quad j = J-1 \end{cases} \tag{3.84}$$

and

$$w_j = \begin{cases} \frac{h_2'}{b_2} & ; \quad j = 2 \\ \frac{h_j' - a_jw_{j-1}}{b_j - d_{j-1}a_j} & ; \quad j = 3(1)J-1 \end{cases} . \tag{3.85}$$

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

$$\theta_j + d_j\theta_{j+1} = w_j, \quad j = 2(1)J-2. \tag{3.86}$$

We note from (3.84) that $d_{J-1} = 0$. Hence for $j = J-1$ we obtain the very simple equation

$$\theta_{J-1} = w_{J-1} \tag{3.87}$$

where w_{J-1} isgiven by use of (3.85). In matrix form we have therefore transformed (3.75) into

$$\boldsymbol{\mathcal{A}}' \cdot \boldsymbol{\theta} = \mathbf{w}, \tag{3.88}$$

where the matrix $\boldsymbol{\mathcal{A}}'$ is

$$\boldsymbol{\mathcal{A}}' = \begin{bmatrix} 1 & d_2 & 0 & \dots & 0 & 0 \\ 0 & 1 & d_3 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & d_{J-2} \\ 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix}, \tag{3.89}$$

and the vector $\mathbf{w}$ is

$$\mathbf{w} = \begin{bmatrix} w_2 \\ w_3 \\ w_4 \\ \vdots \\ w_{J-2} \\ w_{J-1} \end{bmatrix}. \tag{3.90}$$

Backward substitution

We are now ready to do the backward substituion. First we note from (3.87) that θ_{J-1} is simply given by w_{J-1} and that the latter is known from (3.85). Second we note that all the w_j's and the d_j's are known using the recursion formulae (3.84) and (3.85). Thus we can solve for all the remaining θ_j's for $j = 2(1)J-2$ simply by using (3.86) backwards, that is,

$$\theta_j = w_j - d_j\theta_{j+1} \quad \text{for} \quad j = J-2(-1)2. \tag{3.91}$$

This method is very simple to program, and 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 separate Computer Problems.

Exercises

1. Show that the scheme (3.12) is unconditionally unstable, that is, that $|G| > 1$ regardless of the choice made for Δt and Δx.

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

3. Show that the growth factor associates with the scheme (3.40) is

$$G = \left[1 + \frac{4\kappa\Delta t}{\Delta x^2}(1 - \cos\alpha\Delta x)\right]^{-\frac{1}{2}} \tag{3.92}$$

   and hence that the scheme is unconditionally stable. Also show that $|G| < 1$ for all wavelengths. Note that $|G|$ decreases as Δt increases.

4. Show that the growth factor for the DuFort-Frankel scheme (3.49) is

$$G_{1,2} = \frac{\gamma \cos \alpha \Delta x \pm \sqrt{1 - \gamma^2 (\sin \alpha \Delta x)^2}}{1 + \gamma}; \quad \gamma = \frac{2\kappa \Delta t}{\Delta x^2}. \tag{3.93}$$

5. Show that the expression (3.61) is indeed the expression for the growth factor of the scheme (3.60) when using von Neumanns analysis method.

Kapittel 4

THE ADVECTION PROBLEM

4.1 Finite difference approximations

As we did for the diffusion problem we simplify the advection equation to one-dimension in space. Thus we will study numerical methods to solve the following continuous equation:

$$\partial_t \theta + u \partial_x \theta = 0, \tag{4.1}$$

where $u(x,t)$ is the advection speed along the x-axis. As alluded to in Chapter 1 (4.1) is hyperbolic. Note that in general u varies in time and space. We will nevertheless in many instances below assume that it is uniform in time and space, that is, consider $u = u_0 =$ constant.

As before we develop an algorithm by which (4.1) can be solved by numerical means by making finite difference approximations of the derivatives. In this we use, as we did for the diffusion equation, Taylor series expansions as outlined in Section 1.8. One such scheme is the forward in time, centered in space scheme, or

$$\theta_j^{n+1} = \theta_j^n - u\frac{\Delta t}{2\Delta x}\left(\theta_{j+1}^n - \theta_{j-1}^n\right), \quad \left\{ \begin{array}{l} j = 2(1)J-1 \\ n = 0(1)\ldots \end{array} \right. , \tag{4.2}$$

which worked well for the diffusion problem. However, if the scheme is going to be of any use to us, we require that it is stable and consistent in a numerical sense. Since the finite difference approximateion (4.1) is based on Taylor series expansions we know apriori that the latter requirement is already satisfied.

To analyse its stability we use von Neumann's method as outlined in the previous chapter regarding the diffusion equation. Thus we start by substituting the Fourier component (3.15) into (4.2) to obtain

$$\Theta_{n+1} = \Theta_n - \frac{u\Delta t}{2\Delta x}\left(e^{i\alpha\Delta x} - e^{-i\alpha\Delta x}\right)\Theta_n \tag{4.3}$$

where the common factor $e^{i\alpha j\Delta x}$ is removed. Recalling the definition of the growth factor (3.21), and noting that $2i\sin\alpha\Delta x = e^{i\alpha\Delta x} - e^{-i\alpha\Delta x}$, we find that

$$G = 1 - i\frac{u\Delta t}{\Delta x}\sin\alpha\Delta x. \tag{4.4}$$

We observe that the growth factor now is a complex number with a real part given by 1 and an imaginary part given by $\frac{u\Delta t}{\Delta x}\sin\alpha\Delta x$. To find its required absolute value 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[1]. Thus

$$|G| = \sqrt{1 + \left(\frac{u\Delta t}{\Delta x}\right)^2 \sin^2\alpha\Delta x} \geq 1, \tag{4.5}$$

which proves that the scheme is unconditionally unstable. Thus:

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

We emphasize that although this scheme worked fine for the diffusion problem, it is totally unacceptable with regard to the advection problem. We also underscore that this works both ways. As we will show now, what works for the advection problem is the centered in time, centered in space scheme that we strongly emphasized was totally useless for the diffusion problem. This does not come as a total surprise. As alluded to in Chapter 1 the diffusion equation is parabolic while the advection equation is hyperbolic. We should therefore expect that the numerical methods to be used are different.

Thus let us consider the centered in time, centered in space scheme instead, that is,

$$\theta_j^{n+1} = \theta_j^{n-1} - u\frac{\Delta t}{\Delta x}\left(\theta_{j+1}^n - \theta_{j-1}^n\right), \quad \begin{cases} j = 2(1)J-1 \\ n = 0(1)\ldots \end{cases}. \tag{4.6}$$

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, but do not incorporate any information from the point x_j, t^n itself. In a sense we are "leapfrogging" the point x_j, t^n. The scheme is traditionally fairly popular for several reasons. For one, the scheme is, as shown in Section 4.2 below, neutrally stable under the condition that $u\Delta t \leq \Delta x$, the so called *Courant-Friedrich-Levy condition* or CFL condition. That the scheme is neutral means, as shown in Section 4.2, that the absolute value of the growth factor is one, or $|G| = 1$. 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 desireable property. Secondly the scheme is of second order accuracy in the sense that the truncation error is of $\mathcal{O}(\Delta t^2)$ and $\mathcal{O}(\Delta x^2)$. However, as shown in Section 4.5, the scheme contains *numerical dispersion*, that is, gives rise to an artificial dispersion of energy. The scheme also contains *unphysical modes* that has to be dealt with. All these properties are touched upon in Computer problem No. 3 of the Computer Problem Notes, and the reader is encouraged to solve it.

Another popular and stable scheme in common use is the so called *upwind* or *upstream* scheme. As the name indicates the scheme use 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

[1] 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.

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,

$$\theta_j^{n+1} = \theta_j^n - u\frac{\Delta t}{\Delta x}\begin{cases} \theta_j^n - \theta_{j-1}^n & u \geq 0 \\ \theta_{j+1}^n - \theta_j^n & u < 0 \end{cases} . \tag{4.7}$$

The scheme is stable under the CFL condition, that is, stable as long as $u\Delta t \leq \Delta x$. We note for later convenience that if we assume that $u > 0$ then we may write (4.7) as

$$\theta_j^{n+1} = (1 - C)\theta_j^n + C\theta_{j-1}^n \tag{4.8}$$

where

$$C = \frac{u\Delta t}{\Delta x} \tag{4.9}$$

is named the *Courant number*. Since the scheme is forward in time and one-sided in space it is only of $\mathcal{O}(\Delta t)$ and $\mathcal{O}(\Delta x)$, that is, of first order in time and space. This is in contrast to the leapfrog scheme that was of second order. The scheme also has other unwanted numerical characteristics such as numerical diffusivity. Moreover, this diffusivity depends on the particular choice made for Δt and Δx, and increases as the Courant number decreases. For this reason the present authors do not recommend the scheme.

Since the upwind scheme is forward in both time and space its accuracy is $\mathcal{O}(\Delta t)$ og $\mathcal{O}(\Delta x)$, an accuracy one order of magnitude less than the leapfrog scheme. It is however consistent since it is based on Taylor series to produce the finite difference approximations. Despite of this the upwind scheme has, as detailed in Section 4.6, one major drawback. It contains what we refer to as numerical diffusion. In addition this diffusion depends on the time step and space increment chosen, and depending on the choice this diffusion may be large and sometimes larger than the actual physical diffusion of the original problem. It therefore tends to smooth out the solution as time progresses, in aprticulat areas where large gradients appear, e.g. fronts in the ocean and atmosphere.

A third possibility is 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, θ_j^n, is replaced by an interpolated value using the adjacent grid points, that is,

$$\theta_j^n = \frac{1}{2}\left(\theta_{j+1}^n + \theta_{j-1}^n\right), \tag{4.10}$$

in which case the scheme becomes

$$\theta_j^{n+1} = \frac{1}{2}\left(\theta_{j+1}^n + \theta_{j-1}^n\right) - u\frac{\Delta t}{2\Delta x}\left(\theta_{j+1}^n - \theta_{j-1}^n\right). \tag{4.11}$$

This scheme is conditionally stable, but it is not a consistent scheme. Its name also indicates that it contains numerical diffusion. Because of these rather disadvantageous properties we do not recommend the use of this scheme (cf. the solutions to Computer Problem No. 3 contained in the Computer Problem Notes).

There are actually a number of advective schemes recommended by various authors over the past 40 years or so (see e.g., *O'Brien*, 1986, page 165 and onwards for a summary). Many

of these are now oblivious, since the reason for developing them were based on the need for computer efficiency combined with accuracy. As computers have grown in capacity the need for these schemes becomes less obvious. Most modern codes today actually use higher order schemes, say schemes of $\mathcal{O}(\Delta t^4, \Delta x^4)$ or higher (cf. Section 5.1).

4.2 Stability: The Courant-Friedrich-Levy condition

To show that the leapfrog scheme (4.6) is indeed stable we make use of von Neumann's method as outlined several times already and introduced in Section 3.3. Thus we first replace the dependent variable θ in (4.6) by its discrete Fourier component (3.20) to give

$$\Theta_{n+1} = \Theta_{n-1} - 2iu\frac{\Delta t}{\Delta x}\sin\alpha\Delta x\Theta_n \tag{4.12}$$

in (4.6). To find the growth factor G we first make use of (3.21) and then multiply by the growth factor to obtain

$$G^2 + 2i\lambda G - 1 = 0, \tag{4.13}$$

where

$$\lambda = u\frac{\Delta t}{\Delta x}\sin\alpha\Delta x. \tag{4.14}$$

The solution to this equation is

$$G_{1,2} = -i\lambda \pm \sqrt{1-\lambda^2}. \tag{4.15}$$

Thus the growth factor is complex under the condition that the radikand is positive. This was to be expected since the factor in front of the first order term in (4.13) 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. 4.5 of Section 4.1)

$$|G_{1,2}| = \sqrt{G_{1,2}G^*_{1,2}} = 1. \tag{4.16}$$

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 (4.6) is therefore *neutrally stable* implying that the energy associated with θ is conserved. We note that this is in line with the property of advection processes outlined in Section 2.3.

To obtain this desirable property the condition we must require that the radikand in (4.15) is positive. Hence the condition

$$1 - \lambda^2 \leq 0 \quad \text{or} \quad |\lambda| \leq 1 \tag{4.17}$$

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

$$|u|\frac{\Delta t}{\Delta x} \leq 1 \tag{4.18}$$

then (4.17) is satisfied, that is, the condition (4.18) is a sufficient condition for stability for the leapfrog scheme (4.6). Moreover under this condition the leapfrog scheme also becomes neutrally stable. The condition or criterion (4.18) is the CFL condition. The ratio or number

$$C = |u|\frac{\Delta t}{\Delta x} \tag{4.19}$$

is the *Courant number*. Since Δx 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 condition on the time step Δt.

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 Δt 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.

4.3 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 3.1 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 4.1 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

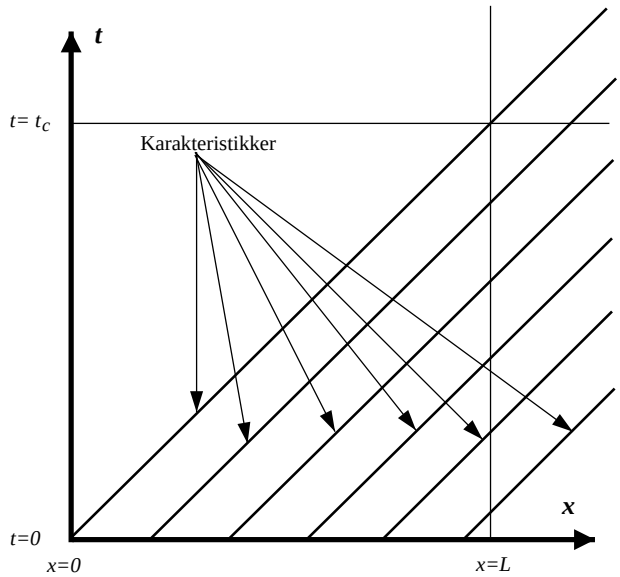
$$\theta_j^1 = \theta_j^0 - u\frac{\Delta t}{2\Delta x}\left(\theta_{j+1}^0 - \theta_{j-1}^0\right). \tag{4.20}$$

For the time level 2 and onwards we then use the leapfrg scheme (4.6). The step (4.20) and more generally the forward in time, centered in space scheme is usually referred to as the Euler scheme. We emphasize that although this 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 4.8).

4.4 Interpreting the CFL condition in terms of physics: The method of characteristics

Let the slopes

$$\frac{D^*x}{dt} = u(x,t) \tag{4.21}$$



Figur 4.1: Sketch of the characteristics in the x, t plane. For $u = u_0 =$ konstant > 0 the characteristics are straight lines sloping to the the right in x, t space as shown. If $x = L$ demarks the end of the computational domain, then all information about the initial condition is lost for times such that for $t > t_c$.

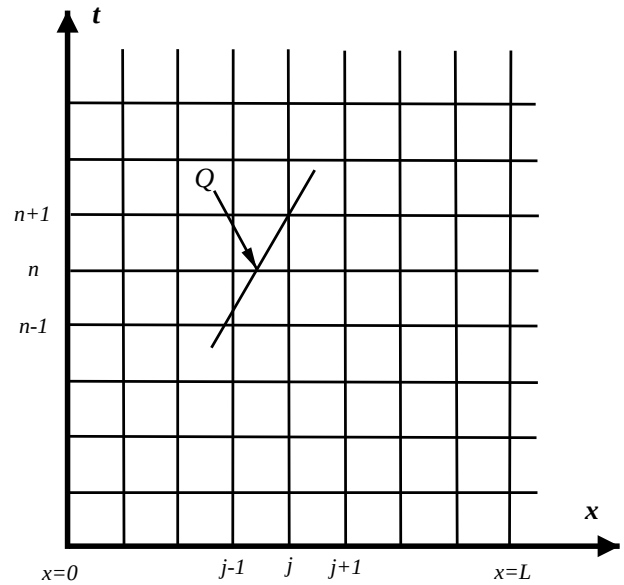
define special curves in the t, x space (cf. Fig. 4.1), and let us simultaneously define the special differential operator

$$\frac{D^*}{dt} \equiv \partial_t + \frac{D^* x}{dt}\partial_x. \tag{4.22}$$

Then (4.1) may be rewritten to read

$$\frac{D^*\theta}{dt} = 0 \quad \text{along the slopes} \quad \frac{D^* x}{dt} = u. \tag{4.23}$$

We commonly refer to the slopes defined by (4.21) as the *characteristics* and (4.21) itself as the *characteristic equation*. Since the solutions to (4.1) also are solutions to (4.23) we often refer (4.23) as the *compatibility equation*. We observe that (4.23) tells us that θ is conserved along the characteristics (4.21). Thus if we know the solution at time level $n = 0$, that is, $\theta(x, t = 0)$ for $0 < x < L$, then the solution at a random time level n and at a particular x_j point in space is found by simply following the characteristic back toward the initial time level $n = 0$ as illustrated in Figure 4.1. For $u = u_0$, where u_0 is a constant the characteristics deform to straight lines with a positive slope $1/u_0$ when $u_0 > 0$. From Figure 4.1 we may conclude that after a critical time $t = t_c = L/u_0$ all information about the initial distribution of θ is lost. Furthermore follows that for times largeer than the critical time level $t > t_c$, the solution in the computational domain $0 < x < L$ is determined wholly by the boundary condition at $x = 0$. Since (4.1) 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 replces the



Figur 4.2: Sketch of the method of characteristics. The distance between the grid points are Δt in the verticaland Δx in the horizntal direction. The slope of the solid line through j, n is derived from (4.21) and is given by $1/u$. The point labelled Q is therefore a distance $u\Delta t$ away from x_j. If $u\Delta t > \Delta x$ then the point Q is located to the left of x_{j-1}.

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 compund 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 7 where we will investigate details concerning open boundary conditions constraining the solutions at open boundaries.

We note in passing that since (4.1) and (4.23) are compatible, a solution to (4.23) is also automatically a solution to (4.1). We may therefore solve (4.23) employing numerical methods as well as (4.1) 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 also be applied to much more complex problems and systems (e.g., *Lister*, 1966).

Since in this simple case (4.23) tells us that θ is conserved along the characteristics, the problem is reduced to find the charateristics in the x, t space, that is, to solve (4.21). As is now common we then divide the computational domain in the x, t space into a grid as displayed in Figure 4.2. 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 (4.21). To find the characteristic we simply make a finite difference approximation of (4.21) considering that u_j^{n+1} can be approximated by u_j^n. Hence

$$x_j - x_Q = u_j^n \Delta t \tag{4.24}$$

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, (4.24) is really an equation which determines the location of x_Q. Commonly the location of the x_Q is in between the grid point, for instance between the points x_j and x_{j-1}. Since θ_j^n is known for all grid points, we may interpolate linearly between the adjacent grid points to find θ_Q^n, or the value of θ at the location x_Q at time level $t = t^n$. To this end we may use a two point linear interpolation. Thus

$$\theta_Q^n = \theta_{j-1}^n + \frac{x_Q - x_j}{\Delta x}(\theta_j^n - \theta_{j-1}^n) \tag{4.25}$$

Substituting x_Q from (4.24) into (4.25) than gives

$$\theta_Q^n = (1 - C)\theta_j^n + C\theta_{j-1}^n, \tag{4.26}$$

where

$$C = u_j^n \frac{\Delta t}{\Delta x} \tag{4.27}$$

is the Courant number. Since θ in accord with (4.23) is conserved along the characteristic we get as a first guess

$$\theta_j^{n+1} = \theta_Q^n = (1 - C)\theta_j^n + C\theta_{j-1}^n. \tag{4.28}$$

Since the characteristic are the curves along which information is propagated (4.28) tells us that θ_j^{n+1}, that is, the value of θ at the point (x_j, t^{n+1}) in time and space equals the value of θ at the point (x_Q, t^n) in time and space (cf. Figure 4.2).

Figure 4.2 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 (4.24) that $x_Q < x_j$. Moreover (4.24) 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 (4.7) we observe that for $u = u_0 > 0$ the information used to compute θ_j^{n+1} does originate from the two points θ_j^n and θ_{j-1}^n. In fact we may rewrite (4.7) for $u = u_0 > 0$ to give

$$\theta_j^{n+1} = (1 - C)\theta_j^n + C\theta_{j-1}^n, \quad C = u_0 \frac{\Delta t}{\Delta x} \tag{4.29}$$

which matches (4.28) exactly. Thus from (4.29) follows that the upwind scheme may be interpreted as the value of θ at the time level $n + 1$, that is, θ_j^{n+1}, is found by a simple weighting of the values θ_j^n and θ_{j-1}^n using the Courant number as weight. What the method of characteristics (4.28) 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 (4.18) must be satisfied.

Moreover, if $u\Delta t > \Delta x$ then the characteristic through x_j at time level $n + 1$ (cf. Figure 4.2) 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 (4.29) to calculate θ 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 (4.29) 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 Δx and Δt, that is, $\Delta x/\Delta t$, is most often referred to as the signal speed of the grid. The CFL criterion (4.18) 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} og x_{j+1} at time level n.

4.5 Numerical dispersion

Fra andre sammenhenger innenfor geofysikk kjenner vi til fenomenet dispersivitet, særlig gjelder dette bølgefenomener (for instance tyngdebølger på grunt vann, planetære Rossby bølger). Dispersivitet betyr at forskjellige bølgelengder forflytter seg med forskjellig fasehastighet. Matematisk uttrykkes dette for bølger ved at bølgefrekvensen, ω, er en ikke-lineær funksjon av bølgetallet, α, that is, $\omega = \omega(\alpha)$, slik at gruppehastigheten, $c_g = \partial_\alpha \omega$ er forskjellig fra en konstant og slik at fasehastigheten $c = \omega/\alpha = c(\alpha)$.

Anvender vi en bølgeløsning for adveksjonslikningen (4.1), that is setter $\theta = \Theta_0 e^{i\alpha(x-ct)}$, finner vi at fasehastigheten $c = u$, that is at når u er konstant beveger alle bølger seg med den samme fasehastigheten, nemlig u, uavhengig av bølgelengden. Den analytiske (fysiske) løsningen er altså *ikke* dispersiv.

Den samme analysen kan vi gjennomføre for den numeriske endelige differanse tilnærmelsen av (4.1). Som eksempel tar vi utgangspunkt i leapfrogskjemaet (4.6), idet dette var det gunstigste med tanke på numerisk dempning. Vi setter derfor inn i (4.6) den numeriske utgaven av en bølge, that is, en diskret Fourier komponent,

$$\theta_j^n = \Theta_0 e^{i\alpha(j\Delta x - cn\Delta t)}, \tag{4.30}$$

og får etter litt regning at

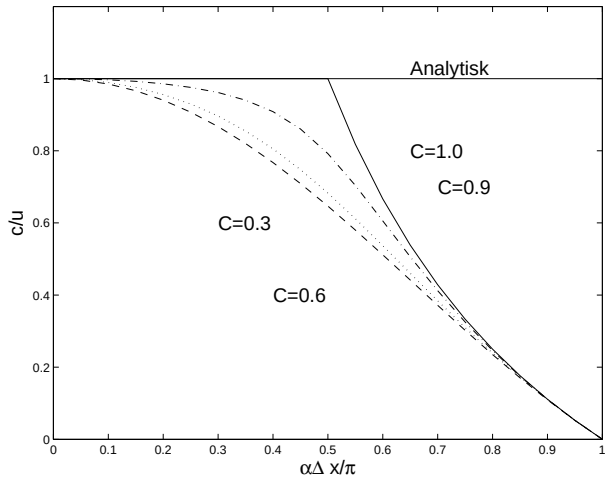
$$c = \frac{1}{\alpha \Delta t} \arcsin\left(u \frac{\Delta t}{\Delta x} \sin \alpha \Delta x\right), \tag{4.31}$$

eller normalisert at

$$\frac{c}{u} = \frac{1}{\pi \alpha' C} \arcsin\left(C \sin \alpha'\right), \tag{4.32}$$

hvor $\alpha' = \alpha \Delta x/\pi$ og C er Couranttallet.

Av Figur 4.3, som viser den normaliserte fasehastigheten som funksjon av det normaliserte bølgetallet, α', som gitt i (4.32), ser vi tydelig at den numeriske dispersjonsrelasjonen avviker fra den korrekte analytiske, nemlig $c/u = 1$. Videre ser vi at de forskjellige bølgelengdene vil forplante seg med forskjellig hastighet, og dermed at leapfrogskjemaet er det vi kaller *numerisk dispersivt*. Allerede for $\alpha' = 0.5$ ser vi at fasehastigheten er betydelig forsinket i forhold til den analytiske for alle Couranttall $C < 1$. I praksis vil dette medføre at for instance en topp i energien vil bli spredt på grunn av numerisk dispersivitet (jmf. Oppgave 5 i Oppgaveheftet). I tillegg ser vi at for $\alpha' > 0.5$ vil gruppehastigheten c_g endre fortegn. Det betyr at for verdier av $\alpha' > 0.5$ vil bølgen og energien i bølgen forplante seg i motsatt retning. Dette ble vist av *Grotjhan and O'Brien* (1976). Dette er ufysisk og må unngås, for instance ved å ha tilstrekkelig romlig oppløsning (velge Dx liten nok).



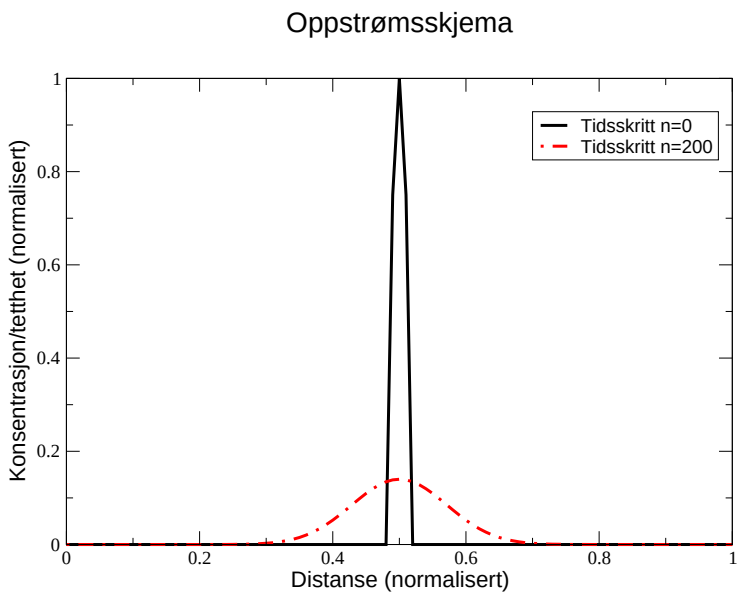
Figur 4.3: Numerical dispersion for the leapfrog scheme. The curves depicts the numerical pahse speed as a function of the wavenumber based on (4.31) for various values of the Courant number $C = u\Delta t/\Delta x$. The vertical axis indicates the phase speed c normalized by the advection speed u. The horizontal axis indicates the wavenumber normalized by $\pi/\Delta x$ where Δx is the space increment or the grid size. The analytic dispersion curve is just a straigt line corresponding to the pahse speed $c = u$, that is $c/u = 1$. Note that as the wavenumber increases (that is the wavelength decreases) the numerical pahse 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. Under these circumstances the energy is propagating in the opposite direction of the waves.

4.6 Numerical diffusion

Som vi så i forrige kapittel gir leapfrogskjemaet en uønsket dispersivitet, særlig for verdier av Couranttallet som avviker fra 1. Dette gjør at leapfrogskjemaet, til tross for at det er nøytralt stabilt og er av annen orden, er uegnet for bruk i problem hvor som inkluderer fronter eller sterke gradienter som er dårlig oppløst.

Disse uheldige sidene ved leapfrogskjemaet medførte at oppstrømsskjemaet lenge var meget populært til løsning av adveksjonsproblemet. Imidlertid har også dette skjemaet uønskede egenskaper. Det er nemlig numerisk diffusivt. Med numerisk diffusjon mener vi at skjemaet gir løsninger som oppfører seg som om den fysiske prosessen diffusjon er tilstede (jmf. Kapittel 2.2). Dette medfører at fronter og/eller skarpe gradienter glattes ut kunstig. Et eksempel på dette er gitt i Figur 4.4 hvor oppstrømsskjemaet er anvendt på en begynnelstilstand som har ekstremt sterke gradienter (jmf. også Oppgave 5 i Oppgaveheftet). Etter 200 tidsskritt er den opprinnelige "bauta"-liknende fordelingen omgjort til en "klokke"-liknende fordeling. Som vist i Opggave 5 i Oppgaveheftet vil også andre fordelinger glattes når en benytter oppstrømskjemaet.

Vi vil nå se litt nærmere på opphavet til en slik diffusjon i oppstrømskjemaet. For dette formål



Figur 4.4: 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 dot-dash curve (red) shows the distribution at time level $n = 200$. The Courant number is $C = 0.5$. Cyclic boundary condiions are used at the boundaries of the computational domain.

omskriver vi (4.7) ved hjelp av en advektiv fluks (jmf. Kapittel 2.3) definert ved

$$F_j^n = \frac{1}{2}\left[(u + |u|)\theta_j^n + (u - |u|)\theta_{j+1}^n\right]\frac{\Delta t}{\Delta x}. \tag{4.33}$$

For $u \geq 0$ ($u = |u|$) er da $F_j^n = \theta_j^n |u| \frac{\Delta t}{\Delta x}$, mens for $u < 0$ ($u = -|u|$) er $F_j^n = -\theta_{j+1}^n |u| \frac{\Delta t}{\Delta x}$. Da vil (4.7) kunne skrives

$$\theta_j^{n+1} = \theta_j^n - (F_j^n - F_{j-1}^n), \tag{4.34}$$

som gjelder uansett om u er større eller mindre enn null. Dersom vi setter inn Taylor rekkene for de enkelte leddene i (4.34), that is, setter

$$\begin{aligned} \theta_j^{n+1} &= \theta_j^n + [\partial_t \theta]_j^n \Delta t + \tfrac{1}{2}[\partial_t^2 \theta]_j^n \Delta t^2 + O(\Delta t^3) \\ \theta_{j\pm1}^n &= \theta_j^n \pm [\partial_x \theta]_j^n \Delta x + \tfrac{1}{2}[\partial_x^2 \theta]_j^n \Delta x^2 \pm O(\Delta x^3) \end{aligned} \tag{4.35}$$

følger etter litt regning, og med bruk av at $\partial_t^2 \theta = u^2 \partial_x^2 \theta$, at

$$[\partial_t \theta]_j^n = -u\,[\partial_x \theta]_j^n + \frac{1}{2}|u|(\Delta x - |u|\Delta t)\left[\partial_x^2 \theta\right]_j^n + O(\Delta x^2) + O(\Delta t^2) \tag{4.36}$$

eller ved bruk av Couranttallet definert i (4.18) at

$$[\partial_t \theta]_j^n = -u\,[\partial_x \theta]_j^n + \frac{|u|\Delta x}{2}(1 - |C|)\left[\partial_x^2 \theta\right]_j^n + O(\Delta x^2) + O(\Delta t^2). \tag{4.37}$$

Til annen orden i tid og rom er altså oppstrømsskjemaet en diffusjons-adveksjonslikning (Kapittel 2),

$$\partial_t \theta + u \partial_x \theta = \kappa \partial_x^2 \theta \tag{4.38}$$

hvor diffusjonskoeffisienten κ er gitt ved

$$\kappa = \frac{1}{2}(1 - |C|)|u|\Delta x. \tag{4.39}$$

Dermed ser vi at de ledd vi sløyfet i (4.7) (av $O(\Delta x)$ og $O(\Delta t)$) gir et diffusjonsledd med en diffusjonskoeffesient gitt ved (4.39). Vi legger merke til at diffusjonen er null dersom øvre grense i CFL kriteriet (4.18) benyttes. Samtidig ser vi at skjemaet er konsistent idet diffusjonsleddet går mot null når Δx og Δt går mot null uavhengig av hverandre.

4.7 Flux correction: Minimizing numerical diffusion

In contrast to the second order leapfrog scheme the first order upwind (or upstream) scheme is a positive definite scheme. Thus if the distribution of say $\theta(x, t)$ at some arbitraty time t is such that $\theta \geq 0$ for all x then it is guarantined that $\theta \geq 0$ also for $t = t + \Delta t$. Another important property, and exemplified in Figure 4.4, is that the maximum value is correctly propagated at any time without any dispersion. These are valuable properties that is well worth retaining in any scheme. The question therefore arises if it is possible to construct a scheme or method whereby these properties are retained and at the same time avoids, or at least minimizes, the numerical diffusion inherent in the upwind scheme as shown in Section 4.6? There are several schemes that offers a solution. Here we will present one of them called MPDATA[2] which was first suggested by *Smolarkiewicz* (1983) (see also *Smolarkiewicz and Margolin*, 1997).

Like many other schemes that offers a solution also the MPDATA method bases its solution on the use of flux correction. The particular method used by *Smolarkiewicz* (1983) is to introduce an advective flux that compensates for the diffusive flux introduced by the first order upwnd scheme. First we note that the upwind scheme

I henhold til *Smolarkiewicz* (1983) kan den innebyggede numeriske diffusjonen motvirkes ved å påføre en tilsvarende negativ diffusjon. Som (4.38) indikerer gir bruk av oppstrømsskjemaet en numerisk løsning som svarer til en analytisk løsning av likningen

$$\partial_t \theta + \partial_x (u\theta) = \partial_x (\kappa \partial_x \theta), \tag{4.40}$$

hvor κ er gitt fra (4.39). Smolarkiewicz's forslag til løsning bygger på observasjonen at man istedet for å løse (4.40) bør løse likningen

$$\partial_t \theta + \partial_x [(u^* + u)\theta] = \partial_x (\kappa \partial_x \theta) \tag{4.41}$$

hvor u^* er en "motdiffusjonshastighet" definert slik at $\partial_x (u^* \theta) = \partial_x (\kappa \partial_x \theta)$, that is

$$u^* = \frac{\kappa \partial_x \theta}{\theta}. \tag{4.42}$$

[2]MPDATA is an abbreviation of "Multiple Positive Definite Advection-Transport Algorithm"

Forslaget går altså ut på å legge til en kunstig advektiv fluks $F^* = u^*\theta$ som i det kontinuerlige tilfellet nøyaktig motvirker diffusjonen på høyre sie av (4.40). Dette ser vi enkelt ved å innsette (4.42) i (4.41), som da blir identisk med den opprinnelige adveksjonslikningen (4.1). Altså vil, teoretisk sett, innføringenen av en advektiv fluks med en hastighet lik motdiffusjonshastigheten u^* medføre at den numeriske diffusjonen oppheves eksakt. Dersom nevneren i (4.42) går mot null raskere enn $\partial_x\theta$ går mot null vil u^* gå mot uendelig. For å unngå dette foreslo *Smolarkiewicz* (1983) og legge til et lite tall i nevneren i (4.42).

Den praktiske fremgangsmetoden foreslått av *Smolarkiewicz* (1983)[3] er også elegant. For hvert tidsskritt består den av en totrinnsrakett. Først trinn består i å finne en midlertidig løsning eller predikator (eng: "predictor"), θ^p. Denne baseres på en løsning av adveksjonslikningen (4.1) ved hjelp av et lavordensskjema slik som oppstrømsskjemaet (4.34). Det vesentlige her er ikke at oppstrømsskjemaet brukes, men at det valgte skjemaet har de egenskapende at den ikke skaper negative verdier dersom fordelingen opprinnelig er positiv definit, og at toppene i fordelingen forfluttes korrekt, altså at skjemaet gir løsninger av orden $O(\Delta t)$ og $O(\Delta x)$ som har de samme egenskapene, spesielt bevaring av energi og monotonisitet (at det ikke skapes falske negative verdier for en størrelse som er en positiv definite) som oppstrømsskjemaet. Predikatoren vil, siden vi bruker oppstrømsskjemaet, være "infisert" av en numerisk diffusjon som har glattet ut alle mer eller mindre skarpe gradienter i tilstanden ved tidskrittet t^n, og slik at de skarpeste gradientene er glattet mest. Imidlertid vet vi at toppene i θ^p er forflyttet til de riktige posisjonene, men dempet i verdi fordi arealet ("energien") under fordelingen ved tidskrittet t^n er bevart. For eksempel vil en tilstand som er en smal klokkefunksjon ved $t = 0$ være fordelt over et større område ved tidskrittet t^n, mens arealet under klokken er det samme som ved tidspunktet $t = 0$ slik som vist i Fig. 4.4. Toppen, eller maksimum, i klokken er forflyttet til riktig posisjon, men har fått en lavere verdi. Dette betyr at alle de initielle og opprinnelige skarpe gradientene er blitt ganske "slappe".

Det som trenges i det andre steget i totrinnsraketten til *Smolarkiewicz* (1983) er derfor at de disse "slappe" gradientene strammes opp. Hans forslag for trinn to, kalt korreksjons-steget (eng: "correction step"), er å korrigere den midlertidige løsningen θ^p ved å løse (4.42) en gang til, men nå med motdiffusjons-hastigheten som gitt i (4.42), that is å løse

$$\partial_t\theta + \partial_x(u^*\theta) = 0. \tag{4.43}$$

Igjen foreslår han å benytte et lavordenskjema, men i tillegg bruke den midlertidige løsningen ("the predictor") som løsningen ved tidskrittet t^n, that is motdiffusjonshastigheten defineres ved

$$u^* = \frac{\kappa\partial_x\theta^p}{\theta^p}, \tag{4.44}$$

mens skjemaet for θ ved tidsskrittet t^{n+1} blir

$$\theta_j^{n+1} = \theta_j^p - (F_j^p - F_{j-1}^p), \tag{4.45}$$

[3]Denne metoden har fått navnet MPDATA som står for "Multiple Positive Definite Advection-Transport Algorithm"

hvor

$$F_j^p = \frac{1}{2}\left[(u^* + |u^*|)\theta_j^p + (u^* - |u^*|)\theta_j^p\right]\frac{\Delta t}{\Delta x}. \tag{4.46}$$

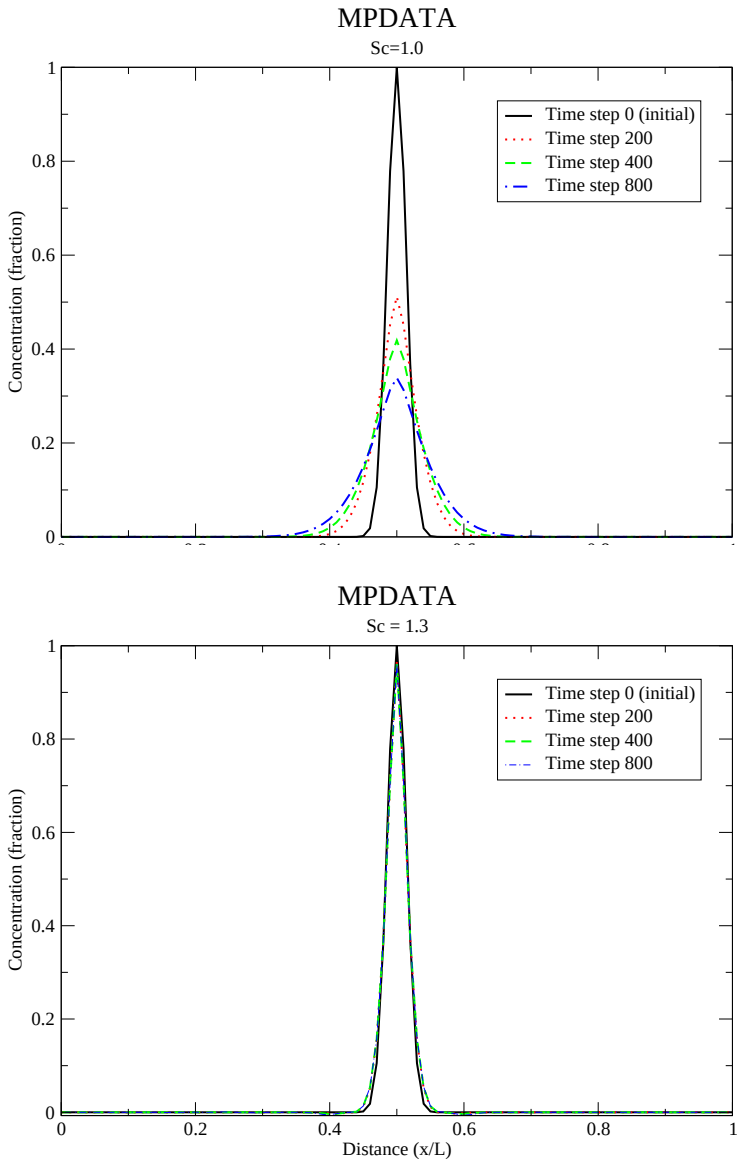
Siden, som vi så over, dette er det samme som å løse en ren adveksjonslikning vil den påførte advektive fluksen ikke medføre noen forflytning av toppen ($u^* = 0$ hvor $\partial_x\theta^p = 0$), men vil føre til en omfordeling av arealet under fordelingen gitt ved predikatoren θ^p. Altså er det kun gradientene som blir påvirket, mens toppen (ekstremalverdien) forblir i ro slik som vist i i øvre panel i Fig. 4.5. At (4.45) gir den ønskede virkning skjønner vi ved å observere at u^* i henhold til (4.44) er null der hvor $\partial_x\theta^p = 0$ er null. Siden κ i henhold til (4.39) er positiv for Couranttall som oppfyller kriteriet for numerisk stabilitet, er fortegnet til u^* bestemt av fortegnet til $\partial_x\theta^p$ (sålenge θ^p er positiv), that is, u^* er negativ til høyre for en topp, mens den er positiv til venstre for en topp. Løsninger av (4.45) vil derfor forflytte løsningen på begge sider av toppen mot toppen, med andre ord stramme opp rundt toppen nettopp slik vi ønsker. I tillegg vil denne oppstrammingen være størst der hvor θ^p er minst og misnt der hvor θ er størst, hvilket også er slik vi ønsker. Altså vil de "slappe" gradientene strammes opp, og, siden arealet under fordelingen bevares, vil maksimumsverdien i toppen øke.

For å oppsummere. Først løses altså adveksjonslikningen (4.1) ved hjelp av oppstrømsskjemaet (4.34). Dette steget kalles prediksjons-steget og gir en midlertidig løsning av θ ved tiden $t^{n+1} = (n+1)\Delta t$ kalt predikatoren. Deretter beregnes en motdiffusjonshastighet, u^*, gitt ved (4.44), og adveksjonslikningen løses på nytt som anvist i (4.45), that is, nå med u erstattet av u^*. Dette steget kalles korreksjons-steget og gir en korrigert verdi av θ.

Som det fremgår av Fig. 4.5 vil imidlertid også korreksjons-steget i Smolarkiewiczs metode (MPDATA) være diffusivt idet det også bygger på et lavordensskjema av typen oppstrømskjema. Dette kan vi rette på ved å innføre et nytt korreksjons-steg hvor nok en ny motdiffusjonshastighet beregnes basert på løsningen fra det første korreksjons-steget, og som deretter settes inn i (4.45). Dette er da begynnelsen på en iterasjonsprosess som kan gjentas i det uendelige. Det krever imidlertid stadig mere regnetid og vil gå utover effektiviteten til programmet vårt. En regnemaskinmessig mer effektiv måte er å gjøre u^* kunstig større ved hjelp av en skaleringsfaktor (her kalt S_c). Verdien av denne skaleringsfaktoren må utprøves nøye, og vil variere fra anvendelse til anvendelse. For store verdier vil gi for mye oppstramning, that is, føre til at toppene blir smalere og derved mere energirike enn de skal være. I Fig. 4.5 nedre panel vises løsningen dersom vi bruker en skaleringsfaktor på $S_c = 1.3$. Forøvrig vises til Oppgave 6 i Oppgaveheftet.

4.8 Unphysical solutions and numerical modes

Over har vi sett på noen egenskaper ved forskjellige skjemaer som gir opphav til numeriske løsninger som inneholder uønskede effekter som egentlig er ufysiske, men som opptrer som tilsynelatende fysikk, altså løsninger som er skapt av det skjemaet vi anvender. Eksempler på slike uønskede effekter var numerisk dissipasjon (ufysisk uttak av energi), numerisk dispersivitet (ufysisk spredning av energi til andre bølgelengder eller frekvenser), og numerisk diffusjon (ufysisk ledning av energi i rom og tid). I tillegg til disse har enkelte skjema også løsninger som kalles ufysiske eller *numeriske moder*, that is, løsninger som ikke er inneholdt i den opprinnelige



Figur 4.5: Eksempel på oppstramning av den numeriske diffusjonen ved bruk Smolarkiewicz metode MPDATA. I begge panelene viser den heltrukne svarte kurven intialtilstanden (tidsskritt $n = 0$). Den røde punktumlinjen viser løsningen etter $n = 200$, den grønne stiplet etter $n = 400$ og den blå stiplet-punktum linjen løsningen etter $n = 800$. Courant tallet er i dette tilfellet satt til $C = 0.5$. Øvre panel viser løsningen med en skaleringsfaktor på $S_c = 1$, mens nedre panel viser løsningen dersom skaleringsfaktoren økes til $S_c = 1.3$. På rendene er det brukt sykliske (eller periodiske) betingelser (jmf. også Oppgave 6 i oppgaveheftet).

kontinuerlige likning.

La oss f.eks. se på adveksjonslikningen (4.1) hvor vi anvender leapfrogskjemaet (4.6). Fra den kontinuerlige likningen følger at dersom begynnelsesbetingelsen settes til en enkel bølge,

$$\theta = \theta_0 e^{i\alpha x}, \tag{4.47}$$

er den fysiske løsningen gitt ved

$$\theta = \theta_0 e^{i\alpha(x-At)} \tag{4.48}$$

hvor som før α er bølgetallet og θ_0 er amplituden i bølgen. Det numeriske leapfrogskjemaet for adveksjonsproblemet (4.6) kan også løses analytisk (jmf. Oppgave 2 på side 54)

$$\theta_j^n = \theta_0 \left[C_1 e^{i\alpha(j\Delta x - \frac{n\chi}{\alpha})} + (-1)^n C_2 e^{i\alpha(j\Delta x + \frac{n\chi}{\alpha})} \right] \tag{4.49}$$

hvor $C_{1,2}$ er to konstanter som bestemmes ut ifra begynnelses- eller initialbetingelsen, og hvor

$$\chi = \arcsin\left[A\frac{\Delta t}{\Delta x}\sin(\alpha\Delta x)\right] \tag{4.50}$$

Likning (4.49) følger av (4.13) og av det faktum at vi i henhold til (4.15) har to muligheter for løsning for vekstfaktoren. Den kontinuerlige likningen er bare derivert en gang med hensyn på tiden. Derav følger at vi bare har en initialbetingelse, som fra (4.47) er gitt ved

$$\theta_j^0 = \theta_0 e^{i\alpha j\Delta x} \tag{4.51}$$

som er utilstrekkelig til å bestemme de to konstantene i (4.49). Dette misforholdet har sin opprinnelse i at leapfrogskjemaet er av annen orden i tid og rom, that is, som om det var et endelig differanseskjema for en kontinuerlig likning av annen orden i tid og rom. Den numeriske løsningen (4.49) krever altså to initialbetingelser mens vi bare har en til disposisjon. Numerisk gjenspeiles dette i at (4.6) ikke kan brukes til å bestemme θ ved tidskrittet $t = \Delta t$, that is θ_j^1, da dette krever at θ_j^{-1}, eller θ ved tidskrittet $t = -\Delta t$ er kjent. Sagt på en annen måte betyr det at den andre konstanten i vår numeriske løsning (4.49) bestemmes utifra hvordan det første tidsskrittet behandles. En meget vanlig metode er å bruke et skjema som er ensidig i tid og sentrert i rom (jmf. f. eks. likning 4.2) for det første steget. Dersom vi bruker det her følger at den numeriske løsningen vår kan skrives (jmf. Oppgave 9.2),

$$\theta_j^n = \theta_0 \left[\left(\frac{1+\cos\chi}{2\cos\chi}\right) e^{i\alpha\left(j\Delta x - \frac{n\chi}{\alpha}\right)} + (-1)^{n+1}\left(\frac{1-\cos\chi}{2\cos\chi}\right) e^{i\alpha\left(j\Delta x + \frac{n\chi}{\alpha}\right)} \right] \tag{4.52}$$

Videre kan det vises at når Δt og Δx begge går mot null, men slik at Couranttallet (4.18) er mindre enn 1, that is at stabilitetskravet er oppfylt, vil det første leddet i (4.52) gå mot den kontinuerlige løsningen (4.48), mens det andre leddet vil gå mot null. Det første leddet i (4.52) svarer derfor til den fysiske moden, mens det andre leddet er en falsk numerisk mode. (På engelsk er dette fenomenet også referert til som “time splitting”). Legg merke til at fortegnet for den ufysiske eller falske numeriske moden skifter tegn for hvert tidsskritt.

Det er svært viktig å merke seg at dersom ikke denne ufysiske eller numeriske moden kontrolleres eller drepes, vil den kunne føre til kraftig støy, og i enkelte tilfelle til numeriske instabiliteter.

4.9 The Asselin filter

Den enkleste måten å bli kvitt den numeriske moden i leapfrog skjemaet, er med jevne mellomrom å foreta ett ensidig i tid, sentrert i rom tidsskritt, that is, et vanlig Euler steg som vist i (4.2). En annen metode som opprinnelig ble foreslått av *Robert* (1966) og senere videreutviklet av *Asselin* (1972) er å foreta en filtrering i tid. En tidsfiltrering av en hvilkensomhelst funskjon i tiden, for instance $\theta(x,t)$, fremkommer ved å anvende operatoren

$$\overline{\theta}(x,t) = \gamma\theta(x,t+\Delta t) + (1-2\gamma)\theta(x,t) + \gamma\theta(x,t-\Delta t) \tag{4.53}$$

hvor γ er en vektfunksjon. Ved bruk av notasjonen innført i Kap. 1.10 kan (4.53) skrives

$$\overline{\theta}_j^n = \theta_j^n + \gamma[\theta_j^{n+1} - 2\theta_j^n + \theta_j^{n-1}]. \tag{4.54}$$

Dersom vi ser på en enkelt periode, that is, setter $\theta_j^n = \hat{\theta}_j e^{i\omega n\Delta t}$, gir (4.54) at

$$\overline{\theta}_j^n = R\theta_j^n. \tag{4.55}$$

Forholdet

$$R(\gamma) = \frac{\overline{\theta}_j^n}{\theta_j^n} = 1 - 2\gamma + 2\gamma\cos\omega\Delta t \tag{4.56}$$

kalles filterets responsfunksjon. For $\gamma = 0.25$ er filteret et vanlig 1-2-1 filter, that is

$$\overline{\theta}_j^n = \frac{1}{4}[\theta_j^{n+1} + 2\theta_j^n + \theta_j^{n-1}], \tag{4.57}$$

hvor tidspunktet n vektlegges med det dobbelte av tidspunktene $n-1$ og $n+1$. I dette tilfellet er responsfunksjonen er gitt ved

$$R(0.25) = \frac{1}{2}(1 + \cos\omega\Delta t) \tag{4.58}$$

Siden $\omega = 2\pi/T$ hvor T er perioden ser vi at et slikt filter gir $R = 0$ for en bølge med periode $T = 2\Delta t$, og $R = 0.5$ for en bølge med periode $T = 4\Delta t$. Altså vil $2\Delta t$ bli filtrert bort mens $4\Delta t$ bølger dempes til halve amplituden. Effekten på de langperiodiske bølgene er derimot minimal.

Det var disse egenskapene ved filteret (4.54) som satte *Robert* (1966) og *Asselin* (1972) på ideen om å benytte et liknende filter for å fjerne den ufysiske moden i leapfrogskjemaet. Dette ble gjort slik. Vi antar at den filtrerte løsningen, $\overline{\theta}_j^{n-1}$, er blitt beregnet og lagret for tidspunktet $(n-1)$, såvel som den ufiltrerte verdien, θ_j^n, ved tidspunktet n. Deretter brukes leapfrogskjemaet til å beregne den nye verdien θ_j^{n+1} ved tidspunktet $(n+1)$ ved hjelp av de ufiltrerte verdiene ved tidspunktet n og de filtrerte verdiene ved tidspunktet $n-1$. For eksempel, dersom likningen vår er den enkle adveksjonslikningen (4.1) gir bruk av leapfrogskjemaet (4.6) at

$$\theta_j^{n+1} = \overline{\theta}_j^{n-1} - u\frac{\Delta t}{\Delta x}(\theta_{j+1}^n - \theta_{j-1}^n). \tag{4.59}$$

En filtrert verdi for tidspunktet n beregnes deretter ved å sette

$$\overline{\theta}_j^n = \theta_j^n + \gamma \left[\theta_j^{n+1} - 2\theta_j^n + \overline{\theta}_j^{n-1} \right]. \tag{4.60}$$

Vi har da ikke noe mere bruk for de filtrerte verdiene ved tidpunktet $n-1$, $\overline{\theta}_j^n$, og kan erstatte denne med de nye filtrerte verdiene ved tidspunktet n, nemlig $\overline{\theta}_j^n$. Vi kan da gjenta prosedyren for å beregne θ_j^{n+2}, og så videre.

Vi skal merke oss at Asselinfilteret påvirker den numeriske stabiliteten, og at den gir opphav til en numerisk diffusjon. Den numeriske diffusjonen øker med økende verdier av faktoren γ, mens den kritiske verdien for stabilitet minker med økende verdi av γ. Det siste nødvendiggjør at tidskrittlengden Δt må minkes med økende verdier av γ. Dette faktum medfører at selvom $\gamma = 1/4$ er ønskelig på grunn av at $2\Delta t$ støy blir øyeblikkelig dempet (og derigjennom den uønskede ufysiske moden), er det fornuftig å bruke en lavere verdi av γ. En ofte brukt verdi er $\gamma = 0.08$. Vi skal også merke oss at gjentatt bruk av selv et svakt Asselinfilter (med $\gamma = 0.08$) gjør at selv de lengre bølgeperiodene til slutt blir påvirket (diffundert). Et slikt filter bør derfor brukes med en viss forsiktighet og ikke nødvendigvis for hvert tidsskritt.

Exercises

1. Show that the CFL criterion for the leapfrog scheme, the diffusive scheme and the upwind scheme all are given by (4.18).

2. Show that (4.49) is a solution to (4.6). Moreover, show that (4.52) follows from (4.49) when the initial distribution is given by (4.51), and where (4.2) is made use of to find θ_j^{-1}. Hint: Show first that $G_{1,2}$ from (4.15) may be written

$$G_1 = e^{-i\chi}, G_2 = e^{i(\chi+\pi)} \tag{4.61}$$

   where χ is given by (4.50).

Kapittel 5

ADVANCED TOPICS

5.1 Higher order advection schemes

As alluded to in Section 1.8 schemes with higher order truncation errors may be constructed using of Taylor series expansion. As an example we show how to construct a fourth order accurate scheme for the advection equation

$$\partial_t \theta + u \partial_x \theta = 0. \tag{5.1}$$

First we recall from the section on Taylor expansions (Section 1.8) that (1.37) and (1.38) in our notation can be written

$$\theta^n_{j\pm 1} = \theta^n_j \pm [\partial_x \theta]^n_j \Delta x + \frac{1}{2}[\partial^2_x \theta]^n_j \Delta x^2 \pm \frac{1}{6}[\partial^3_x \theta]^n_j \Delta x^3 + \mathcal{O}(\Delta x^4) \tag{5.2}$$

Thus follows that

$$\frac{\theta^n_{j+1} - \theta^n_{j-1}}{2\Delta x} = [\partial_x \theta]^n_j + \frac{1}{6}[\partial^3_x \theta]\Delta x^2 + \mathcal{O}(\Delta x^4), \tag{5.3}$$

which we then used to find a second order approximation to the spatial derivative in (5.1). In (5.3) we have used the points adjacent to θ^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 (5.2) becomes

$$\theta^n_{j\pm 2} = \theta^n_j \pm [\partial_x \theta]^n_j 2\Delta x + \frac{1}{2}[\partial^2_x \theta]^n_j (2\Delta x)^2 \pm \frac{1}{6}[\partial^3_x \theta]^n_j (2\Delta x)^3 + \mathcal{O}(\Delta x^4) \tag{5.4}$$

and hence that

$$\frac{\theta^n_{j+2} - \theta^n_{j-2}}{4\Delta x} = [\partial_x \theta]^n_j + \frac{2}{3}[\partial^3_x \theta]\Delta x^2 + \mathcal{O}(\Delta x^4). \tag{5.5}$$

Use of (5.5) to construct a finite difference approximation to $[\partial_x \theta]^n_j$ is as valid as using (5.3). In the limit $\Delta x \to 0$ they both tend to $\partial_x \theta$. We may therefore combine them linearly to give

$$a\frac{\theta^n_{j+1} - \theta^n_{j+1}}{2\Delta x} + b\frac{\theta^n_{j+2} - \theta^n_{j+2}}{4\Delta x} = (a+b)[\partial_x \theta]^n_j + \frac{1}{6}(a+4b)[\partial^3_x \theta]\Delta x^2 + \mathcal{O}(\Delta x^4) \tag{5.6}$$

where a and b are linear weigths yet to be found. We may therefore construct a finite difference approximation to $\partial_x\theta$ by truncating the series

$$[\partial_x\theta]_j^n = \left(\frac{a}{a+b}\right)\frac{\theta_{j+1}^n - \theta_{j+1}^n}{2\Delta x} + \left(\frac{b}{a+b}\right)\frac{\theta_{j+2}^n - \theta_{j+2}^n}{4\Delta x} - \frac{1}{6}\left(\frac{a+4b}{a+b}\right)[\partial_x^3\theta]\Delta x^2 + \mathcal{O}(\Delta x^4) \tag{5.7}$$

Hence by chosing the numbers a and b so that $a + 4b = 0$ we eliminate the second term on the right-hand side of (5.7), and thus the leading truncation error becomes $\mathcal{O}(\Delta x^4)$. Another obvious requirement, necessary to ensure that the right-hand side of (5.7) tends to $\partial_x\theta$ when $\Delta x \to 0$, is that $a + b = 1$ which then gives that $a = \frac{4}{3}$ and $b = -\frac{1}{3}$. Hence a fourth order in space and second order in time scheme for the advection equation becomes

$$\frac{\theta_j^{n+1} - \theta_j^{n-1}}{2\Delta t} + u\left\{\frac{4}{3}\frac{\theta_{j+1}^n - \theta_{j-1}^n}{2\Delta x} - \frac{1}{3}\frac{\theta_{j+2}^n - \theta_{j-2}^n}{4\Delta x}\right\} = 0. \tag{5.8}$$

As we did in Section 4.5 we may analyse this scheme in terms of its dispersion properties. Thus we use

$$\theta_j^n = \Theta_0 e^{i\alpha(\Delta x - cn\Delta t)} \tag{5.9}$$

to substitute in (5.8) which gives

$$c = \frac{1}{\alpha\Delta t}\arcsin\left\{u\alpha\Delta t\left[\frac{4}{3}\left(\frac{\sin\alpha\Delta x}{\alpha\Delta x}\right) - \frac{1}{3}\left(\frac{\sin 2\alpha\Delta x}{2\alpha\Delta x}\right)\right]\right\}. \tag{5.10}$$

To leading order in $\alpha\Delta x$ we then obtain for the fourth order scheme

$$c \approx u\left\{1 - \frac{4}{5!}(\alpha\Delta x)^4 + \cdots\right\} \tag{5.11}$$

Recalling (cf. Section 4.5) the second order in space leapfrog scheme phase speed was

$$c = \frac{1}{\alpha\Delta t}\arcsin\left[u\alpha\Delta t\left(\frac{\sin\alpha\Delta x}{\alpha\Delta x}\right)\right] \approx u\left\{1 - \frac{1}{3!}(\alpha\Delta x)^2 + \cdots\right\}, \tag{5.12}$$

Since $0 \le \alpha\Delta x \le \pi$ it follows that 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

$$\frac{\theta_j^{n+1} - \theta_j^{n-1}}{2\Delta t} + u\left\{\frac{3}{2}\frac{\theta_{j+1}^n - \theta_{j-1}^n}{2\Delta x} - \frac{3}{5}\frac{\theta_{j+2}^n - \theta_{j-2}^n}{4\Delta x} + \frac{1}{10}\frac{\theta_{j+3}^n - \theta_{j-3}^n}{4\Delta x}\right\} = 0. \tag{5.13}$$

is good to $\mathcal{O}(\Delta x^6)$. Thus the dispersion relation becomes

$$c = \frac{1}{\alpha\Delta t}\arcsin\left\{u\alpha\Delta t\left[\frac{4}{3}\frac{\sin\alpha\Delta x}{\alpha\Delta x} - \frac{3}{5}\frac{\sin 2\alpha\Delta x}{2\alpha\Delta x} + \frac{1}{10}\frac{\sin 3\alpha\Delta x}{3\alpha\Delta x}\right]\right\}. \tag{5.14}$$

which to leading order gives

$$c \approx u \left\{ 1 - \frac{36}{7!}(\alpha \Delta x)^6 + \cdots \right\}. \tag{5.15}$$

Comparing (5.15) with (5.11) and (5.12) 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 4.6), troubles at boundaries, and more stringent conditions for numerical stability.

5.2 Non-linear instability

Som nevnt tidligere vil ethvert ikke-lineært problem av hyperbolsk natur for en friksjonsfri væske bli instabilt etter mange nok tidskritt, til tross for at det lineære CFL kriteriet er oppfylt. At dette er tilfelle er lett å overbevise seg selv om bare ved å løse selv et enkelt ikke-lineært, friskjonsfritt hyperbolsk problem. Støy på $4\Delta x$ til $2\Delta x$ skalaen vil før eller senere dukke opp. Amplityden av støyen vil være liten til å begynne med, men etterhvert, og helt uforutsigbart vil støyen vokse eksponensielt. Dermed er løsninen ubrukbar. Vanligvis krediteres *Phillips* (1959) for den første analytiske løsningen som viser dette teoretisk.

Før vi går inn i noen detaljer om dette merker vi oss følgende;

1. Enhver "god" funksjon[1] kan fremstilles som en diskret sum av bølger.
2. Dersom vi har et lineært system eksisterer hver enkelt bølge som om de andre bølgene ikke var tilstede.
3. For et ikke-lineært system vil bølgene vekselvirke og eventuelt danne nye bølgelengder.
4. For en numerisk løsning med en gitt endelig gitteravstand Δx kan ikke alle bølger eksistere.

Punkt 1 kjenner vi godt til. Det forteller oss at enhver 'god' funksjon $\Psi(x)$ med periode $2L$ kan fremstilles som en Fourierrekke, that is, for $x \in [-L, L]$ kan $\Psi(x)$ skrives som

$$\Psi(x) = a_0 + \sum_{m=1}^{\infty} a_m \sin(\alpha_m x) + b_m \cos(\alpha_m x) \tag{5.16}$$

hvor $\alpha_m = m\pi/L$ er (det diskrete) bølgetallet, a_m og b_m er amplituden eller energien for bølgetallet α_m og a_0 er middelverdien av Ψ for $x \in [-L, L]$. Punkt 2 sier oss at dersom systemet er lineært vil det ikke være noen vekselvirkning mellom bølgene. To bølgetog med forskjellig amplitude, retning og bølgelengde vil passere hverandre uten at de endrer hverken amplitude, retning eller bølgelengde. Punkt 3 understreker at dette ikke er tilfelle for ikke-linære bølger. Her

[1] Med "god" menes her at funksjonen eksisterer og at funksjonen selv og alle dens høyere ordens deriverte til en vilkårlig orden er kontinuerlige

vil bølgene etter å ha passert hverandre ha endret enten retning, amplitude eller bølgelengde eller alle tre.

Punkt fire påpeker det faktum at når vi fremstiller funksjonen $\Psi(x)$ diskret med en gitt gitteravstand Δx har vi et båndbegrenset bølgerom hvor bølgelengden ikke kan være kortere enn $2\Delta x$, eller at $\alpha_m \in [0, \pi/\Delta x]$, ellers kan de ikke eksistere som bølger i vårt gitter. Dersom vi har et ikke-lineært system hvor bølgene vekselvirker og produserer bølger med bølgetall $\alpha > \pi/\Delta x$ (bølger med bølgelengder kortere enn $2\Delta x$), kan ikke gitteret vårt oppløse disse, og de vil bli foldet (gitt) til et eller annet lavt bølgetall. La oss noe vilkårlig kalle $\alpha < \pi/2\Delta x$ for lave bølgetall eller lange bølger (that is bølger med bølgelengder større enn $4\Delta x$) og bølger med bølgetall $\pi/\Delta x < \alpha < \pi/2\Delta x$ for høye bølgetall eller korte bølger. De siste er bølger med bølgelengder mellom $2\Delta x$ og $4\Delta x$ som er de korteste bølgene som kan oppløses i et gitter med gitteravstand Δx. Punkt tre og fire over sier altså at vi på forhånd forventer at selvom all vår energi til å begynne med er samlet på lave bølgetall (lange bølger), så vil de ikke-linære vekselvirkningene en eller annen gang gi oss variabilitet (eller energi) på bølger med høye bølgetall (korte bølger).

For å belyse dette teoretisk skal vi se på en modell hvor farten er gitt ved likningen

$$\partial_t u + u\partial_x u = 0, \tag{5.17}$$

altså en ikke lineær-adveksjonslikning. La oss som *Richtmyer* (1963) bruke leapfrogskjemaet siden vi vet at dette er stabilt i det lineære tilfellet, that is

$$u_j^{n+1} = u_j^{n-1} - \frac{\lambda}{2}\left[(u_{j+1}^n)^2 - (u_{j-1}^n)^2\right] \tag{5.18}$$

hvor $\lambda = \Delta t/\Delta x$. Gjør vi nå bruk av (5.16) og antar at middelverdien av u er null ($a_0 = 0$) kan vi formelt skrive løsningen av (5.18) på formen

$$u_j^n = C_n \cos(\frac{\pi j}{2}) + S_n \sin(\frac{\pi j}{2}) + U_n \cos(\pi j) + V, \tag{5.19}$$

hvor amplitudene C_n, S_n er assosiert med bølgelengden $4\Delta x$, amplituden U_n med bølgelengden $2\Delta x$, og V assosieres med de resterende bølger med lavere bølgetall ($\alpha < \pi/2\Delta x$). Setter vi nå (5.19) inn i (5.18) og bemerker at $(u_{j+1}^n)^2 - (u_{j-1}^n)^2 = (u_{j+1}^n - u_{j-1}^n)(u_{j+1}^n + u_{j-1}^n)$ får vi at amplitudene må forholde seg til hverandre som følger:

$$\begin{aligned} C_{n+1} - C_{n-1} &= 2\lambda S_n(U_n - V) \\ S_{n+1} - S_{n-1} &= 2\lambda C_n(U_n + V) \\ U_{n+1} &= U_{n-1}. \end{aligned} \tag{5.20}$$

Den siste likningen i (5.20) sier at U_n beholder sine initialverdier, men også at den kan ha forskjellige initialverdier for ulike ($n = 1, 3, 5, \ldots$) og like ($n = 2, 4, 6, \ldots$) tidskritt, for instance $U_{2m} = A$ og $U_{2m-1} = B$ for $m = 1, 2, 3, \ldots$. Ved å eliminere S_n fra den første likningen i (5.20) ved hjelp av de to andre får vi

$$C_{n+2} - 2C_n + C_{n-2} = 4\lambda^2(A + V)(B - V)C_n. \tag{5.21}$$

For å være en stabil løsning må amplitudefaktoren for $4\Delta x$ bølgen og $2\Delta x$ bølgene være (numerisk) stabile. For å sikre dette må tallverdien av vekstfaktoren for amplitudene være mindre eller lik 1. For å undersøke under hvilke betingelser dette er tilfellet gjør vi som før og definerer vekstfaktoren ved $G \equiv C_{n+2}/C_n$. Innsatt i (5.21) gir det likningen

$$G^2 - 2\gamma G + 1 = 0 \tag{5.22}$$

til bestemmelse av G, hvor γ er et reelt tall gitt ved

$$\gamma = 1 + 2\lambda^2(A+V)(B-V). \tag{5.23}$$

Likning (5.22) har røttene

$$G_{1,2} = \gamma \pm i\sqrt{1-\gamma^2}. \tag{5.24}$$

Så lenge radikanden er reell vil derfor

$$|G_{1,2}| = \sqrt{\gamma^2 + 1 - \gamma^2} \equiv 1 \tag{5.25}$$

Bølgen $4\Delta x$ er derfor nøytralt, betinget stabil under betingelsen

$$1 - \gamma^2 \geq 0, \text{ eller } -1 \leq \gamma \leq 1. \tag{5.26}$$

Som vi ser av (5.23) er dette bare mulig dersom initialverdien av amplityden til $2\Delta x$ bølgen er slik at $|A| < V$ og/eller $|B| < V$. Med andre ord vil $4\Delta x$ bølgen være instabil dersom energien i bølgen med bølgelengde $2\Delta x$ er for stor i forhold til energien i den delen som består av lange bølger.

I numeriske modeller for hav og atmosfære som beskriver havets og atmosfærens ikke-lineære oppførsel er det derfor svært viktig å dempe eller fjerne energien i de korteste bølgene, eller "støyen", slik som beskrevet i Kap. 2.2 og Kap. 3.6.

5.3 The advection-diffusion problem

I kapitlene 3 og 4 lærte vi at diffusjonslikningen var ustabil for et skjema som var sentrert i rom og tid (leapfrog-skjemaet) , mens adveksjonslikningen var stabilt for det samme skjemaet. Samtidig lærte vi at adveksjonslikningen var ustabil om vi brukte et skjema som var forlengs i tid og sentrert i rom, mens dette skjemaet gjorde diffusjonslikningen stabil.

Som nevnt i Kapittel 2 vil de fleste atmosfæriske og oseanografiske problem innholde både adveksjon og diffusjon i en og samme likning. Vi er derfor nødt til kombinere flere skjemaer og spørsmålet blir da hvordan vi skal konstruere et skjema som da er stabilt.

For å se på dette skal vi i det videre anta at vi arbeider i en dimensjon. Den kontinuerlige form av adveksjons-diffusjonsproblemet som gitt i (2.1) med den advektive fluksen gitt ved (2.2) og den diffusive fluksen ved (2.3) får vi ved å kombinere (3.1) og (4.1), that is

$$\partial_t \theta + u\partial_x \theta = \kappa \partial_x^2 \theta. \tag{5.27}$$

Vi kan for instance bruke skjemaet

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

Merk at vi har evaluert diffusjonsdelen ved tiden $(n-1)\Delta t$ og adveksjonsdelen ved tiden $n\Delta t$. Dette gjør at skjemaet er forlengs i tid med tidskritt $2\Delta t$ for diffusjonsdelen og sentrert i tid for adveksjonsdelen. Hver del er derfor stabilt for seg; dersom $\kappa = 0$ er skjemaet stabilt overfor betingelsen $C \equiv u\frac{\Delta t}{\Delta x} \leq 1$ og dersom $u = 0$ er skjemaet stabilt for $\kappa\frac{\Delta t}{\Delta x^2} \leq \frac{1}{4}$. Faktoren $\frac{1}{4}$ fremkommer fordi vi har brukt tidskrittet $2\Delta t$ for diffusjonsdelen.

Når vi kombinerer skjemaene får vi en modifisert stabilitetsbetingelse. Betingelsen finner vi ved å benytte von Neumanns analysemetode. Likningen for vekstfaktoren er (se Oppgave 1 på slutten av dette kapittelet)

$$G^2 + 2i\lambda_1 G - \lambda_2 = 0 \tag{5.29}$$

hvor

$$\lambda_1 = \frac{u\Delta t}{\Delta x}\sin(\alpha\Delta x) \text{ og } \lambda_2 = 1 + 4\frac{\kappa\Delta t}{\Delta x^2}(\cos\alpha\Delta x - 1) \tag{5.30}$$

er to reelle tall. De to løsningene for vekstfaktoren er derfor

$$G_{1,2} = -i\lambda_1 \pm \sqrt{\lambda_2 - \lambda_1^2}. \tag{5.31}$$

Siden $\lambda_2 \leq 1$ følger at så lenge radikanden i (5.31) er reell er skjemaet (5.28) alltid stabilt. Den tilstrekkelige betingelsen for stabilitet av skjemaet (5.28) er derfor

$$\frac{(u\Delta t)^2 + 4\kappa\Delta t}{\Delta x^2} \leq 1. \tag{5.32}$$

Vi ser at for respectively $u = 0$ og $\kappa = 0$ sitter vi igjen med de vanlige betingelsene for stabilitet av respectively diffusjonslikningen og adveksjonslikningen. Men det (5.32) viser oss er at ved å kombinere de to får vi en betingelse som er strengere enn de individuelle kravene. Dette er visualisert i Fig. (5.1) .

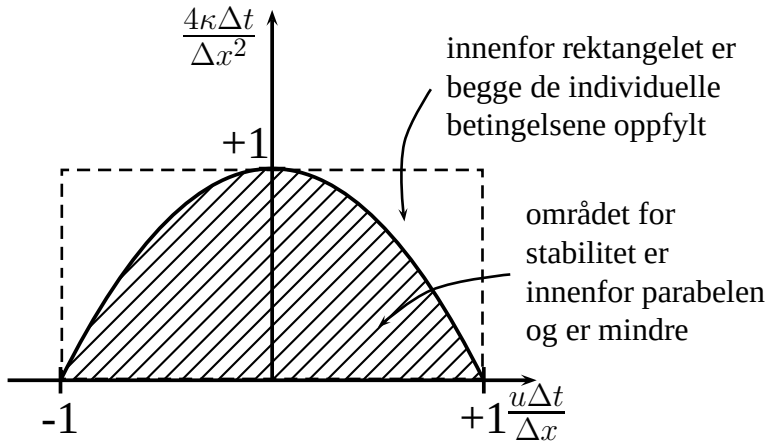
Vi får det noe overraskende resultat at det å legge til diffusjon, som fysisk virker dempende på støy i systemet, faktisk reduserer det maksimale tidskrittet vi kan bruke for for instance adveksjon. Dette er likevel ikke et alvorlig problem i oseanografi idet vi vanligvis har

$$\frac{\kappa\Delta t}{\Delta x^2} \ll \frac{u\Delta t}{\Delta x}, \quad u > 0 \tag{5.33}$$

Tidligere i Kap. 3.6 nevnte vi at siden diffusjon vanligvis legges på for å dempe eller forhindre ikke-lineære instabiliteter (se også Kap. 5.2) kan det være hensiktmessig å bruke et Dufort-Frankel skjema for diffusjonsleddet. Gjør vi dette i (5.27) endres skjemaet (5.28) til

$$\theta_j^{n+1} = \theta_j^{n-1} - \frac{u\Delta t}{\Delta x}(\theta_{j+1}^n - \theta_{j-1}^n) + \frac{2\kappa\Delta t}{\Delta x^2}(\theta_{j+1}^n - \theta_j^{n+1} - \theta_j^{n-1} + \theta_{j-1}^n). \tag{5.34}$$

Her har vi kombinert et konsistent, betinget stabilt skjema for adveksjonsdelen (leapfrog) med et ikke-konsistent, betingelsesløst stabilt skjema for diffusjonsdelen. I det en dimensjonale tilfellet



Figur 5.1: Visulisering av område hvor skjemaet til løsning av det kombinerte adveksjons-diffusjonsproblemet er stabilt. Området innenfor det stiplete rektangelet er der hvor adveksjons-delen og diffusjonsdelen er stabile hver for seg. Området innenfor parabelen (skravert) er der hvor skjemaet (5.28) er stabilt. Vi ser altså at ved å kombinere et sentrert skjema i tid og rom for adveksjon med et forlengs i tid, sentrert i rom skjema for diffusjon får vi en strengere betingelse for stabilitet.

her kan det vises at det er stabilt så lenge CFL kriteriet for adveksjonsdelen er oppfylt, men i det mer generelle tilfellet med flere dimensjoner blir kriteriet litt strengere (*Cushman-Roisin*, 1984).

Mange forfattere (se for instance *Clancy*, 1981) argumenterer for å bruke et forlengs i tid og sentrert i rom for både adveksjons- og diffusjonsdelen (FTCS) i (5.27). Da blir skjemaet slik:

$$\theta_j^{n+1} = \theta_j^n - \frac{u\Delta t}{2\Delta x}(\theta_{j+1}^n - \theta_{j-1}^n) + \frac{\kappa\Delta t}{\Delta x^2}(\theta_{j+1}^n - 2\theta_j^n + \theta_{j-1}^n). \tag{5.35}$$

Vekstfaktoren følger da liknignen

$$G = 1 - i\frac{u\Delta t}{2\Delta x}\sin\alpha\Delta x + 2\frac{\kappa\Delta t}{\Delta x^2}(\cos\alpha\Delta x - 1). \tag{5.36}$$

Som vist i *Clancy* (1981) er da skjemaet stabilt dersom de to betingelsene

$$\frac{\kappa\Delta t}{\Delta x^2} \leq 1, \text{ og } \frac{|u|\Delta t}{\kappa} \leq 1 \tag{5.37}$$

begge er oppfylt. Til tross for at FTCS skjemaet støttes entusiastisk av mange forfattere, anbefaler vi det likevel ikke fremfor de mer konservative skjemaene over.

5.4 The shallow water equations

The shallow water equations describe motions in a shallow ocean or a barotropic atmosphere (density and pressure surfaces are concurrent). It may also be shown that the full three-dimensional equations of motion may be written as a sum of vertical modes which individually can be modeled with a set of shallow water equations. If a numerical model is discretizised into say N vertical levels one will have one set of equations for each vertical level of the model, i.e. N set of equations. Each of these has a so called "equivalent depth" corresponding roughly to the height of the coordinate surface above ground/bottom.

One formulation of the shallow water equations are,

$$\partial_t u + u\partial_x u + v\partial_y u - fv + \partial_x \phi = 0 \tag{5.38}$$

$$\partial_t v + u\partial_x v + v\partial_y v + fu + \partial_y \phi = 0 \tag{5.39}$$

$$\partial_t \phi + u\partial_x \phi + v\partial_y \phi + \phi(\partial_x u + \partial_y v) = 0 \tag{5.40}$$

We will study what kind of motions these equations support. We will assume that the dependent variables may be written as a basic state plus a perturbation, i.e. $u = U + u'$, $v = V + v'$, $\phi = \Phi + \phi'$. The equations are linearized around the basic state with velocity $\mathbf{V} = U\mathbf{i} + V\mathbf{j}$ and a geopotential height Φ . We then obtain,

$$\partial_t u = -U\partial_x u - V\partial_y u + fv - \partial_x \phi \tag{5.41}$$

$$\partial_t v = -U\partial_x v - V\partial_y v - fv + \partial_y \phi \tag{5.42}$$

$$\partial_t \phi = -U\partial_x u + V\partial_y u + \Phi(\partial_x u + \partial_y v) \tag{5.43}$$

where we have dropped the primes and u, v and ϕ are the perturbed quantities. We will further assume that U, V and Φ are constants.

To proceed we will assume a Fourier solution

$$\mathbf{h} = \mathbf{h}_0 e^{-i\omega t} e^{i(\alpha x + \beta y)} \tag{5.44}$$

where α and β are wave numbers in the x- and y-direction, respectively, and $\mathbf{h}$ denotes the vector

$$\mathbf{h} = \begin{bmatrix} u \\ v \\ \phi \end{bmatrix}. \tag{5.45}$$

Insertion into the linearized equations give the following homogeneous linear equation,

$$\boldsymbol{\mathcal{A}} \cdot \mathbf{h} = 0 \tag{5.46}$$

where the tensor $\boldsymbol{\mathcal{A}}$ is given by

$$\boldsymbol{\mathcal{A}} = \begin{bmatrix} -i(\alpha U + \beta V) + i\omega & f & -i\alpha \\ -f & -i(\alpha U + \beta V) + i\omega & -i\beta \\ -i\alpha\Phi & -i\beta\Phi & -i(\alpha U + \beta V) + i\omega \end{bmatrix}. \tag{5.47}$$

For non-trivial solutions to exists, the determinant of the tensor $\boldsymbol{\mathcal{A}}$ must be zero giving three solutions for the frequency ω, namely

$$\omega_1 = U\alpha + V\beta \tag{5.48}$$
$$\omega_2 = U\alpha + V\beta + \sqrt{\Phi(\alpha^2+\beta^2)+f^2} \tag{5.49}$$
$$\omega_3 = U\alpha + V\beta - \sqrt{\Phi(\alpha^2+\beta^2)+f^2} \tag{5.50}$$

The first solution is simply the geostrophic balance. This is easily seen by for instance studying the equation for u, i.e.,

$$\partial_t u = -U\partial_x u - V\partial_y u + fv - \partial_x\phi \tag{5.51}$$

Introducing the Fourier solution (5.44) we obtain

$$-i\omega u = -i\alpha U u - i\beta V u + fv - i\alpha\phi. \tag{5.52}$$

Substituing for ω_1 from (5.48) then gives

$$-i(\alpha U + \beta V)u = -i(\alpha U + \beta V)u + fv - ia\phi \tag{5.53}$$

or

$$v = \frac{1}{f} i\alpha\phi \Longleftrightarrow v = \frac{1}{f}\partial_x\phi, \tag{5.54}$$

i.e. in geostrophic balance

The two other solutions are combined inertia (the frequency ω is proportional to f) and gravity waves (the frequency goes as $\sqrt{\Phi(\alpha^2+\beta^2)}$). Recalling that $w = \sqrt{(}\alpha^2+\beta^2)c$ where c is the phase speed, we see that c is given by $c = \sqrt{gH}$, i.e. gravity waves. The full solution to the linear problem is a sum over all possible solutions for all wave numbers α and β.

If these equations are solved numerically with horizontal derivative replaced by finite differences, we must replace α by $\sin(\alpha\Delta x)/(\alpha\Delta x)$ and β by $\sin(\beta\Delta y)/(\beta\Delta y)$. Normally the inertia-gravity mode has a much higher frequency than the Rossby mode $\Phi^2 \gg U^2 + V^2$, and these modes becomes unstable first. For simplicity we therefore put $U = V = 0$.

Thus

$$|\omega|\Delta t = \Delta t\sqrt{\Phi\left[\frac{\sin(\alpha\Delta x)}{\alpha\Delta x} + \frac{\sin(\beta\Delta y)}{\beta\Delta y}\right] + f^2} \tag{5.55}$$

and the CFL criterium then becomes,

$$\sqrt{2\Phi + f^2\Delta x^2}\frac{\Delta t}{\Delta x} < 1 \tag{5.56}$$

Since the first term dominates it follows that in practice

$$\sqrt{2\Phi}\frac{\Delta t}{\Delta x} < 1 \quad \text{or} \quad \Delta t < \frac{\Delta x}{\sqrt{2\Phi}} \tag{5.57}$$

where the factor $\sqrt{2}$ stems from the two-dimensionality of the problem.

In an atmospheric model the largest equivalent depth is approximately 10000m 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 equilibrium geopotential height $\Phi = gH$, where $H = 4$km, thus giving limiting the time steps to seconds[2]. In contrast to the atmosphere the inertia-gravity waves contains tidal motion and the storm surge signal. Thus we are restricted to such limitations on the time step if we want to simulate these important oceanic features.

5.5 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 stronger (shorter time-step). It is therefore tempting to treat terms responsible for this behaviour implicitly while we treat other terms explicitly, the so- called semi-implicit method.

For clarity we write

$$\partial_t u = A_u - \partial_x \phi, \tag{5.58}$$
$$\partial_t v = A_v - \partial_y \phi, \tag{5.59}$$
$$\partial_t \phi = A_\phi - \Phi(\partial_x u + \partial_y v), \tag{5.60}$$

where A_u, A_v and A_Φ include advection and Coriolis terms on which we will use the leapfrog method. The remaining terms will be integrated implicitly. We then get the following finite difference eauations,

$$\frac{u^{n+1} - u^{n-1}}{2\Delta t} = [A_u]^n - [\partial_x \phi]^{n+1}, \tag{5.61}$$
$$\frac{v^{n+1} - v^{n-1}}{2\Delta t} = [A_v]^n - [\partial_y \phi]^{n+1}, \tag{5.62}$$
$$\frac{\phi^{n+1} - \phi^{n-1}}{2\Delta t} = [A_\phi]^n - \Phi\left([\partial_x u]^{n+1} + [\partial_y v]^{n+1}\right), \tag{5.63}$$

To proceed, the first two equations are solved with respect to u_{jk}^{n+1} and v_{jk}^{n+1} respectively giving,

$$u^{n+1} = u^{n-1} + 2\Delta t[A_u]^n - 2\Delta t\,[\partial_x \phi]^{n+1} \tag{5.64}$$
$$v^{n+1} = v^{n-1} + 2\Delta t[A_v]^n - 2\Delta t\,[\partial_y \phi]^{n+1} \tag{5.65}$$

derivation with respect to x and y respectively and inserting into the equation for ϕ gives a Helmholz equation,

$$\Phi \Delta t^2 \left([\partial_x^2 \phi]^{n+1} + [\partial_y^2 \phi]^{n+1}\right) - \phi^{n+1} = B \tag{5.66}$$

[2]For $H = 4000$m $g = 10$ms^2 and grid size $\Delta x = 20000$m follows that $\Delta t < 141$s

where B contains known quantities at time levels $n, n-1, \ldots$

With proper boundary conditions (ϕ or its normal derivative at lateral boundaries), these equations may easily be solved by standard numerical methods. Having obtained ϕ^{n+1}, we easily find u^{n+1} and v^{n+1}. This method is widely used in atmospheric models as it is possible to use longer time-steps. We do not have to take the gravity mode speed $\sqrt{gH}$ into account when estimating an upper bound for the time step. Note that we cannot do this in the ocean because then the inertia-gravity waves constins the signal as alluded to above.

5.6 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 (n) must be estimated by interpolation from the surrounding grid point values. The new value at the grid points at time (n+1) due to advection are then simply the value at the departure point at time (n). 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 calculate 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.

Exercises

1. Use von Neumanns method (Kap. 3.3) to show that the expression (5.29) is indeed the correct expression for the growth factor when using the scheme given in (5.28).

2. Show that the condition (5.32) is a sufficient condition for numerical stability.

Kapittel 6

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 latter are 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.

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 verical coordinate.

6.1 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

$$x' = x, \quad y' = y, \quad s = s(x, y, z, t), \quad \text{and} \quad t' = t, \tag{6.1}$$

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

$$\partial_z s \gtrless 0, \quad \text{and } \partial_z s \neq 0. \tag{6.2}$$

This is also a necessary condition to ensure that also the inverse transformation $z = z(x', y', s, t')$ exists.

Let ξ' denote any of the three independent variables x', y', and t' in the new coordinate system, and similarly let ξ denote any of the three independent variables x, y, and t in the original system. Then the transformation (6.1) gives

$$\partial_z \xi' = 0, \quad \partial_t x' = \partial_t y' = 0, \quad \partial_y x' = \partial_x y' = 0, \quad \text{and} \quad \partial_x t' = \partial_y t' = 0, \tag{6.3}$$

while

$$\partial_x x' = \partial_y y' = \partial_t t' = 1. \tag{6.4}$$

Likewise follows

$$\partial_s \xi = 0, \quad \partial_{t'} x = \partial_{t'} y = 0, \quad \partial_{y'} x = \partial_{x'} y = 0, \quad \text{and} \quad \partial_{x'} t = \partial_{y'} t = 0, \tag{6.5}$$

while

$$\partial_{x'} x = \partial_{y'} y = \partial_{t'} t = 1. \tag{6.6}$$

Det understrekes her at siden s er monoton med hensyn på z vil $\partial_z s \neq 0$ og likeledes $\partial_s z \neq 0$. Vi legger også merke til at dersom vi transformerer fra z til seg selv, dvs $s = z$ vil $\partial_z s = \partial_s z = 1$.

La nå $\psi = \psi(x, y, z, t) = \psi(x', y', s, t')$ betegne en hvilkensomhelst skalar. Den første egenskapen ved transformasjonen er at

$$\partial_z \psi = \partial_z s \partial_s \psi. \tag{6.7}$$

Hvis vi nå deriverer ψ med hensyn på en av de uavhengige variable i det koordinatsystemet vi transformerer til, for instance t', får vi

$$\partial_{t'} \psi = \partial_t \psi \partial_{t'} t + \partial_x \psi \partial_{t'} x + \partial_y \psi \partial_{t'} y + \partial_z \psi \partial_{t'} z = \partial_t \psi + \partial_z s \partial_s \psi \partial_{t'} z, \tag{6.8}$$

hvor det siste likhetstegnet følger ved å anvende (6.3) - (6.7). Vi kan også løse (6.8) mhp $\partial_t \psi$. Da får vi

$$\partial_t \psi = \partial_{t'} \psi - \partial_z s \partial_s \psi \partial_{t'} z, \tag{6.9}$$

Tilsvarende følger også at

$$\partial_x \psi = \partial_{x'} \psi - \partial_z s \partial_s \psi \partial_{x'} z, \text{ og } \partial_y \psi = \partial_{y'} \psi - \partial_z s \partial_s \psi \partial_{y'} z. \tag{6.10}$$

La oss nå definere den horisontale gradienten til ψ i det nye koordinatsystemet med

$$\nabla_s \psi = \mathbf{i} \partial_{x'} \psi + \mathbf{j} \partial_{y'} \psi. \tag{6.11}$$

Ved bruk av (6.9) og (6.10) får vi da at

$$\nabla_H \psi = \nabla_s \psi - \partial_z s \partial_s \psi \nabla_s z. \tag{6.12}$$

Videre finner vi at den horisontale divergensen til en hvilkensomhelst vektor, for instance $\mathbf{a}$ transformerer som

$$\nabla_H \cdot \mathbf{a} = \nabla_s \cdot \mathbf{a} + \partial_s \partial_s \mathbf{a} \cdot \nabla_s z \tag{6.13}$$

Merk at alle vektorer projiseres inn i horisontalplanet. Dette gjelder også gradienten i (6.12). Dette gjør at metriske termer assosiert med vertikalgradienten til flaten s i et geopotensielt koordinatsystem blir eliminert.

Som kjent er den individuelt eller materielt deriverte, for instance $D\psi/dt$ (på engelsk 'material derivative') uavhengig av koordinattransformasjoner. I geopotensielle koordinater er denne utskrevet

$$\frac{D\psi}{dt} = \partial_t \psi + \mathbf{u} \cdot \nabla_H \psi + w \partial_z \psi, \tag{6.14}$$

mens den i det nye generelle vertikale koordinatsystemet er

$$\frac{D\psi}{dt} = \partial_{t'} \psi + \mathbf{u} \cdot \nabla_s \psi + \frac{Ds}{dt} \partial_s \psi, \tag{6.15}$$

hvor Ds/dt er endringen av flaten s i retning av den tredimensjonale hastigheten. Dersom vi nå gjør bruk av (6.9) - (6.12) til å erstatte leddene i (6.14) får vi

$$\frac{D\psi}{dt} = \partial_{t'} \psi + \mathbf{u} \cdot \nabla_s \psi + (w - \partial_{t'} z - \mathbf{u} \cdot \nabla_s z) \partial_z s \partial_s \psi. \tag{6.16}$$

For å repetere så er den indivduelt deriverte invariant overfor koordinat-tranformasjoner. Dermed må høyre siden i (6.15) være lik høyresiden i (6.16), eller

$$\frac{Ds}{dt} = (w - \partial_{t'} z - \mathbf{u} \cdot \nabla_s z) \partial_z s \equiv \omega \partial_z s. \tag{6.17}$$

Den siste identiteten definere hastigheten ω,

$$\omega = w - (\partial_{t'} z + \mathbf{u} \cdot \nabla_s z) = w - (\partial_t s + \mathbf{u} \cdot \nabla_H s) \partial_s z. \tag{6.18}$$

Herav ser vi at dersom $s = z$ vil $\omega = w$. I dette tilfellet er altså ω hastigheten gjennom de faste nivåflatene i et geopotensielt koordinatsystem. Er s en materiell flate[1] er den kinematiske grenseflatebetingelsen $w = (\partial_t s + \mathbf{u} \cdot \nabla_H s)$. I det tilfellet er altså $\omega = 0$, hvilket er i tråd med definisjonen på en materiell flate. For enhver ikke materiell flate vil $\omega \neq 0$ og beskrive hastigheten til partiklene gjennom den ikke materielle flaten s. Forskjellen mellom ω og w er altså flatens egenhastighet i det geopotensielle koordinatsystemet, mens w selv som før beskriver den totale vertikale hastigheten i det samme koordinatsystemet.

6.2 Transformation of the governing equations

6.2.1 The hydrostatic equation

Ser vi nå først på den hydrostatiske likningen

$$\partial_z p + \rho g = 0 \tag{6.19}$$

[1]En materiell flate er en flate hvor ingen materielle partikler får lov å passere gjennom

er denne transformert

$$\partial_s p + \rho g \partial_s z = 0. \tag{6.20}$$

Denne likningen kan for instance brukes til å erstatte de metriske faktorene $\partial_s z$ og $\partial_z s$ som følger

$$\partial_s z = -\frac{\partial_s p}{\rho g}, \text{ og } \partial_z s = -\frac{\rho g}{\partial_s p} \tag{6.21}$$

6.2.2 Mass conservation

La oss så se på transformeringen av massebevaringslikningen

$$\partial_t \rho + \nabla \cdot (\mathbf{v}\rho) = 0. \tag{6.22}$$

Innsetting av divergensen fra (6.13) og for den vertikalintegrerte fra (6.7) og bruk av (6.18) gir ganske enkelt at

$$\partial_{t'}(\partial_s z) + \nabla_s \cdot (\mathbf{u}\partial_s z) + \partial_s \omega = 0. \tag{6.23}$$

6.2.3 The momentum equation

Bevaring av bevegelsesmengde er litt mer komplisert

6.3 Terrain-following coordinates

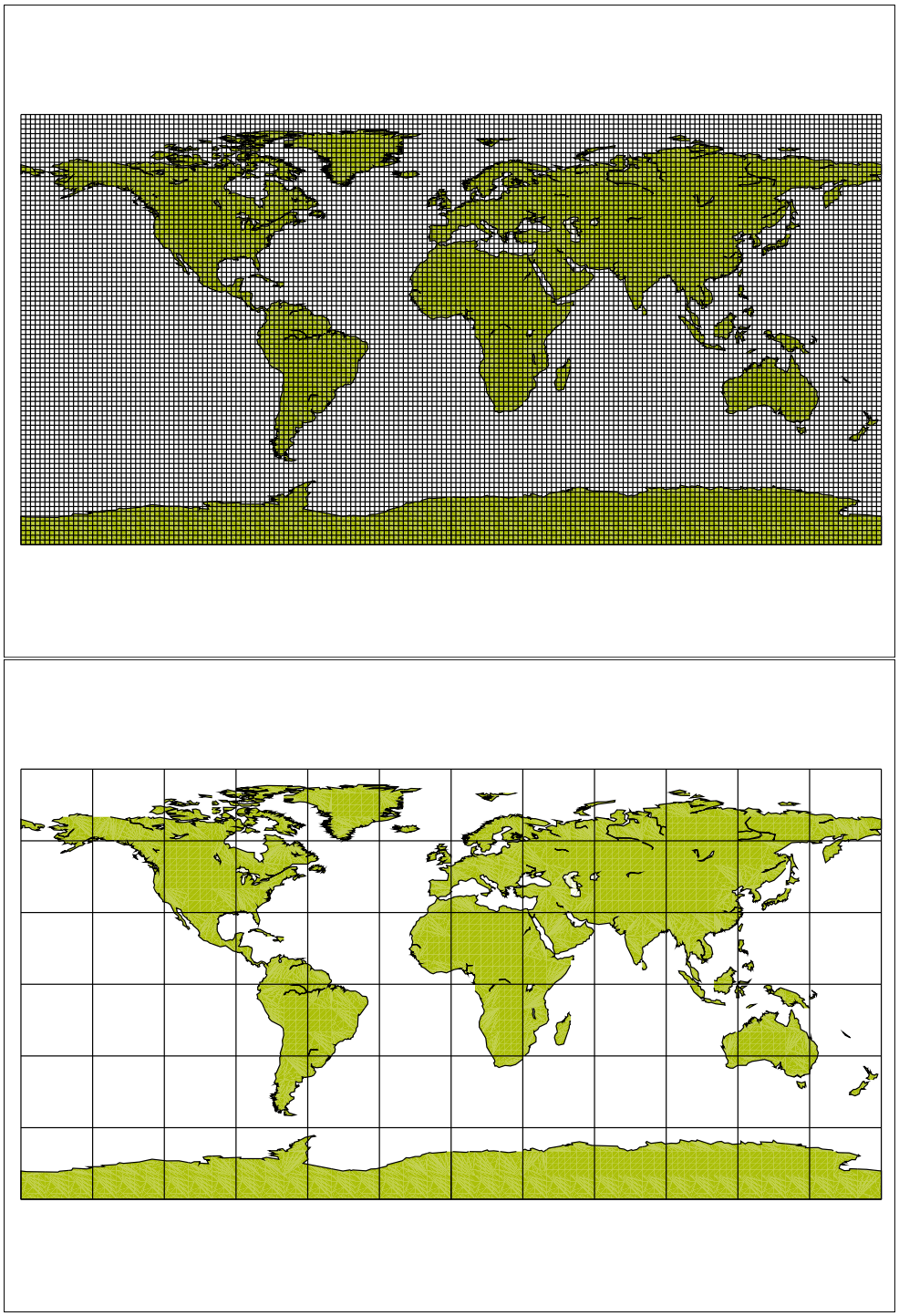
Kapittel 7

OPEN BOUNDARY CONDITIONS

As we all know, computers can only hold a finite number of numbers in their *random access memory* (RAM). Thus even the biggest computers can only hold a finite number of variables in their memory at any time. This is one of the main reason why numerical oceanography is less mature than numerical meteorology. Recall that in the atmosphere the typical lenght scale of a synoptic low at subpolar latitudes is about 500 - 1000 km, while the typical time scale is a few days. In contrast the length scale of a low in the ocean is about 10 - 100 km while the time scale is a few weeks to months. 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 7.1 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 7.1 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 since the mesh size is decreased so is the time step needed 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. So to enable computers to provide numerical ocean weather forecasts as fast as todays NWP models for the same area we 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 correponds to an ocean forecasts of say 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 as at the European Centre for Medium-range Weather Forecasts (ECMWF). However, there are still local effects, noteably processes associated with irregular topography, that is not yet resolved properly by the global model. Thus most national institutes providing public meteorological



Figur 7.1: 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.

services run a limited area model which is “nested” into a global model[1]. 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 the baoundary these common boundaries 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 dynamically consistent downscaling of the coarser solution to a scale that take 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 are not yet feasible at all on todays computers. Thus the global ocean models that exist does *not* resolve the oceanic weather. 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 ocean weather predictions we therefore have to make use of the nesting techniques, and that is what is done today. Such forecasts, updated daily, for Norwegian waters are for instance provided by the Norwegian Meteorological Institute to the public at the website: http://met.no/kyst_og_hav/havvarsel.html.

For å gjøre ting verre med tanke på havvarslingsmodeller kommer det faktum i tillegg at tidsskalaen i havet er mye lengre enn i atmosfæren. Mens den synoptiske meteorologi har en tidsskala på en til tre dager, er den tilsvarende i havet på noen dager til opp i mot en måned og kanskje lenger. Disse forhold har som konsekvens at havmodeller med samme geografiske dekning som en tilsvarende modell for atmosfæren stiller mye større krav til en regnemaskin. Konkret gir dette seg utslag i at en oseanografisk modell krever mye mer av maskinens minnekapasitet (RAM - Random Access Memory). Samtidig krever den forholdsvis korte gitteravstanden at et mye kortere tidsskritt må anvendes (på grunn av kravet til numerisk stabilitet) hvilket betyr at en havvarslingsmodell vil bruke uforholdsmessig mye lenger tid på et 60 timers varsel enn en tilsvarende atmosfærisk modell som dekker samme geografiske område.

Konsekvensen er at dersom man ønsker å konstruere en global havvarslingsmodell som oppløser den synoptiske skalaen vil selv en forholdsvis enkel modell stille så store krav til maskinkapasitet at en løsning ikke er mulig. Det geografiske området for dagens synoptiske havvarslingsmodeller[2] må derfor begrenses til et regionalt område hvilket medfører at modellen har åpne render, that is, render som ender i det åpne havet og ikke langs en kyst eller annen form for naturlig rand (for en definisjon se kap. 7.1). Også i atmosfæren har vi åpne render. For eksempel er den øvre vertikale grenseflate egentlig “uendelig” høyt oppe. På en regnemaskin må imidlertid beregningene avsluttes for en endelig høyde over bakken. En annen grunn til at åpne render er av betydning i atmosfære og hav er at man ofte benytter såkalt nøsting som teknikk for å nedskalere fra for instance en global modell til en regional modell slik som det gjøres til daglig i værvarsling for for instance Norge. På tvers av disse rendene kan, i det virkelige hav og atmosfære, væske fritt strømme igjennom, og bølger forplante seg uhindret. Åpne render er altså ikke materielle

[1] At the Norwegian Meteorological Insitute the limited area model at the time of writing is HIRLAM - the High Resolution Limited Area Model

[2] se for instance http://met.no/kyst_og_hav/havvarsel.html

grenseflater. Når vi avslutter våre beregninger ved slike åpne render krever våre likninger at det settes betingelser eller randkrav også her. Som for mange andre matematiske problem vil de betingelsene vi pålegger på en åpen rand være med på å bestemme løsningen innenfor randen, altså i det området hvor vi er interessert i løsningen.

I den numeriske værvarslings barndom strevet også meteorologene med laterale åpne render, slik som oseanografene gjør i dag. Allerede i slutten av 40 årene under forsøk på en numerisk integrasjon av den barotrope virvellikningen støtte J. Charney, R. Fjørtoft og von Neumann på dette problemet (se *Charney et al.*, 1950). På den tiden var regnemaskinene "små" og de var henvist til å løse virvlingslikningen for et begrenset område. Derfor var de også nødt til å ta problemet med åpne render på alvor. De løste problemet ved å spesifisere virvlingen langs de rendene der det strømmet inn, mens de brukte en form for gradient betingelse (se kap 7.2) langs de rendene det strømmet ut. Som det ble vist av *Platzman* (1954), oppfylte ikke denne type randbetingelse de krav som må settes til en slik betingelse. Løsningen til *Charney et al.* (1950) viste seg å være instabil. Problem forbundet med åpne render og fastsettelse av åpne randbetingelser er altså ikke nye innen væskedynamikk.

7.1 What do we require from an OBC

Røed and Cooper (1986) har gitt følgende definisjon på en åpen rand:

> *En åpen rand er en kunstig rand med den egenskap at forstyrrelser skapt innenfor randen, that is, beregningsområdet, uhindret kan forlate dette uten å forstyrre eller på annen måte ødelegge løsningen innenfor beregningsområdet.*

Dette medfører at vi kan (og må) stille visse krav til en betingelse som skal gjelde ved åpne render. Disse kravene er de kriterier vi etterpå vil bruke for å bedømme om vår anvendte betingelse er god eller dårlig. Før vi ser nærmere på disse kravene er det naturlig å legge til at også det motsatte bør være tilfelle ved en åpen rand, nemlig at informasjon om hendelser skapt utenfor vårt modell- eller beregningsområde må få bre seg uhindret inn i beregningsområdet. Problemet med det siste er at vi slett ikke alltid har slik informasjon.

Ett absolutt krav ligger allerede i definisjonen, that is, forstyrrelser (for instance Kelvin bølger) skapt innenfor beregningsområdet skal ikke forstyrre eller på annen måte ødelegge løsningen innenfor beregningsområdet. Altså skal for instance bølger skapt i det indre ikke reflekteres ved den åpne randen. Videre er det et absolutt krav at den anvendte betingelsen ikke skal føre til numeriske instabiliteter. Det er også naturlig å føye til noen formelle krav. For det første at skjemaet som brukes ved randen bør ha samme nøyaktighet som det indre, og for det andre at betingelsen bør sammen med de styrende likningene være et noenlunde velformulert matematisk problem, that is, ha en løsning som eksisterer og er noenlunde entydig.

En oppsummering av åpne grenseflatebetingelser finnes i *Chapman* (1985) og *Røed and Cooper* (1986, 1987). I det videre skal vi komme inn på noen av de mest brukte betingelsene, og se på deres svakheter og deres sterke sider. I det siste kapittelet er også omtalt en i det senere mye brukt metode kalt FRS (Flow Relaxation Scheme) (se for eksempel *Martinsen and Engedahl*, 1987; *Cooper and Thompson*, 1989). Det spesielle med FRS betingelsen er at den foruten å

oppfylle de fleste krav vi stiller til en åpen randbetingelse, også gir mulighet til å pådytte det indre informasjon om den ytre løsningen dersom vi har slik informasjon (for instance informasjon om tidevann). Vi skal i det videre gå inn på noen populære og velkjente åpne randbetingelser.

7.2 Radiation conditions

Strålingsbetingelser er en fellesbetegnelse på betingelser av typen

$$\partial_t \phi + c_\phi \partial_x \phi = 0 \tag{7.1}$$

Her representerer ϕ en av de avhengige variable, mens c_ϕ er x-komponenten av fasehastigheten tilhørende denne avhengige variable. Man antar med andre ord at de forstyrrelsene som er på vei ut er bølger, som brer seg med en viss fasehastighet. Ved sammenlikning av (7.1) med (4.1) gjenkjenner vi (7.1) som en adveksjonslikning med $\mathbf{F}_A = \mathbf{i} \cdot \mathbf{c}_\phi \phi$, hvor $\mathbf{c}_\phi$ er fasehastigheten. Fra Kap. 4.4, som omhandler karakteristikkmetoden, følger da at fasehastigheten gir helningen av karakteristikkene i faserommet. Det er altså bare helt spesielle fysiske problem som tilfredstiller en slik likning, nemlig bølger som beveger seg i fasehastighetens retning.

To opplagte spesialtilfeller av (7.1) har fått sitt eget navn; den første når fasehastigheten er null ($c_\phi = 0$) og den andre når fasehastigheten går mot uendelig ($c_\phi \to \infty$). Dersom vi setter fasehastigheten lik null (svarende til at karakteristikkene er vertikale i t, x rommet) gir (7.1) ved en tidsintegrasjon at,

$$\phi = \text{const.} \tag{7.2}$$

Dette kalles en *fastholdt betingelse* idet den avhengige variable, ϕ, ikke endrer seg med tiden.

Dersom fasehastigheten antar den andre ytterlighet ($c_\phi \to \infty$), følger av (7.1) at dersom $\partial_t \phi$ skal være endelig må gradienten på tvers av grenseflaten være null, eller,

$$\partial_x \phi = 0. \tag{7.3}$$

En slik betingelse (som svarer til at karakteristikkene er horisontale i faserommet) kalles en *gradient betingelse*.

Dersom fasehastigheten er endelig, men forskjellig fra null, har vi en ekte strålingsbetingelse. Problemet er da redusert til å bestemme fasehastigheten c_ϕ. Dersom vi vet at løsningen har en bølgeform idet den når den åpne randen, og dersom denne er kjent, kjenner vi også fasehastigheten. For en barotrop Kelvin bølge vet vi for instance at fasehastigheten er gitt ved,

$$c_\phi = \sqrt{gH} \tag{7.4}$$

hvor g er tyngdens akselerasjon og H vanndypet på stedet.

La oss som et eksempel se på likningene for bølger i en kanal med dybde H. La h betegne dybden av en vannkolonne og u vannkolonnens fart over bunnen (antas uavhengig av dybden idet vi ser bort ifra friksjon mot bunnen). La oss også anta at bevegelsen ikke er påvirket av jordrotasjonen eller andre krefter. Da kan de styrende likningene skrives

$$\partial_t u = -g \partial_x h \tag{7.5}$$

$$\partial_t h = -H \partial_x u \tag{7.6}$$

Tar vi nå og multipliserer likningen for h med en ukjent funksjon λ og legger til likningen for u får vi etter litt regning at

$$\partial_t u + \lambda H \partial_x u + \lambda \left(\partial_t h + \frac{g}{\lambda} \partial_x h \right) = 0 \tag{7.7}$$

La oss videre definere en differensial operator $\frac{D^*}{dt}$ slik at

$$\frac{D^*}{dt} = \partial_t + \frac{D^* x}{dt} \partial_x \tag{7.8}$$

that is, en individuelt derivert i den spesielle retningen $D^* x$ (jmf. Kap. 4.4). Kan vi nå finne en λ som tilfredstiller likningen

$$\frac{D^* x}{dt} = \lambda H = \frac{g}{\lambda} \tag{7.9}$$

kan likningen (7.7) skrives

$$\frac{D^* u}{dt} + \lambda \frac{D^* h}{dt} = 0 \tag{7.10}$$

Fra (7.9) følger at

$$\lambda_{1,2} = \pm \frac{c_0}{H}, \tag{7.11}$$

hvor $c_0 = \sqrt{gH}$ og dermed at (7.10) kan skrives

$$\left(\frac{D^*}{dt} \right)_{1,2} \left(u \pm c_0 \frac{h}{H} \right) = 0 \tag{7.12}$$

langs

$$\left(\frac{D^* x}{dt} \right)_{1,2} = \pm c_0. \tag{7.13}$$

Vi legger merke til at (7.12) og (7.13) egentlig er to likninger som utskrevet er

$$\partial_t \left(u + c_0 \frac{h}{H} \right) + c_0 \partial_x \left(u + c_0 \frac{h}{H} \right) = 0, \tag{7.14}$$

$$\partial_t \left(u - c_0 \frac{h}{H} \right) - c_0 \partial_x \left(u - c_0 \frac{h}{H} \right) = 0. \tag{7.15}$$

Som før kalles de to likningene i (7.12), that is, (7.14) og (7.15), for kompatibilitetslikningene. De to likningene i (7.13) er de to karakteristiske likninger. Den første likningen i (7.12) (med + tegn og operator nummer 1) tilsvarende (7.14), beskriver en bølge i positiv x-retning med fasehastighet c_0. Den andre, som tilsvarer (7.15), beskriver en bølge i negativ x-retning med fasehastighet $-c_0$. Likning (7.12) uttrykker at den spesifikke kombinasjonen av de avhengige variable u og h, nemlig $u \pm c_0 \frac{h}{H}$, er bevart langs karakteristikkene. Av disse likningene skjønner vi at løsningen av (7.5) er sammensatt av to bølger med fasehastighet hhv. $+c_0$ og $-c_0$.

La oss anta at vårt problem er å løse (7.5) for $0 < x < L$ og at rendene $x = 0$ og $x = L$ er åpne. La oss videre anta at en bevegelse er generert i det indre, for instance i form av en initiell

hevning av vannstanden lokalt. Hva er nå den rette randbetingelsen? Fra kompatibilitetslikningene (7.12) ser vi at informasjon om den initielle hevningen vil bre seg langs de to karakteristikkene gitt fra (7.13). Mot den høyre randen $x = L$ vil informasjonen bre seg langs $\frac{D^*x}{dt} = c_0$, og mot den venstre randen $x = 0$ langs $\frac{D^*x}{dt} = -c_0$. For å unngå refleksjon må vi nå sørge for at informasjon ikke kan bre seg tilbake til det indre. Ved $x = L$ svarer dette til at ingen karakteristikker her må helle innover, that is, her må

$$\left(\frac{D^*x}{dt}\right)_2 = 0 \tag{7.16}$$

Fra (7.15) følger at for $x = L$ er

$$\partial_t \left(u - c_0 \frac{h}{H}\right) = 0 \tag{7.17}$$

eller

$$u = c_0 \frac{h}{H} + \text{const.}, \quad \text{for } x = L \tag{7.18}$$

Dette er strålingsbetingelsen for et likningsett med to variable. Imidlertid ser vi fra (7.5) at ved å derivere den første med hensyn på t og den andre med hensyn på x får vi at u må tilfredstille bølgelikningen med fasehastigheten gitt ved

$$c_u = \sqrt{gH}. \tag{7.19}$$

Den naturlige randbetingelsen for u er da (7.1) med denne fasehastigheten. Dette er nøyaktig hva (7.18) uttrykker. Setter vi uttrykket for h fra (7.5) inn i (7.18) får vi nettopp (7.1) med $\phi = u$ og $c_\phi = c_u$. Dette utgangspunktet ble brukt av *Røed and Cooper* (1987) til å konstruere en svakt reflekterende åpen randbetingelse også for mer generelle problem basert på et arbeid av *Hedstrøm* (1979).

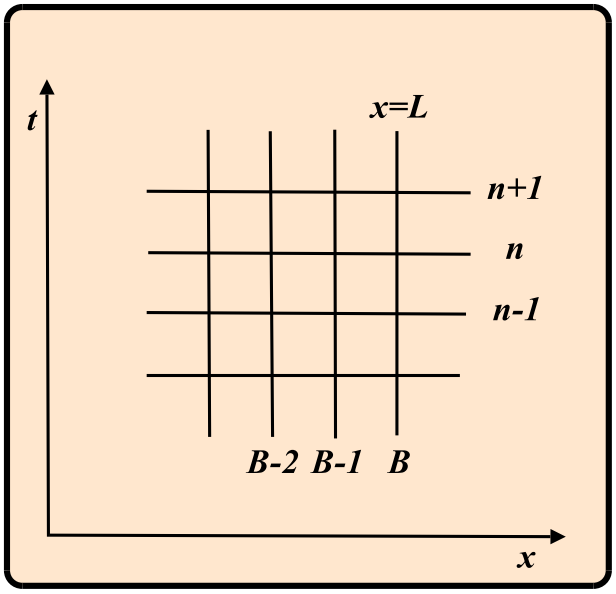
7.3 Implementation of the radiation condition

The radiation condition (7.1) is often used in combination with other conditions. Thus it is of interest to investigate its implementation in a numerical model. The following is essentially the implementation given in *Røed and Cooper* (1986).

To get started we assume that the computational domain is within $0 < x < L$ with $x = L$ as the open boundary. In addition we use subscript B to denote ϕ values at the open boundary. As before we will use superscript n to denote the time level, that is, $t^n = n\Delta t$ (cf. Fig. 7.2).

Siden (7.1) er en adveksjonslikning er det naturlig å bruke ett av skjemaene angitt i Kap. 4.1. Videre er det viktig at det skjemaet vi velger har samme nøyaktighet som det skjemaet vi har brukt i det indre. Her skal vi for enkelhetsskyld anta det er blitt brukt et skjema som er av første orden i tid og rom, for instance oppstrømsskjemaet. Da kan vi bruke det samme ensidige skjemaet i tid og rom for å oppdatere våre variable på randen, altså oppstrømsskjemaet. Vi antar derfor at $c_\phi \geq 0$ og følger notasjonen i Fig. 7.2, that is, diskretiserer (7.1) som følger

$$\frac{\phi_B^{n+1} - \phi_B^n}{\Delta t} + c_\phi \frac{\phi_B^n - \phi_{B-1}^n}{\Delta x} = 0 \tag{7.20}$$



Figur 7.2: 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 B, $B-1$, and $B-2$ denote grid points respectively at the open boundary, the first point inside the computational domain, etc, while n, $n-1$, etc. denotes the time levels.

eller om vi vil

$$\phi_B^{n+1} = (1 - r_\phi)\phi_B^n + r_\phi \phi_{B-1}^n \tag{7.21}$$

hvor

$$r_\phi = c_\phi \frac{\Delta t}{\Delta x}. \tag{7.22}$$

Likning (7.18) forteller oss at strålingsbetingelsen egentlig gir en ekstrapolasjon av verdiene av den variable ved tidligere tider mot randen. Men hva er fasehastigheten c_ϕ? Dersom denne må bestemmes må dette gjøres ved vårt kjennskap til løsningen for tidligere tider. Vi kan for instance, delvis som anbefalt av *Orlanski* (1976), løse (7.17) med hensyn på fasehastigheten (eller om vi vil r_ϕ). Men siden vi ikke kjenner verdien på randen ved tidspunktet t_{n+1} må vi gå ett tidsskritt tilbake i tid. Samtidig bør vi, som argumentert av *Røed and Cooper* (1987), også gå ett romskritt innover, idet vi ønsker å følge karakteristikker. Da følger fra (7.21) med $B \to B-1$ og $n \to n-1$ at en mulig fasehastighet er

$$r'_\phi = -\frac{\phi_{B-1}^n - \phi_{B-1}^{n-1}}{\phi_{B-1}^{n-1} - \phi_{B-2}^{n-1}} \tag{7.23}$$

Siden fasehastigheten er antatt å være større enn null må vi påse at så er tilfellet. Vi definerer derfor r_ϕ som følger

$$r_\phi = \begin{cases} r'_\phi & ; \quad 0 \le r'_\phi \\ 0 & ; \quad r'_\phi < 0 \end{cases} \tag{7.24}$$

som deretter brukes i (7.21) til å finne ϕ_B^{n+1}. Dersom vår åpne rand er til venstre (ikke til høyre som antatt her) vil ulikhetene i (7.24) måtte snus.

Som nevnt over er det bare det rene bølgeproblemet, that is, frie bølger uten påvirkning av friksjon, vind, eller jordrotasjon, som tilfredstiller likningen (7.1). I alle praktiske oseanografiske modeller vil jordrotasjonen og vindkreftene være tilstede og vil påvirke løsningen. Altså vil strålingsbetingelsene i mange sammenhenger være langt fra perfekte, og vil i de tilfelle hvor jordrotasjonen og vindkreftene sterkt påvirker løsningen være et dårlig alternativ. Det finnes derfor også en rekke andre åpne randbetingelser (se for eksempel *Røed and Cooper*, 1986, 1987; *Palma and Matano*, 2000). I det videre skal vi derfor ta for oss noen av de mest populære åpne grenseflatebetingelser som brukes.

7.4 The sponge

En meget anvendt betingelse, populært kalt svampen, går ut på å legge til et område utenfor det opprinnelige beregningsområdet som har egenskaper som en svamp. Dette oppnås ved at man i tilleggsområdet, eller svampområdet, øker de naturlig forekommende friksjons- og/eller diffusjons-parameterene slik at løsningene dempes progressivt innover i svampområdet.

Som et eksempel la oss igjen se på (7.5) men nå med bunnfriksjon, that is,

$$\partial_t u = -g\partial_x h - \gamma u \tag{7.25}$$

$$\partial_t h = -H\partial_x u \tag{7.26}$$

hvor γ er en konstant. Videre skal (7.25) og (7.26) løses for $0 < x < L$ med $x = 0$ og $x = L$ som åpne render. Utenfor de åpne rendene legger vi nå til en svamp, for instance for $x > L$ et område fra $L < x < LL$. I dette området økes nå verdien av γ eksponensielt, that is,

$$\gamma = \begin{cases} \gamma_0 & ; \quad 0 \le x \le L \\ \gamma_0 e^{\lambda(x-L)} & ; \quad L < x \le LL \end{cases} \tag{7.27}$$

hvor γ_0 er verdien av friksjonsparameteren γ i det opprinnelige beregningsområdet, ofte kalt det indre, og λ er en parameter som bestemmer hvor hurtig friksjonen i svampen øker.

Sålenge γ er en konstant lokalt, that is, er en saktevarierende funksjon av x, kan en bølgeløsning av (7.25) og (7.26) skrives

$$h = h_0 e^{-\frac{1}{2}\gamma t} e^{i\alpha(x-ct)}, \tag{7.28}$$

hvor

$$c = \sqrt{gH - \left(\frac{\gamma}{2\alpha}\right)^2}, \tag{7.29}$$

og hvor α er bølgelengden og h_0 en konstant. Derav ser vi at når γ øker vil amplituden avta, og, sålenge bølgelengden er fast, at også fasehastigheten, c, avtar. Dette medfører at bølger som går inn i svampen, hvor γ øker i henhold til (7.27), vil bremses opp samtidig som de dempes. Kravet

om at γ skal være en lokal konstant gjør at svampområdet, som må legges til det opprinnelige beregningsområdet må ha en viss størrelse. Dersom γ øker for raskt, eller om vi vil svampområdet er for kort, vil ikke bølgene oppføre seg i henhold til (7.28) og refleksjoner vil oppstå.

Et annet problem med svampen oppstår dersom vi har tvungne bølger, that is, har pådrag på randen (se *Røed and Cooper*, 1986), that is, et likningsett av typen

$$\partial_t u = -g\partial_x h - \gamma u + \tau \tag{7.30}$$
$$\partial_t h = -H\partial_x u \tag{7.31}$$

hvor τ er pådraget. I dette tilfellet vil løsningen i svampen lokalt være sammensatt av en bølgeløsning og en løsning dominert av balansen mellom vindkreftene, τ, og friksjonskreftene, that is, $u = \tau/\gamma$, hvilket medfører at når γ øker vil u avta så kraftig at masse hoper seg opp i svampen. Dette medfører på sikt at løsningen i det indre også påvirkes.

7.5 A weakly reflective OBC

Ved å benytte karakteristikkmetoden er det mulig å konstruere en svakt reflekterende grenseflatebetingelse også for problem som er ikke lineære og som inneholder både effekten av jordrotasjon og vindkrefter (se *Røed and Cooper*, 1987).

Som et eksempel la oss igjen betrakte likning (7.5), men nå legge på effekter av ikke-lineære ledd, jordrotasjon og vindkrefter, that is, gruntvannslikningene,

$$\partial_t u + u\partial_x u - fv = -g\partial_x h + F^x \tag{7.32}$$
$$\partial_t v + u\partial_x v + fu = F^y \tag{7.33}$$
$$\partial_t h + \partial_x(hu) = 0 \tag{7.34}$$

hvor F^x, F^y er komponentene av vindkreftene, og f er Coriolis parameteren. Tar vi nå og multipliserer (7.34) med λ, og legger til den første likningen og danner en felles operator definert ved

$$\frac{D^*}{dt} = \partial_t + \frac{D^* x}{dt}\partial_x, \tag{7.35}$$

får vi at λ må oppfylle likningen

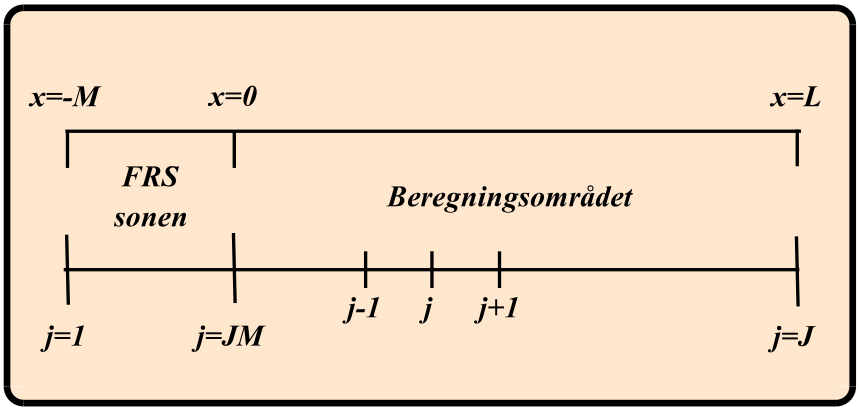
$$\frac{D^* x}{dt} = u + \lambda h = u + \frac{g}{\lambda}, \tag{7.36}$$

som gir

$$\lambda_{1,2} = \pm\frac{c}{h}, \quad c = \sqrt{gh}. \tag{7.37}$$

Etter litt regning følger herav at de to kompatibilitetslikningen som svarer til (7.32) og (7.34) og de tilhørende karakteristiske liknignene er gitt ved,

$$\left(\frac{D^*}{dt}\right)_{1,2}(u \pm 2c) = fv + F^x \quad \text{langs} \left(\frac{D^* x}{dt}\right)_{1,2} = u \pm c. \tag{7.38}$$



Figur 7.3: Skisse av FRS sonen, beregningsområdet og de tilhørende tellere.

For å unngå refleksjon for $x = L$ må vi sette at

$$\left(\frac{D^* x}{dt}\right)_2 \bigg|_{x=L} = 0. \tag{7.39}$$

Ved innsetting i (7.38) gir dette at kompabilitetslikningene erstattes av likningene,

$$\partial_t(u + 2c) + (u + c)\partial_x(u + 2c) = fv + F^x \tag{7.40}$$

$$\partial_t(u - 2c) = fv + F^x \tag{7.41}$$

for $x = L$. En endelig differansetilnærmelse av disse likningen på randen vil gi en svak reflektiv betingelse.

7.6 The flow relaxation scheme (FRS)

En metode som først er kommet til senere, og som også kommer fra meteorologi, kalles "Flow Relaxation Scheme", eller strømtilpassningsskjemaet (*Davies*, 1983; *Martinsen and Engedahl*, 1987; *Cooper and Thompson*, 1989; *Engedahl*, 1995a). FRS som åpen grenseflatebetingelse brukes for instance i Meteorologisk institutt's havvarslingsmodell MI-POM som daglig produserer varsler av vannstand, strøm og hydrografi i norske farvann (*Engedahl*, 1995b; *Røed and Fossum*, 2004)[3]. En relativt god beskrivelse av FRS som åpen grenseflatebetingelse finnes også i *Shi et al.* (1999, 2001). En av fordelene med dette skjemaet er at det relativt enkelt tillater å spesifisere en gitt ytre løsning, for instance tidevann, som dermed kan påtvinges som løsning i beregningsområdet.

I korthet går metoden ut på å modifisere den numeriske løsningen av et hyperbolsk likningssystem innenfor en buffersone (kalt FRS sonen) for hvert tidsskritt. Buffersonen, som består av et

[3] se http://met.no/kyst_og_hav/havvarsel.html

fåtall gitterpunkter (vanligvis 5-20 gitterpunkter), legges til beregningsområdet (se Fig.7.3). Innenfor FRS sonen relakseres (eller tilpasses) den numeriske løsningen til en på forhånd spesifisert løsning, ofte kalt den ytre løsningen. Tilpassningen foregår ved at det defineres en vektfunksjon som for hvert gitterpunkt i FRS sonen beregner et vektet middel av den spesifiserte ytre løsningen og den løsningen som er beregnet numerisk ut i fra de styrende likningene.

La for instance $\phi(x,t)$ være en slik variabel og la beregningsområdet være $0 \leq x < L$, hvor $x = 0$ er en åpen rand. Som Fig. 7.3 viser legges da buffersonen utenfor beregningsområdet fra $x = -M$ til $x = 0$. La nå gitterpunktene være definert ved $x_j = (j-1)\Delta x$, hvor telleren $j = 1(1)J$ gjennomløper alle gitterpunktene, både buffersonen og beregningsområdet. Da svarer $j = J$ til den fysiske randen $x = L$, $j = 1$ til ytterkanten av buffersonen $x = -M$, mens $j = JM$ svarer til punktet $x = 0$, that is, det punktet der buffersonen limes til beregningsområdet ved den åpne randen. La $(\phi_e)_j^{n+1}$ betegne den ytre spesifiserte verdien i ethvert gitterpunkt til tidspunktet $t^{n+1} = (n+1)\Delta t$, og la ϕ_j^* være den numeriske løsningen ved samme tidspunkt i de samme gitterpunkt. I buffersonen $j = 1(1)JM$ er da løsningen ved tidspunktet t^{n+1} gitt som et vektet middel av ϕ_j^* og $(\phi_e)_j^{n+1}$, ved hjelp av formelen

$$\phi_j^{n+1} = (1-\alpha_j)\phi_j^* + \alpha_j(\phi_e)_j^{n+1} \quad ; \quad j = 1(1)JM, \tag{7.42}$$

hvor α_j betegner vektene. Ved å la vekten α_j variere fra $\alpha_1 = 1$ ved ytterkanten av FRS sonen ($j = 1$) til $\alpha_{JM} = 0$ ved den åpne randen $j = JM$, ser vi av (7.42) at ved ytterkanten av buffersonen er den numeriske løsningen lik den spesifiserte, $\phi_j^{n+1} = (\phi_e)_j^{n+1}$, mens ved den åpne randen er den numeriske løsningen lik løsningen av de styrende likningene, $\phi_j^{n+1} = \phi_j^*$.

Eksperimenter viser at oppførselen til løsningen når en anvender FRS er følsom overfor formen til α såvel som vidden til FRS sonen (*Martinsen and Engedahl*, 1987; *Engedahl*, 1995a). I følge *Martinsen and Engedahl* (1987) er en hyperbolsk tangens en brukbar form for α, that is,

$$\alpha_j = 1 - \tanh\frac{j-1}{2} \quad ; \quad j = 1(1)JM. \tag{7.43}$$

En av ulempene med FRS er at man må utvide beregningsområdet til også å omfatte buffersonene, men dette oppveies i en stor grad av fleksibiliteten den gir ved at en ytre løsning kan spesifiseres. En annen ulempe er at anvendelsen av (7.42) gjør brudd på fundamentale konserveringsprinsipper. For å se litt nærmere på det siste skal vi som et eksempel anta at vi skal finne ϕ hvor ϕ er en løsning av det kontinuerlige problemet

Som et eksempel skal vi se på numeriske løsninger av det kontinuerlige problemet

$$\partial_t\phi = \mathcal{L}[\phi] \quad ; \quad x \in [0, L], \tag{7.44}$$

hvor $\mathcal{L}$ er en romlig differensialoperator. La oss som over si at $x = 0$ svarer til den åpne grenseflaten, mens $x = L$ svarer til en fysisk grenseflate, that is, her overtar grenseflatebetingelsen for (7.44). La som før $(\phi_e)_j^{n+1}$ betegne den spesifiserte løsningen. Løser vi nå (7.44) ved hjelp av et i tid ensidig skjema får vi

$$\phi_j^* = \phi_j^n + \Delta t\mathcal{L}_j^n, \tag{7.45}$$

ϕ_j^* er med andre ord en foreløpig løsning. Ved hjelp av (7.42 tilpasses løsningen nå slik at

$$\phi_j^{n+1} = (1-\alpha_j)\phi_j^* + \alpha_j(\phi_e)_j^{n+1} \quad ; \quad j = 1(1)J, \tag{7.46}$$

hvor vi har utvidet området for (7.42) til å omfatte buffersonen såvel som beregningsområdet. For å sikre at løsningen ϕ_j^{n+1} i beregningsområdet er løsning av (7.44) settes vektfunksjonen $\alpha = 0$ for $j = JM(1)J$, that is,

$$\alpha_j = \begin{cases} 1 - \tanh\left(\frac{j-1}{2}\right) & ; \quad j = 1(1)JM - 1 \\ 0 & ; \quad j = JM(1)J \end{cases} \tag{7.47}$$

Dersom vi setter ϕ_j^* fra (7.45) inn i (7.46) etterfulgt av å trekke fra $\alpha_j \phi_j^{n+1}$ på begge sider av likhetstegnet følger at

$$\frac{\phi_j^{n+1} - \phi_j^n}{\Delta t} = \mathcal{L}_j^n + \gamma_j \left[(\phi_e)_j^{n+1} - \phi_j^{n+1} \right] \tag{7.48}$$

hvor koeffisienten γ_j er gitt ved,

$$\gamma_j = \frac{\alpha_j}{1 - \alpha_j}. \tag{7.49}$$

Lar vi nå Δt og Δx gå mot null ser vi at (7.48) er en endelig differansetilnærmelse av den kontinuerlige likningen

$$\partial_t \phi = \mathcal{L}[\phi] + \gamma \left(\phi_e - \phi \right) \quad ; \quad x \in [0, L]. \tag{7.50}$$

Vi observerer at bortsett fra et tilleggsledd er (7.50) lik (7.44). Vi observerer også at tilleggsleddet er differansen mellom den spesifiserte ytre løsningen og den variable selv ganget med en faktor. Denne faktoren er i henhold til (7.49) null ved den åpne randen mens den går mot uendelig når vi nærmer oss enden av buffersonen. Det betyr at betydningen av dette leddet øker jo lengre innover i FRS sonen vi kommer. La oss nå anta at den spesifiserte løsningen settes lik null. Da vil (7.50) være en endelig differansetilnærmelse av den kontinuerlige likningen

$$\partial_t \phi = \mathcal{L}[\phi] - \gamma \phi \tag{7.51}$$

that is, at FRS'en oppfører seg som en svamp med friksjonskoeffesient γ. Fremdeles vil γ være null ved den åpne randen, øke monotont ettersom vi beveger oss innover i buffersonen for til slutt å gå mot uendelig i enden av FRS sonen. Av dette følger at løsningene i FRS'en på mange måter er en svamp, og at løsningen i FRS sonen vil oppføre seg som beskrevet i Kap. 7.4.

For å belyse at FRS metoden gir opphav til falske divergenser skal vi her se på et enkelt eksempel, nemlig problemet gitt i (7.5) og (7.6). Vi observerer først at dersom Δx og Δt går mot null vil (7.46) anta formen

$$\phi = (1 - \alpha)\phi^* + \alpha \phi_e. \tag{7.52}$$

Vi benytter nå denne betingelsen både for h og u. Da følger at

$$(1 - \alpha)\partial_t h^* + \alpha \partial_t h_e = -H(1 - \alpha)\partial_x u^* - H\alpha \partial_x u_e + H(u^* - u_e)\partial_x \alpha. \tag{7.53}$$

De to første leddene på høyre side av (7.53) er de leddene vi ville fått om vi hadde relaksert divergensen til hastigheten direkte, that is, høyre siden av (7.6). Ved å benytte (7.52) som grenseflatebetingelse dukker det opp et tredje ledd på høyre side av (7.53) som skyldes at også tilpassningsparameteren α varierer over FRS sonen.

For at ikke disse "falske" divergensene skal bli for påtrengende følger at FRS sonen må ha en viss lengde og at tilpassningsparameteren α må være saktevarierende, særlig i overgangen mellom beregningsområdet og FRS sonen.

En skal merke seg at denne falske divergensen og det falske "friksjonen" i (7.51) begge forsvinner om den spesifiserte løsningen settes lik løsningen av (7.44), i hvilket tilfelle FRS er en perfekt åpen grenseflatebetingelse. Som regel er ikke dette tilfellet. Hvor god FRS'en er som åpen randbetingelse er derfor avhengig av hvor godt vi kan spesifisere den ytre løsningen.

7.7 Favored combinations

Oppgaver

1. Show that a one-sided, finite difference scheme in time and space of the radiation condition (7.1) can be written

$$\phi_B^{n+1} = \begin{cases} \phi_B^n & ; \quad c_\phi > 0 \\ \left(1 + c_\phi \frac{\Delta t}{\Delta x}\right) \phi_B^n - c_\phi \frac{\Delta t}{\Delta x} \phi_{B+1}^n & ; \quad c_\phi \leq 0 \end{cases} \tag{7.54}$$

   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. Bruk (7.54) til å vise at strålingsbetingelsen "bare" er en interpolasjon av verdier fra det indre.

Kapittel 8

FLERDIMENSJONALE PROBLEM

I de foregående kapitlene har vi, med unntak av Kapittel **??** mest sett på problem med bare en romlig dimensjon. I dette kapittelet skal vi se litt på virkningen av å ta med flere romlige dimensjoner, og se litt på forskjellige måter å konstruere flerdimensjonale gittere.

8.1 Diffusjonsproblemet i to dimensjoner

La oss først se på det to-dimensjonale diffusjonsproblemet, that is,

$$\partial_t \theta = \kappa(\partial_x^2 \theta + \partial_y^2 \theta). \tag{8.1}$$

Ved å benytte en ensidig differanse i tid og sentrerte endelige differanser i x og y (dette var stabilt i en dimensjon) slik vi gjorde for Poisson problemet i Kapittel **??**, kan den numeriske analogen av (8.1) skrives

$$\theta_{jk}^{n+1} = \theta_{jk}^n + \kappa \frac{\Delta t}{\Delta x^2} \left(\theta_{j-1k}^n - 2\theta_{jk}^n + \theta_{j+1k}^n\right) + \kappa \frac{\Delta t}{\Delta y^2} \left(\theta_{jk-1}^n - 2\theta_{jk}^n + \theta_{jk+1}^n\right). \tag{8.2}$$

For å undersøke stabiliteten av dette skjemaet bruker vi von Neumanns metode. Imidlertid innsettes nå en Fourierkomponent som er to-dimensjonal, that is, som har et bølgetall i begge de to retningene,

$$\theta_{jk}^n = \Theta_n e^{i\alpha j \Delta x} e^{i\beta k \Delta y}. \tag{8.3}$$

Innsetting av (8.3) i (8.2) gir da vekstfaktoren

$$G = 1 + 2\kappa \frac{\Delta t}{\Delta x^2} \left(\cos \alpha \Delta x - 1\right) + 2\kappa \frac{\Delta t}{\Delta y^2} \left(\cos \beta \Delta y - 1\right), \tag{8.4}$$

hvor β er bølgetallet i y-retningen. Som vi ser er denne helt lik (3.28) bare med et tillegg på grunn av vi nå har to retninger. Anvender vi von Neumans tilstrekkelig kriterium for numerisk stabilitet (3.25) følger

$$-1 \leq 1 + 2\kappa \frac{\Delta t}{\Delta x^2} \left(\cos \alpha \Delta x - 1\right) + 2\kappa \frac{\Delta t}{\Delta y^2} \left(\cos \beta \Delta y - 1\right) \leq 1 \tag{8.5}$$

Som for det en-dimensjonale tilfellet er den høyre ulikheten i (8.5) automatisk oppfylt, mens den venstre ulikheten gir

$$\kappa\frac{\Delta t}{\Delta x^2}\left(1-\cos\alpha\Delta x\right)+\kappa\frac{\Delta t}{\Delta y^2}\left(1-\cos\beta\Delta y\right)\leq 1 \tag{8.6}$$

Siden venstre side av ulikheten i (8.6) oppnår sin største verdi når $\cos\alpha\Delta x=\cos\beta\Delta y=-1$ følger at dersom

$$\kappa\frac{\Delta t}{\Delta x^2}+\kappa\frac{\Delta t}{\Delta y^2}\leq\frac{1}{2} \tag{8.7}$$

er kriteriet oppfylt for alle valg av α og β. Dette betyr at vi må velge tidsskrittet slik at

$$\Delta t\leq\frac{1}{2\kappa}\frac{\Delta x^2\Delta y^2}{\Delta x^2+\Delta y^2} \tag{8.8}$$

Dersom rutenettet er kvadratisk, that is $\Delta x=\Delta y=\Delta s$, følger at

$$\Delta t\leq\frac{\Delta s^2}{4\kappa} \tag{8.9}$$

Sammenlikner vi nå (8.9) med (3.31) finner vi at tidsskrittet reduseres med en faktor på 2 når flere dimensjoner innføres. Stabilitetskriteriet er altså blitt skjerpet når flere dimensjoner innføres. Dette er et generelt resultat som gjelder for alle skjema, nemlig at stabilitetskravet er strengere jo flere dimensjoner som innføres, selvom om reduksjonsfaktoren kan variere noe.

8.2 Adveksjonsproblemet i to dimensjoner

Vi skal nå ta for oss adveksjonslikningen i to romlige dimensjoner. Denne skrives da

$$\partial_t\theta+\mathbf{u}\cdot\nabla_H\theta=\partial_t\theta+u\partial_x\theta+v\partial_y\theta=0. \tag{8.10}$$

En sentrert annen ordens tilnærming i x og y (leapfrogskjemaet) gir da

$$\theta_{jk}^{n+1}=\theta_{jk}^{n+1}-u\frac{\Delta t}{\Delta x}(\theta_{j+1k}^{n}-\theta_{j-1k}^{n})-v\frac{\Delta t}{\Delta y}(\theta_{jk+1}^{n}-\theta_{jk-1}^{n}). \tag{8.11}$$

Stabiliteten undersøkes som før ved hjelp av von Neumanns metode, that is vi setter inn Fourierkomponenten (8.3) inn i (8.11) får vi som før at vekstfaktoren er bestemt av likningen (4.13), that is

$$G^2+2i\lambda G-1=0, \tag{8.12}$$

men nå med λ gitt ved

$$\lambda=u\frac{\Delta t}{\Delta x}\sin\alpha\Delta x+v\frac{\Delta t}{\Delta y}\sin\beta\Delta y. \tag{8.13}$$

Som før er altså vekstfaktoren G gitt ved

$$G_{1,2}=\pm\sqrt{1-\lambda^2}-i\lambda, \tag{8.14}$$

og som før er altså leapfrogskjemaet også i to romlige dimensjoner nøytralt stabilt under betingelsen av at $|\lambda| \leq 1$. Med det nye uttrykket for λ betyr dette at

$$-1 \leq u\frac{\Delta t}{\Delta x}\sin\alpha\Delta x + v\frac{\Delta t}{\Delta y}\sin\beta\Delta y \leq 1. \tag{8.15}$$

Siden både $-1 \leq \sin\alpha\Delta x \leq 1$ og $-1 \leq \sin\beta\Delta y \leq 1$ er dette oppfylt dersom

$$u\frac{\Delta t}{\Delta x} + v\frac{\Delta t}{\Delta y} \leq 1, \tag{8.16}$$

eller

$$\Delta t \leq \frac{\Delta x\Delta y}{u\Delta y + v\Delta x}. \tag{8.17}$$

Dersom vi setter $u = v = u_s$ og $\Delta x = \Delta y = \Delta s$ følger at bevegelsen er på bølgen

$$u_s\frac{\Delta t}{\Delta s} \leq \frac{1}{2}, \tag{8.18}$$

som bekrefter hypotesen over at stabilitetskravet skjerpes når vi utvider til flere dimensjoner. Dette henger sammen med at når bevegelsen som her går diagonalt i gitteret er "gitteravstanden" ikke gitteravstanden langs aksene men lengden av hypotenusen, altså $\Delta s\sqrt{2}$. Farten i gitteret er derfor gitt ved $\frac{\sqrt{2}\Delta s}{\Delta t}$. Samtidig er den fysiske farten gitt $|u| = \sqrt{u^2 + v^2} = u_s\sqrt{2}$. Som rimelig er må stabilitetskriteriet ta høyde for dette.

Kapittel 9

NOEN AVSLUTTENDE KOMMENTARER

Av de foregående kapitlene, som i hovedsak har tatt utgangspunkt i diffusjons- og adveksjonsproblemet, går det med all tydelighet frem at for å gjøre bruk av numeriske metoder til løsning av atmosfæriske og oseanografiske problem er det nødvendig å vite når og hvorfor en endelig differanse metode virker. Dette kalles for *numerisk analyse*. Den numeriske analysen består altså i sikre oss at den metoden vi har valgt er riktig, og gir oss den "korrekte" løsningen på problemet. Vi har da en algoritme eller et numerisk skjema for løsningen. Det neste problemet er da å få dette til å virke på regnemaskinen. Dette kalles for implementering og er et emne innenfor *regnemaskinvitenskap* (eng: computer science). Dette kan være en frusterende oppgave i seg selv, og tar ofte lang tid. Prosessen kalles ofte for "debugging" i det mye av tiden går med til å fjerne kodefeil, både av formell art og ren feilkoding fra vår egen hånd. Når vi endelig er kommet så langt gjenstår å vise resultatene frem. Det har ingen hensikt å gjøre den numeriske analysen og implementeringen dersom vi ikke kan "se" resultatet. Det siste trinnet er derfor å *visualisere* resultatene. Dette er også en egen vitenskap og kalles *grafisk design*.

De to siste emnene har vi ikke i noen særlig grad berørt, men gjennom løsning av oppgavene i Oppgaveheftet er enhver nødt til også å sette seg inn i de to siste emnene. Alt i alt ser vi at det å løse atmosfæriske og oseanografiske problem ved hjelp av numeriske metoder består av tre deler, nemlig (i) *numerisk analyse*, (ii) *implementering* på en gitt maskinvare, og (iii) *visualisering* av resultatene.

Til slutt skal det nevnes at det finnes flere gode bøker som omhandler emnene over. Et godt eksempel er boken skrevet av *Haltiner and Williams* (1980). Som det fremgår av bokens tittel er den rettet mot meteorologi, men den er blitt stående som klassikeren for meteorologer såvel som oseanografer som en god og grunnleggende lærebok i numerisk analyse. I den senere tid er det også kommet mange tildels gode lærebøker i numerisk analyse rettet mer mot oseanografi. Et godt eksempel er boken skrevet av *Haidvogel and Beckmann* (1999). Også boken skrevet av *Griffies* (2004) bør nevnes, men den kan nok bli i det tyngste laget for enkelte.

Bibliografi

Adamec, D., and J. J. O'Brien (1978), The seasonal upwelling in the Gulf of Guinea due to remote forcing, *J. Phys. Oceanogr.*, *8*, 1050–1060.

Asselin, R. A. (1972), Frequency filter for time integrations, *Mon. Weath. Rev.*, *100*, 487–490.

Chapman, D. C. (1985), Numerical treatment of cross-shelf open boundaries in a barotropic coastal ocean model, *J. Phys. Oceanogr.*, *15*, 1060–1075.

Charney, J. G., R. Fjørtoft, and J. von Neumann (1950), Numerical integration of the barotropic vorticity equation, *Tellus*, *2*, 237–254.

Clancy, R. M. (1981), On wind-driven quasi-geostrophic water movement at fast ice edges, *Mon. Weath. Rev.*, *109*, 1807–1809.

Cooper, C., and J. D. Thompson (1989), Hurricane generated currents on the outer continental shelf, *J. Geoph. Res.*, *94*, 12,513–12,539.

Cushman-Roisin, B. (1984), Analytic, linear stability criteria for the leap-frog, dufort-frankel method, *J. Comp. Phys.*, *53*, 227–239.

Davies, H. C. (1983), Limitation of some common lateral boundary schemes used in regional nwp models, *Mon. Weath. Rev.*, *111*, 1002–1012.

Engedahl, H. (1995a), Use of the flow relaxation scheme in a three-dimensional baroclinic ocean model with realistic topography, *Tellus*, *47A*, 365–382.

Engedahl, H. (1995b), Implementation of the Princeton ocean model (POM/ECOM-3D) at the Norwegian Meteorological Institute (DNMI)., *Research Report 5*, Norwegian Meteorological Institute.

Gill, A. E. (1982), *Atmosphere-ocean dynamics*, *International Geophysical Ser.*, vol. 30, Academic Press.

Griffies, S. M. (2004), *Fundamentals of ocean climate models*, Princeton University Press.

Grotjhan, R., and J. J. O'Brien (1976), Some inaccuracies in finite differencing hyperbolic equations, *Mon. Weath. Rev.*, *104*, 180–194.

Haidvogel, D. B., and A. Beckmann (1999), *Numerical ocean circulation modeling, Series on environmental science and management*, vol. 2, 320 pp., Imperial College Press.

Haltiner, G. J., and R. T. Williams (1980), *Numerical prediction and dynamic meteorology, second edition*, 477 pp., John Wiley & Sons.

Hedstrøm, G. W. (1979), Nonreflecting boundary conditions for nonlinear hyperbolic systems, *J. Comp. Phys.*, *30*, 222–237.

Lighthill, M. J. (1970), *Fourier analysis and generalised functions*, Cambridge University Press, First printed 1958.

Lister, M. (1966), The numerical solution of hyperbolic partial differential equations by the method of characteristics, in *Mathematical Methods for Digital Computers*, edited by A. Ralston and H. S. Wilf, John Wiley, New York.

Martinsen, E. A., and H. Engedahl (1987), Implementation and testing of a lateral boundary scheme as an open boundary condition in a barotropic ocean model., *Coast. Eng.*, *11*, 603–627.

O'Brien, J. J. (1986), *Advanced Physical Oceanographic Numerical Modelling, NATO ASI Series, Ser. C: Mathematical and Physical Sciences*, vol. C 186, D. Reidel Publishing Company.

Orlanski, I. (1976), A simple boundary condition for unbounded hyperbolic flows, *J. Comp. Phys.*, *21*, 251–269.

Palma, E. D., and R. P. Matano (2000), On the implementation of open boundary conditions to a general circulation model: The 3-d case., *J. Geophys. Res.*, *105*, 8605–8627.

Phillips, N. A. (1959), An example of nonlinear computational instability, in *The Atmosphere and the Sea in motion*, edited by B. Bolin, pp. 501–504, Rockefeller Institute Press, New York.

Platzman, G. W. (1954), The computational stability of boundary conditions in numerical integration of the vorticity equation, *Arch. Met. Geophys. Biokl.*, *7*, 29–40.

Richtmyer, R. D. (1963), A survey of difference methods for non-steady fluid dynamics, *Technical Note 63-2*, National Center for Atmospheric Research (NCAR).

Robert, A. J. (1966), The integration of a low order spectral form of the primitive meteorologocal equations, *J. Meteor. Soc. Japan*, *44*, 237–245.

Røed, L. P., and C. K. Cooper (1986), Open boundary conditions in numerical ocean models, in *Advanced Physical Oceanographic Numerical Modelling, Series C: Mathematical and Physical Sciences*, vol. 186, edited by J. O'Brien, pp. 411–436, D. Reidel Publishing Co.

Røed, L. P., and C. K. Cooper (1987), A study of various open boundary conditions for wind-forced barotropic numerical ocean models, in *Three-dimensional models of marine and estuarine dynamics*, *Elsevier Oceanography Series*, vol. 45, edited by J. C. J. Nihoul and B. M. Jamart, pp. 305–335, Elsevier Science Publishers B.V.

Røed, L. P., and I. Fossum (2004), Mean and eddy motion in the Skagerrak/northern North Sea: insight from a numerical model, *Ocean Dynamics*, *54*, 197–220.

Shi, X. B., B. Hackett, and L. P. Røed (1999), Documentation of DNMI's MICOM version, part 2: Implementation of a Flow Relaxation Scheme (FRS), *Research Report 87*, Norwegian Meteorological Institute, [Avaialable from Norwegian Meteorological Institute, Postboks 43 Blindern, N-0313 Oslo, Norway].

Shi, X. B., L. P. Røed, and B. Hackett (2001), Variability of the Denmark Strait overflow: a numerical study, *J. Geophys. Res.*, *106*, 22,277–22,294.

Smolarkiewicz, P., and L. Margolin (1997), MPDATA: a finite difference solver for geophysical flows, *J. Comp. Phys.*, *140*, 459–480.

Smolarkiewicz, P. K. (1983), A simple positive definite advection scheme with small implicit diffusion, *Mon. Wea. Rev.*, *111*, pp. 479.