Chapter 12

Partial differential equations

12.1 Classification

Partial differential equations are differential equations for functions $f(x_1, x_2, \ldots, x_n)$ of several variables. For linear equations of second order in two independent variables,

$$a\frac{\partial^2 f}{\partial x^2} + 2b\frac{\partial^2 f}{\partial x \partial y} + c\frac{\partial^2 f}{\partial x^2} \quad + d\frac{\partial f}{\partial x} + e\frac{\partial f}{\partial y} \quad + gf + h \;=\; 0 \tag{12.1}$$

a useful classification is based on the discriminant $b^2 - ac$ of the quadratic form $F(x, y) \equiv ax^2 + 2bxy + cy^2$:

$b^2 - ac < 0$ **(equation describes an ellipse):** *Elliptic equation*.
Example: Poisson equation

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = -h(x, y) \; . \tag{12.2}$$

Typically steady (time-independent) problems like steady-state heat conduction, elasticity (of thin strings and membranes), stationary wave or Schrödinger equations.

$b^2 - ac = 0$ **(equation describes a parabola):** *Parabolic equation*.
Example: (Time-dependent) heat conduction equation

$$\frac{\partial f}{\partial t} - \chi\frac{\partial^2 f}{\partial x^2} = q(x, t) \; . \tag{12.3}$$

Typically time-dependent problems with damping like diffusion and heat conduction or time dependent Schrödinger equation.

$b^2 - ac > 0$ **(equation describes a hyperbola):** *Hyperbolic equation*.
Example: (Time-dependent) classical wave equation

$$\frac{\partial^2 f}{\partial t^2} - \chi\frac{\partial^2 f}{\partial x^2} = 0 \; . \tag{12.4}$$

Typically time-dependent problems with wave propagation like advection problems, equations for sound waves, electromagnetic waves (possibly with damping: telegrapher's equation).

12.2 Finite differences

Consider a smooth function $f(x)$ sampled on an equidistant grid x_k, which gives the values $f_k = f(x_k)$.[1] We define the following finite difference operators:

shift operator	$(T_\pm f)_k = f_{k\pm1}$,	(12.5)
forward difference operator	$(\Delta f)_k = f_{k+1} - f_k$,	(12.6)
central difference operator	$(\delta f)_k = f_{k+1/2} - f_{k-1/2}$,	(12.7)
backward difference operator	$(\nabla f)_k = f_k - f_{k-1}$,	(12.8)
averaging operator	$(Mf)_k \equiv \overline{f_k} = \frac{f_{k+1/2} + f_{k-1/2}}{2}$.	(12.9)

There are many interrelations between these operators, e.g.

$$T_- = T_+^{-1} , \tag{12.10}$$
$$\Delta = T_+ - 1 \quad \curvearrowright \quad T_+ = 1 + \Delta , \tag{12.11}$$
$$\nabla = 1 - T_- \quad \curvearrowright \quad T_- = 1 - \nabla , \tag{12.12}$$
$$\delta = T_+^{1/2} - T_+^{-1/2} \quad , \tag{12.13}$$
$$M = T_+^{1/2} + T_+^{-1/2} \quad . \tag{12.14}$$

Powers of these operators are obtained by iteratively applying them several times. Schematically:

$$\begin{array}{c|c|c|c} x_2 & f_2 & & \\ & & \Delta f_1 & \\ x_1 & f_1 & & \Delta^2 f_0 \\ & & \Delta f_0 & \\ x_0 & f_0 & & \end{array} \qquad \begin{array}{c|c|c|c} x_1 & f_1 & & \\ & & \delta f_{1/2} & \\ x_0 & f_0 & & \delta^2 f_0 \\ & & \delta f_{-1/2} & \\ x_{-1} & f_{-1} & & \end{array} \qquad \begin{array}{c|c|c|c} x_n & f_n & & \\ & & \nabla f_n & \\ x_{n-1} & f_{n-1} & & \nabla^2 f_n \\ & & \nabla f_{n-1} & \\ x_{n-2} & f_{n-2} & & \end{array}$$

We can define arbitrary analytical functions of the operators via power series, e.g.

$$e^\Delta \equiv 1 + \Delta + \frac{\Delta^2}{2!} + \frac{\Delta^3}{3!} + \dots \tag{12.15}$$

[1] Even if the values x_k are not equidistant, the y can often be written as a smooth function $x(\xi_k)$ of an equidistant variable ξ. In that case, $f_k = f(x(\xi_k))$ is obtained from a smooth function $\xi \mapsto f(x(\xi))$ on an equidistant grid ξ_k, so the following still applies, albeit with some modification.

Once we accept the idea that functions can be applied to operators, we can e.g. invert Eq. (12.13) using the relation

$$\frac{t^{1/2}-t^{-1/2}}{2} = \frac{e^{(1/2)\ln t}-e^{-(1/2)\ln t}}{2} = \sinh\frac{\ln t}{2}\ ,$$

and find

$$\delta = 2\sinh\frac{\ln T_+}{2}\ , \tag{12.16}$$

which can immediately be inverted to

$$\ln T_+ = 2\,\mathrm{arsinh}\,\frac{\delta}{2}\ , \tag{12.17}$$

a result we are going to use later. Similarly,

$$M = \cosh\frac{\ln T_+}{2} = \sqrt{1+\sinh^2\frac{\ln T_+}{2}} = \sqrt{1+\frac{\delta^2}{4}}\ . \tag{12.18}$$

Newton's interpolation formula: Interpolation can be understood as a fractional shift operation of the known data:

$$f(x) \;=\; T_+^t f_0 \;=\; (1+\Delta)^t f_0 \;=\; \sum_{k=0}^{n}\binom{t}{k}\Delta^k f_0 + R_n \tag{12.19}$$

$$\;=\; f_0 + \binom{t}{1}\Delta f_0 + \binom{t}{2}\Delta^2 f_0 + \ldots + \binom{t}{n}\Delta^n f_0 + R_n\ , \qquad t := \frac{x-x_0}{h} \tag{12.20}$$

$$\tag{12.21}$$

Remainder term

$$R_n = \binom{t}{n+1} h^{n+1} f^{(n+1)}(x_0+n\vartheta h)\ , \qquad 0<\vartheta<1\ . \tag{12.22}$$

Stirling's interpolation formula:

$$f(t) \;=\; f_0 + t\,\overline{\delta f_{\pm\frac12}} + \frac{t^2}{2!}\delta^2 f_0 + \frac{t(t^2-1)}{3!}\,\overline{\delta^3 f_{\pm\frac12}} + \frac{t^2(t^2-1)}{4!}\,\delta^4 f_0 +$$

$$+\frac{t(t^2-1)(t^2-4)}{5!}\,\overline{\delta^5 f_{\pm\frac12}} + \ldots \tag{12.23}$$

$$\;=\; f_0 + \binom{t}{1}\left(\overline{\delta f_{\pm\frac12}} + \frac{t}{2}\delta^2 f_0\right) + \binom{t+1}{3}\left(\overline{\delta^3 f_{\pm\frac12}} + \frac{t}{4}\delta^4 f_0\right) + \tag{12.24}$$

$$+\binom{t+1}{5}\left(\overline{\delta^5 f_{\pm\frac12}} + \frac{t}{6}\delta^6 f_0\right) + + \ldots \tag{12.25}$$

with

$$\overline{\delta^k f_{\pm\frac12}} \;:=\; \frac{\delta^k f_{\frac12} + \delta^k f_{-\frac12}}{2} \tag{12.26}$$

Example: Interpolate the following function values:

$x =$	-1	0	1	2
$f(x) =$	-7	1	-1	-7

The finite-difference tableau becomes

k	x_k	f_k	Δf	$\Delta^2 f$	$\Delta^3 f$
2	2	-7			
			-6		
1	1	-1		-4	
			-2		6
0	0	1		-10	
			8		
-1	-1	-7			

We have $t = x + 1$, and Newton's formula becomes

$$\begin{aligned} f &= f_0 + \binom{t}{1}\Delta f_0 + \binom{t}{2}\Delta^2 f_0 + \binom{t}{3}\Delta^3 f_0 && (12.27)\\ &= -7 + 8t - 10\frac{t(t-1)}{2} + 6\frac{t(t-1)(t-2)}{6} && (12.28)\\ &= -7 + 8(x+1) - 5(x+1)x + (x+1)x(x-1) && (12.29)\\ &= x^3 - 5x^2 + 2x + 1\ . && (12.30) \end{aligned}$$

Taylor's theorem: We introduce the derivative operator D with

$$Df_0 \equiv f_0'\ . \tag{12.31}$$

If $f(x)$ is sufficiently smooth, it has a Taylor series

$$f(x_0 + h) = \sum_{j=0}^{\infty} \frac{h^j f^{(j)}}{j!}\ , \tag{12.32}$$

which in terms of our operators becomes

$$T_+ f_0 = \sum_{j=0}^{\infty} \frac{(hD)^j}{j!} f_0 = e^{hD} f_0\ . \tag{12.33}$$

Thus,

$$T_+ = e^{hD}\ , \tag{12.34}$$

which can (formally) be inverted as

$$hD \;=\; \ln T_+ \;=\; \ln(1+\Delta) \;=\; -\ln(1-\nabla) \;=\; 2\,\mathrm{arsinh}\,\frac{\delta}{2}\ . \tag{12.35}$$

We can use this and similar formulae to express the derivative operator as a power series in Δ, ∇ or δ:

$$hD = \ln(1+\Delta) = \frac{\Delta}{1} - \frac{\Delta^2}{2} + \frac{\Delta^3}{3} - + \dots \tag{12.36}$$

$$= -\ln(1-\nabla) = \frac{\nabla}{1} + \frac{\nabla^2}{2} + \frac{\nabla^3}{3} + \dots \,. \tag{12.37}$$

For the central differences, a few manipulations [I1996] result in[2]

$$hD = M\left(1+\frac{\delta^2}{4}\right)^{-1/2} 2\,\mathrm{arsinh}\,\frac{\delta}{2} \tag{12.39}$$

$$= M\delta\left[\sum_{j=0}^{\infty}(-1)^j\binom{2j}{j}\left(\frac{\delta^2}{16}\right)^j\right]\left[\sum_{j=0}^{\infty}\frac{(-1)^j}{2j+1}\binom{2j}{j}\left(\frac{\delta^2}{16}\right)^j\right] \tag{12.40}$$

$$= M\left(\delta - \frac{\delta^3}{6} + \frac{\delta^5}{30} - \frac{\delta^7}{140} + \frac{\delta^9}{630} - + \dots\right), \tag{12.41}$$

More explicitly, these operator identities read

$$h\,f_0' = \Delta f_0 - \frac{\Delta^2 f_0}{2} + \frac{\Delta^3 f_0}{3} - \frac{\Delta^4 f_0}{4} + - \dots \tag{12.42}$$

$$= \nabla f_0 + \frac{\nabla^2 f_0}{2} + \frac{\nabla^3 f_0}{3} + \frac{\nabla^4 f_0}{4} + \dots \tag{12.43}$$

$$= \overline{\delta f_0} - \frac{\overline{\delta^3 f_0}}{6} + \frac{\overline{\delta^5 f_0}}{30} - \frac{\overline{\delta^7 f_0}}{140} + - \dots \,. \tag{12.44}$$

For the second-order derivatives, we get

$$h^2D^2 = \ln^2(1+\Delta) = \ln^2(1-\nabla) = 4\,\mathrm{arsinh}^2\,\frac{\delta}{2}\,, \tag{12.45}$$

and thus (eventually)

$$h^2\,f_0'' = \Delta^2 f_0 - \Delta^3 f_0 + \frac{11}{12}\Delta^4 f_0 - \frac{5}{6}\Delta^5 f_0 + - \dots \tag{12.46}$$

$$= \nabla^2 f_0 + \nabla^3 f_0 + \frac{11}{12}\nabla^4 f_0 + \frac{5}{6}\nabla^5 f_0 + \dots \tag{12.47}$$

$$= \delta^2 f_0 - \frac{\delta^4 f_0}{12} + \frac{\delta^6 f_0}{90} - \frac{\delta^8 f_0}{560} + - \dots \,. \tag{12.48}$$

[2] The reason this expression is so complex is that we need an expansion in $M\delta f_0, M\delta^3 f_0, M\delta^5 f_0, \dots$ (because these operators live on the main grid, just as f_i), while the straight-forward Taylor expansion yields

$$hD = 2\,\mathrm{arsinh}\,\frac{\delta}{2} = \delta\sum_{j=0}^{\infty}\frac{(-1)^j}{2j+1}\binom{2j}{j}\left(\frac{\delta^2}{16}\right)^j = \delta - \frac{\delta^3}{24} + \frac{3}{640}\delta^5 - + \dots\,, \tag{12.38}$$

and thus leads to an expansion in $\delta f_0, \delta^3 f_0, \dots$, all terms of which are defined on the shifted (half-index) grid.

Using the function values instead of the difference operators, these formulae (truncated at different levels) become

$$f_0' = \frac{-f_{-1} + f_{-1}}{2h} + O\left(h^2\right) \tag{12.49}$$

$$f_0' = \frac{f_{-2} - 8f_{-1} + 8f_1 - f_2}{12h} + O\left(h^4\right) \tag{12.50}$$

$$f_0' = \frac{-f_{-3} + 9f_{-2} - 45f_{-1} + 45f_1 - 9f_2 + f_3}{60h} + O\left(h^6\right) \tag{12.51}$$

and

$$f_0'' = \frac{f_{-1} - 2f_0 + f_{-1}}{h^2} + O\left(h^2\right) \tag{12.52}$$

$$f_0'' = \frac{-f_{-2} + 16f_{-1} - 30f_0 + 16f_1 - f_2}{12h^2} + O\left(h^4\right) \tag{12.53}$$

$$f_0'' = \frac{2f_{-3} - 27f_{-2} + 270f_{-1} - 490f_0 + 270f_1 - 27f_2 + f_3}{180h^2} + O\left(h^6\right) \tag{12.54}$$

12.3 Elliptic problems

Consider the Poisson equation

$$\Delta f \equiv \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = g(x, y)\,. \tag{12.55}$$

Introducing an equidistant grid

$$x_k = x_0 + kh\,, \qquad y_k = y_0 + kh \tag{12.56}$$

(with identical grid spacing for x and y), we have a two-dimensional array $f_{kl} \equiv f(x_k, y_l)$. To second order in δx and δy, we can approximate the second derivatives by

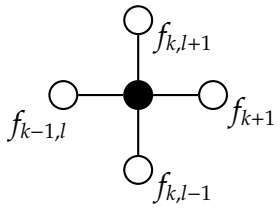
$$\left(\frac{\partial^2 f}{\partial x^2}\right)_{kl} = \frac{f_{k+1,l} - 2f_{k,l} + f_{k-1,l}}{h^2} + O\left(h^2\right)\,, \tag{12.57}$$

$$\left(\frac{\partial^2 f}{\partial y^2}\right)_{kl} = \frac{f_{k,l+1} - 2f_{k,l} + f_{k,l-1}}{h^2} + O\left(h^2\right)\,, \tag{12.58}$$

$$\tag{12.59}$$

and thus

$$(\Delta f)_{kl} = \frac{f_{k+1,l} + f_{k-1,l} + f_{k,l+1} + f_{k,l-1} - 4f_{k,l}}{h^2} + O\left(h^2\right)\,. \tag{12.60}$$



Using this, we can write a *finite-difference version* of the Poisson equation in the form

$$\frac{f_{k+1,l} + f_{k-1,l} + f_{k,l+1} + f_{k,l-1} - 4f_{k,l}}{h^2} = g_{kl} \ . \tag{12.61}$$

Solving this equation, we will get an approximation to the correct $f(x_k, y_l)$ that improves if we increase the number of grid points (i.e. decrease h).

Note: The linear system (12.61) gives rise to a *sparse* matrix, where most elements are zero (each line of the matrix has only 5 non-zero elements). Solving this with a general-purpose method like full LU decomposition would be an extreme waste of computing time (as we would mostly multiply some numbers by zero), however there are special methods to solve such systems, typically by iteration (see below).

12.3.1 Fourier method

In Fourier space, the Poisson equation (12.55) becomes an algebraic equation:

$$-\mathbf{k}^2 \tilde{f}(\mathbf{k}) = \tilde{g}(\mathbf{k}) \ . \tag{12.62}$$

Here

$$\tilde{f}(\mathbf{k}) = \mathcal{F}\{f(\mathbf{x}); \mathbf{k}\} = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(\mathbf{x})\, e^{-i\mathbf{k}\cdot\mathbf{x}}\, dx^2 \tag{12.63}$$

is the forward Fourier transform, with

$$f(\mathbf{x}) = \mathcal{F}^{-1}\{\tilde{f}(\mathbf{k}); \mathbf{x}\} = \int_{-\infty}^{\infty} \tilde{f}(\mathbf{k})\, e^{i\mathbf{k}\cdot\mathbf{x}}\, dk^2 \tag{12.64}$$

being the corresponding backward transform.

Equation (12.62) yields simply

$$\tilde{f}(\mathbf{k}) = -\frac{\tilde{g}(\mathbf{k})}{\mathbf{k}^2} \ , \tag{12.65}$$

or

$$f(\mathbf{x}) = -\mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{g(\mathbf{x}), \mathbf{k}\}}{\mathbf{k}^2}, \mathbf{x}\right\} \ . \tag{12.66}$$

For $k \equiv |\mathbf{k}| \to 0$ there is an apparent problem because we divide by k^2, but $k = 0$ simply represents a constant term in $f(\mathbf{x})$, and as Eq. (12.55) is not affected by a constant shift in $f(\mathbf{x})$, we can choose $\tilde{f}(\mathbf{0}) = 0$ (and re-add a constant in the end, if necessary).

If we solve the Poisson equation not in an infinite volume, but in a finite box with periodic boundary conditions, the backward Fourier integral (12.64) turns into a Fourier series, while integration in the forward transform (12.63) is only over the volume of the box.

In numerical mathematics, however, even the positions $\mathbf{x}$ in configuration space are discrete, as we *discretize* the box volume by introducing an equidistant grid. In that case, we are left with the *discrete Fourier transform*

$$\tilde{f}(\mathbf{k}_{jl}) \;=\; \mathcal{F}\{f(\mathbf{x}_{mn});\mathbf{k}_{jl}\} \;=\; \sum_{m,n} f(\mathbf{x}_{mn})\, e^{-i\mathbf{k}_{jl}\cdot\mathbf{x}_{mn}} \;=\; \sum_{m,n} f(\mathbf{x}_{mn})\, e^{-2\pi i\left(\frac{jm}{N_x}+\frac{ln}{N_y}\right)} \tag{12.67}$$

$$f(\mathbf{x}_{mn}) \;=\; \mathcal{F}^{-1}\{\tilde{f}(\mathbf{k}_{jl});\mathbf{x}_{mn}\} \;=\; \frac{1}{N}\sum_{j,l} f(\mathbf{k}_{jl})\, e^{\mathbf{k}_{jl}\cdot\mathbf{x}_{mn}} \;=\; \frac{1}{N}\sum_{j,l} f(\mathbf{k}_{jl})\, e^{2\pi i\left(\frac{jm}{N_x}+\frac{ln}{N_y}\right)}\,, \tag{12.68}$$

where $\delta k_x = 2\pi/L_x$, $\delta k_y = 2\pi/L_y$, and summation is over all points in the box.

To approximately solve the Poisson equation on the grid points, we use Eq. (12.66) and avoid division by zero for $\mathbf{k}_{jl} = 0$ as discussed above.

We thus have the following recipe:

1. Choose an equidistant grid $\mathbf{x}_{mn}$ to cover the box; make sure it is compatible with the periodic boundary conditions (no point counted twice).
2. Calculate $g(\mathbf{x}_{mn})$ on the grid.
3. Calculate the Fourier transform $\tilde{g}(\mathbf{k}_{jl})$
4. Use
$$\tilde{f}(\mathbf{k}_{jl}) = \begin{cases} -\dfrac{\tilde{g}(\mathbf{k}_{jl})}{k^2}\,, & k \neq 0 \\ 0\,, & k = 0 \end{cases} \tag{12.69}$$
to calculate $\tilde{f}(\mathbf{k}_{jl})$.
5. Transform back to get $f(\mathbf{x}_{jl})$.

How do we calculate the discrete Fourier transform? Many software packages (and IDL) provide the *Fast Fourier Transform* (FFT). This is a very efficient method, particularly if the number of points N_x and N_y are powers of 2. In two dimensions, the operation count of the Fast Fourier Transform is

$$n_{\rm ops} \;\propto\; N_x \ln N_x \times N_y \ln N_y \;\propto\; N \ln^2 N\,, \tag{12.70}$$

where $N = N_x N_y$ is the total number of points; this is almost linear in N, as the logarithm grows very slowly.

Note 1: For Dirichlet boundary conditions $f|_{\partial V} = 0$, one uses the Fourier sine transform instead of the complex Fourier transform. Similarly, von Neumann boundary conditions require the Fourier cosine transform.

Note 2: The Fourier method can only be applied to equations with constant coefficients. For that case, it is one of the most efficient methods to solve elliptic PDEs.

12.3.2 Multigrid method

A very efficient way of solving the discretized Poisson equation is the *multigrid method* which, together with the main grid, introduces a number of coarser grids, typically with two, four, etc. times the grid spacing in each direction.

Consider the linear system (12.61). Under certain conditions (which are typically met), it can be solved iteratively using

$$f_{k,l} = \frac{f_{k+1,l} + f_{k-1,l} + f_{k,l+1} + f_{k,l-1} - h^2 g_{kl}}{4} \,. \tag{12.71}$$

There are two variants of this iteration scheme. For *Jacobi iteration*, at a given stage the right hand sides are evaluated for all grid points and only then the values f_{kl} are updated. Convergence is two times faster with *Gauss–Seidel* iteration, where Eq. (12.71) is evaluated for each point in sequence, using the latest updated values on the right hand side. Gauss–Seidel iteration also requires only half the memory of Jacobi iteration and has the important advantage of damping the highest wave number. This is the scheme we will consider here.

The iterative solution of Eq. (12.71) is quite slow. While the small scales ($\sim \delta x$) are quickly converging, it is the largest scales that take very long to reach their 'equilibrium' values due to the (local) iteration. For these large scales, however, one would not need the fine grid, and on a coarser grid they would converge much faster. Hence the idea of multigrid methods: Solve the problem iteratively on grids of different resolution by

1. *coarse-graining* (downsampling) using the *restriction matrix R*:

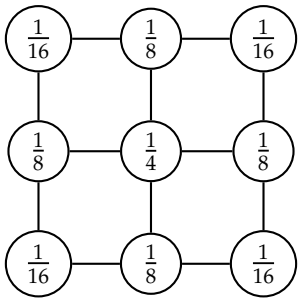
$$\mathbf{r}_{\text{coarse}} = \mathcal{R}\mathbf{r}_{\text{fine}} \,, \tag{12.72}$$

   and

2. *fine-graining* (refining = interpolation), using the *prolongation matrix P*:

$$\delta\mathbf{f}_{\text{fine}} = \mathcal{P}\delta\mathbf{f}_{\text{coarse}} \,. \tag{12.73}$$

A popular choice for the restriction matrix is represented by



This matrix acts as a *lowpass filter*: signals at the Nyquist frequency $f_{\text{Ny,fine}}$ of the fine grid are completely filtered out, while signals at $f_{\text{Ny,coarse}}$ are damped as little as possible.

This is crucial for the scheme to be efficient, as any contribution of $f_{\mathrm{Ny,fine}}$ would only be misinterpreted as a larger frequency on the coarser grid (*aliasing*).

The refinement is done using linear interpolation.

To see how the method works, consider the differential equation

$$\mathcal{L}f = g \ , \tag{12.74}$$

which, discretized at the grid spacing h, becomes

$$\mathcal{L}_h f_h = g_h \ . \tag{12.75}$$

If we have an approximate solution $\tilde{f_h}$, we can introduce the error

$$\delta f_h = f_h - \tilde{f_h} \ , \tag{12.76}$$

and find

$$-\mathcal{L}_h \delta f_h = \mathcal{L}_h \tilde{f_h} - \mathcal{L}_h f_h = \mathcal{L}_h \tilde{f_h} - g_h \ , \tag{12.77}$$

i.e.

$$-\mathcal{L}_h \delta f_h = r_h \ , \tag{12.78}$$

where $r_h \equiv \mathcal{L}_h \tilde{f_h} - g_h$ is the *residual*, which is a measure of how well our approximate solution $\tilde{f}$ solves the original problem.

Coarse-graining r_h to the coarser grid with spacing $H = 2h$,

$$r_H = \mathcal{R} r_h \ , \tag{12.79}$$

we arrive at

$$-\mathcal{L}_H \delta f_H = r_H \ , \tag{12.80}$$

which we solve (this is faster than on the finer grid) to obtain δf_H.

Then we fine-grain (interpolate) δf_H onto the finer grid,

$$\delta \tilde{f_h} = \mathcal{P} \delta f_H \ , \tag{12.81}$$

and calculate the new value of $\tilde{f_h}$ as

$$\tilde{f_h}^{\mathrm{new}} = \tilde{f_h} + \delta \tilde{f_h} \ . \tag{12.82}$$

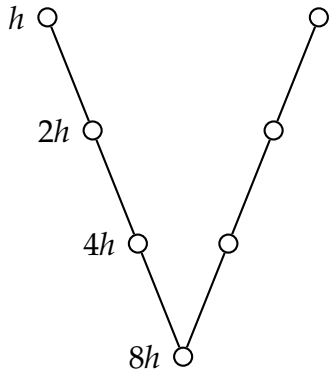
Formulated as a recipe, this two-grid method reads:

0. Start with a guess $\tilde{f_h}$ on the fine grid (e.g. choose 0).
1. Calculate the residual $r_h = \mathcal{L}_h \tilde{f_h} - g_h$, and coarse-grain it, $r_H = \mathcal{R} r_h$.
2. Solve $\mathcal{L}_H \, \delta f_H = -r_H$ on the coarser grid, fine-grain the correction, $\delta \tilde{f_h} = \mathcal{P} \delta f_H$, and add it to the initial guess. $\tilde{f_h}^{\mathrm{new}} = \tilde{f_h} + \delta \tilde{f_h}$.
3. Do one Gauss–Seidel iteration.
4. Continue with 1 until $\delta \tilde{f_h}$ is small enough.

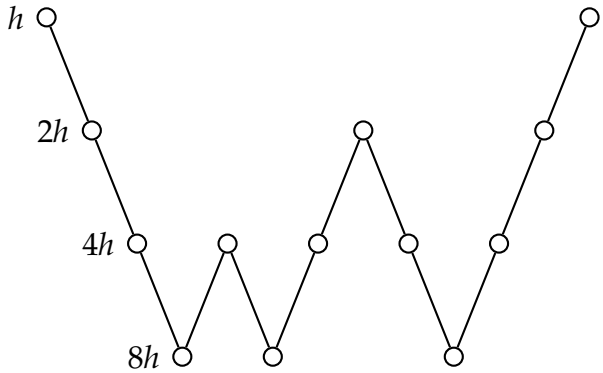
To turn this from a two- into a multi-grid method, we simply use another two-grid scheme for obtaining δf_H at stage 2, etc. The best way to code a multigrid method is obviously recursive.

Note 1: The numerical cost for the multigrid method is roughly $\propto N = N_x N_y$ and thus comparable to Fourier methods. However, multigrid methods can be used for equations with variable coefficients and also for nonlinear equations.

Note 2: The multigrid scheme thus described is referred to as 'V' cycle (the name should be evident from the scheme below). There are other popular cycles like the 'W' cycle.



The 'V' cycle.



A 'W' cycle (many other W cycles exist).

12.4 Parabolic problems

The heat conduction equation

$$\frac{\partial T}{\partial t} = \chi \frac{\partial^2 T}{\partial x^2} \tag{12.83}$$

is the prototype of a parabolic differential equation. To solve it numerically, we need to discretize in space and time:

$$T_k^l \equiv T(x_k, t_l) \; . \tag{12.84}$$

Starting from an initial condition T_k^l, which we assume to be known everywhere, we need to construct the solution at the next time, T_k^{l+1}.

12.4.1 Explicit scheme

A simple scheme is obtained from the discretization

$$\frac{T_k^{l+1} - T_k^l}{\delta t} = \chi \; \frac{T_{k-1}^l - 2T_k^l + T_{k+1}^l}{\delta x^2} \; . \tag{12.85}$$

We can explicitly solve for the unknown T_k^{l+1},

$$T_k^{l+1} \; = \; T_k^l + \frac{\chi \, \delta t}{\delta x^2} \left(T_{k-1}^l - 2T_k^l + T_{k+1}^l \right) \; = \; C T_{k-1}^l + (1-2C) T_k^l + C T_{k+1}^l \; , \tag{12.86}$$

where

$$C_{\text{dif}} \equiv \frac{\chi \delta t}{\delta x^2} \tag{12.87}$$

is the (diffusive) *Courant number*; we will mostly omit the index 'dif' and refer to C.

According to the discretizations used, this *explicit scheme* is of first order in time (Euler stepping), and of second order in space (centred second derivative).

12.4.2 Fully implicit scheme

Another scheme is obtained from the discretization

$$\frac{T_k^{l+1} - T_k^l}{\delta t} = \chi \, \frac{T_{k-1}^{l+1} - 2T_k^{l+1} + T_{k+1}^{l+1}}{\delta x^2} \, . \tag{12.88}$$

This time, we have a coupled system of equations for the unknown T_k^{l+1},

$$-CT_{k-1}^{l+1} + (1+2C)T_k^{l+1} - CT_{k+1}^{l+1} = T_k^l \tag{12.89}$$

Again, this *fully implicit scheme* is of first order in time (Euler stepping), and of second order in space (centred second derivative).

Note: The system (12.89) is a tridiagonal system and can thus be solved much more efficiently than a general linear system.

Nevertheless, it takes more numerical effort than the explicit scheme and (like any implicit scheme) has the major disadvantage that it leads to nonlinear systems of equations if the heat conduction equation is nonlinear (i.e. if χ depends on temperature T).

12.4.3 General implicit and Crank-Nicholson schemes

We can mix the explicit and the fully implicit schemes with a weighting factor $0 \le q \le 1$ to obtain the *general implicit scheme*:

$$\frac{T_k^{l+1} - T_k^l}{\delta t} = \frac{\chi}{\delta x^2} \, \left[q \left(T_{k-1}^{l+1} - 2T_k^{l+1} + T_{k+1}^{l+1} \right) + (1-q) \left(T_{k-1}^{l} - 2T_k^{l} + T_{k+1}^{l} \right) \right] \, . \tag{12.90}$$

The special cases $q = 0$ and $q = 1$ correspond to the explicit and fully implicit schemes, respectively.

The linear system takes the form

$$-qCT_{k-1}^{l+1} + (1+2qC)T_k^{l+1} - qCT_{k+1}^{l+1} \;=\; (1-q)CT_{k-1}^{l} + [1-2(1-q)C]T_k^l + (1-q)CT_{k+1}^l \, , \tag{12.91}$$

which is again a tridiagonal system.

The general implicit scheme is typically of first order in time and of second order in space, but in the special case $q = 1/2$, the temporal order is 2. This scheme,

$$-\frac{C}{2}T_{k-1}^{l+1} + (1+C)T_k^{l+1} - \frac{C}{2}T_{k+1}^{l+1} = \frac{C}{2}T_{k-1}^{l} + (1-C)T_k^{l} + \frac{C}{2}T_{k+1}^{l} , \tag{12.92}$$

is called *Crank–Nicholson scheme*.

12.4.4 Stability

If we apply the explicit scheme with a relatively large time step, we find that oscillatory perturbations grow rapidly and make the numerical solution quickly useless. On the other hand, for small δt the solution is very well behaved and settles to the physical steady state.

To assess the *stability* of a scheme and predict the useful range of δt, we use *von Neumann stability analysis*: Assume that at time t_l, temperature T varies harmonically in x,

$$T_j^l = e^{ikx_j} . \tag{12.93}$$

For each of these Fourier modes (characterized by the wave number k), investigate the amplitude factor A given by

$$T_j^{l+1} = AT_j^l . \tag{12.94}$$

If $|A| < 1$, the Fourier mode is stable, otherwise it is unstable. If at least one Fourier mode is unstable, the scheme is unstable, too, as any weak perturbation containing that Fourier mode will grow exponentially and make the solution unusable.

What is the range of wave numbers that are meaningful on a grid of spacing δx? The lowest wave number is 0, while the highest wave number corresponds to a zigzag profile with a period of $2\,\delta x$, thus $k \le 2\pi/(2\delta x) = \pi/\delta x \equiv k_{\mathrm{Ny}}$.[3] Here $k_{\mathrm{Ny}} \equiv \pi/\delta x$ is called the *Nyquist wave number*. For a finite number of grid points, the wave numbers will be discrete, with a spacing $\delta k = 2\pi/L_x$, where $L_x = N_x\,\delta x$ is the interval length. By increasing N_x, we can thus decrease δk as far as we want, hence we will treat k as a continuous variable.

[3] The fact that k_{Ny} is the highest wave number that can be distinguished on a grid of spacing δx can also be seen from

$$e^{i(k_{\mathrm{Ny}}+k')x_l} = (-1)^l e^{ik'x_l} = e^{i(-k_{\mathrm{Ny}}+k')x_l} , \tag{12.95}$$

which implies that the wave number $k_{\mathrm{Ny}}+k'$ is equivalent to $-k_{\mathrm{Ny}}+k'$.

Von Neumann stability analysis:

Assume

$$T_j^l = e^{ikx_j} \ . \tag{12.96}$$

Consider the wave length k to be a continuous number in the interval

$$0 \le k \le k_{\rm Ny} \ , \tag{12.97}$$

Analyze amplification factor A: if $A \le 1 \forall k$, scheme is stable, otherwise unstable.

Often we will work with the dimensionless wave number

$$\kappa \equiv k\,\delta x \ , \tag{12.98}$$

for which $\kappa_{\rm Ny} = \pi$.

Let us now see when the explicit scheme is stable. Plugging Eq. (12.93) into (12.86), we find

$$Ae^{ikx} = e^{ikx}\left[Ce^{-ik\delta x} + (1-2C) + Ce^{ik\delta x}\right] \ , \tag{12.99}$$

or

$$A \ = \ 1 - 2C + 2C\cos k\delta x \ = \ 1 - 2C(1-\cos\delta x) \ . \tag{12.100}$$

As $1-\cos\xi \le 0$, the amplitude factor can never exceed 1. However, instability occurs also if $A < -1$. The cos function takes its minimum of -1 for $k\delta x = \pi$, which will thus be the least stable Fourier mode. The stability threshold is thus where

$$-1 = 1 - 2C[1-(-1)] \ , \tag{12.101}$$

or $C = \dfrac{1}{2}$.

We thus find that the explicit scheme (12.86) is stable only if the *Courant criterion*

$$C \le \frac{1}{2} \tag{12.102}$$

is met. In practise, C should be chosen well below that threshold. If the Courant criterion is violated, the smallest scales are the most unstable ones. Also, because $A < -1$, the unstable modes will change sign from one time step to the next.

Interpretation of Courant criterion: "Information must not travel more than half a grid cell in one time step." (will become clearer for hyperbolic problems).

Next, let us look at the stability of the fully implicit scheme. Here we find

$$A\left[-Ce^{-ik\delta x} + 1 + 2C - Ce^{ik\delta x}\right] = 1 \ , \tag{12.103}$$

or

$$A = \frac{1}{1 + 2C(1-\cos k\delta x)} \,. \tag{12.104}$$

One can easily see that

$$\frac{1}{1+4C} \leq A \leq 1 \,, \tag{12.105}$$

and thus *the fully implicit scheme is always stable*.

Interpretation: No restriction on information propagation as solving the tridiagonal system propagates information across the whole grid.

Note 1: As A is strictly positive, there is no change of sign for any of the Fourier modes.

Note 2: Stability does not necessarily imply correctness. If we choose a very large time step (large Courant number), we will get an evolution that is considerably different from the exact one. However, the dissipative nature of parabolic problems leads to one finite state (thermal equilibrium) and this often makes the qualitative behaviour similar to the exact solution even when δt is large.

Finally, let us look at the stability of the general implicit scheme. Here,

$$A = \frac{1 - (1-q)2C(1-\cos k\delta x)}{1 + q2C(1-\cos k\delta x)} \,, \tag{12.106}$$

and in particular

$$A = \frac{1 - C(1-\cos k\delta x)}{1 + C(1-\cos k\delta x)} \,, \tag{12.107}$$

for the Crank–Nicholson scheme, and one can show that the general implicit scheme is

$$\begin{cases} \text{unconditionally stable} & \text{for } q \geq 1/2 \,, \\ \text{only stable if } C \leq \dfrac{1}{2(1-2q)} & \text{for } q < 1/2 \,. \end{cases} \tag{12.108}$$

The Crank–Nicholson scheme is thus unconditionally stable.[4]

[4] This result holds for linear stability for constant thermal diffusivity χ. For more complicated settings, the fact that the Crank–Nicholson scheme is just on the border between conditional and unconditional stability may lead to unsatisfactory stability properties. In that case, a more implicit scheme ($q > 1/2$) may be needed. But you should not sacrifice second-order accuracy in time unless you are sure you need to.

12.4.5 Schemes to avoid: the Dufort–Frankel scheme; (in)consistency

There is a great deal of arbitrariness in choosing the discretizations for solving partial differential equations, and seemingly small changes to the scheme can make a huge difference.

As an example, consider the plausible scheme

$$\frac{T_k^{l+1} - T_k^{l-1}}{2\delta t} = \chi \, \frac{T_{k-1}^l - 2T_k^l + T_{k+1}^l}{\delta x^2} \,, \tag{12.109}$$

which is nice because it is of second order in time. Unfortunately, this scheme turns out to be *unconditionally unstable*.

The reason for the instability is related to the fact that even and odd time steps are somewhat decoupled (in the sense that if the left-hand side is used to step from one odd time step to the following one, the right-hand side represents only the even time step in between).

To mitigate this, we can replace T_k^l on the right-hand side by the average $(T_k^{l-1} + T_k^{l+1})/2$:

$$\frac{T_k^{l+1} - T_k^{l-1}}{2\delta t} = \chi \, \frac{T_{k-1}^l - (T_k^{l-1} + T_k^{l+1}) + T_{k+1}^l}{\delta x^2} \,, \tag{12.110}$$

which can be written as

$$T_k^{l+1} = \frac{CT_{k-1}^l + \left(\frac{1}{2} - C\right) T_k^{l-1} + CT_{k+1}^l}{\frac{1}{2} + C} \,. \tag{12.111}$$

Von Neumann Stability analysis gives

$$A = \frac{2C\cos k\delta x \pm \sqrt{1 - 4C^2 \sin^2 k\delta x}}{1 + 2C} \,, \tag{12.112}$$

which turns out to always satisfy $|A| \leq 1$.

However, Eq. (12.111) is an *explicit* equation for T_k^{l+1}. The only additional price we have to pay compared to our explicit scheme is the fact that we need to retain two levels of values (T_k^l and T_k^{l-1}), but then we are rewarded with second-order accuracy in time, and with absolute stability.

So it looks like we have found an explicit scheme that is unconditionally stable! Unfortunately, there is a downside to the Dufort–Frankel scheme that makes me strongly recommend against its use: it solves the wrong equations. To see this, let us expand T around x_k and t_l up to second order:

$$T_{k\pm1}^l = T \pm \partial_x T \delta x + \frac{\partial_x^2 T}{2} \delta x^2 + O\left(\delta x^2\right) \,, \tag{12.113}$$

$$T_k^{l\pm1} = T \pm \partial_t T \delta t + \frac{\partial_t^2 T}{2} \delta t^2 + O\left(\delta t^3\right) \,, \tag{12.114}$$

where we have dropped the indices $()_k^l$ for brevity. Equation (12.111) then becomes

$$\frac{2\partial_t T\,\delta t}{2\delta t} = \chi\frac{2T + \partial_x^2 T\delta x^2 - (2T + \partial_t^2 T\delta t^2)}{\delta x^2} + O\left(\delta t^2\right) + O\left(\delta x^2\right) + O\left(\delta t^4/\delta x^2\right)\ . \tag{12.115}$$

Ignoring the terms that get small when $\delta t \sim \delta x \to 0$, we are left with

$$\partial_t T = \chi\partial_x^2 T - \frac{\chi\,\delta t^2}{\delta x^2}\partial_t^2 T \tag{12.116}$$

This equation is of completely different type than the heat-conduction equation (it is the *telegrapher's equation*, which is hyperbolic), and certainly not what we wanted to solve in the first place.

We thus find that the Dufort–Frankel scheme – although constructed as a discretization of the heat conduction equation – in fact represents the discretization of a different equation. The discretization leading to the scheme was not *consistent*.

If we insist on using the Dufort–Frankel scheme for the heat conduction equation, we will need to make the additional term small. In order to achieve this, δt must decrease faster than δx, and we essentially have the requirement $\delta t \ll \chi\delta x^2$ or $C \ll 1$. The unconditional stability of the scheme does *not* allow us to choose a larger time step than for other explicit schemes.

To gain some peace of mind, let us verify the consistency of our general implicit scheme. Here we get

$$\frac{\partial_t T\delta t + \dfrac{\partial_t^2 T}{2}\delta t^2}{\delta t} = \chi\frac{\partial_x^2 T\delta x^2 + O\left(\delta x^2\delta t\right) + O\left(\delta x^4\right)}{\delta x^2}\ , \tag{12.117}$$

and hence

$$\partial_t T = \chi\partial_x^2 T + O\left(\delta t\right) + O\left(\delta x^2\right)\ . \tag{12.118}$$

In the limit of both time step and grid spacing tending to zero, we obtain exactly the heat conduction equation, thus the general implicit scheme is consistent, together with all of its special cases like the explicit scheme, the fully implicit scheme, and the Crank–Nicholson scheme.

12.4.6 Boundary conditions

A Dirichlet boundary condition $T_0 = a(t)$ translates into the first line of

$$\begin{pmatrix} 1 & 0 & 0 & 0 & \dots & 0 \\ -qC & 1{+}2qC & -qC & 0 & \dots & 0 \\ 0 & -qC & 1{+}2qC & -qC & \dots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \end{pmatrix}\begin{pmatrix} T_0^{l+1} \\ T_1^{l+1} \\ T_2^{l+1} \\ \vdots \end{pmatrix} = \begin{pmatrix} a(t_{l+1}) \\ (1{-}q)CT_0^l + [1{-}2(1{-}q)C]T_1 + (1{-}q)CT_2^l \\ (1{-}q)CT_1^l + [1{-}2(1{-}q)C]T_2 + (1{-}q)CT_3^l \\ \vdots \end{pmatrix} \tag{12.119}$$

A von Neumann boundary condition $(\partial T/\partial x)_0 = b(t)$ can be discretized as

$$\frac{T_1 - T_0}{\delta x} = b(t) \tag{12.120}$$

with first-order accuracy.[5] This maps to the first line of

$$\begin{pmatrix} 1 & -1 & 0 & 0 & \dots & 0 \\ -qC & 1{+}2qC & -qC & 0 & \dots & 0 \\ 0 & -qC & 1{+}2qC & -qC & \dots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \end{pmatrix} \begin{pmatrix} T_0^{l+1} \\ T_1^{l+1} \\ T_2^{l+1} \\ \vdots \end{pmatrix} = \begin{pmatrix} -\delta x\, b(t_{l+1}) \\ (1{-}q)CT_0^l + [1{-}2(1{-}q)C]T_1 + (1{-}q)CT_2^l \\ (1{-}q)CT_1^l + [1{-}2(1{-}q)C]T_2 + (1{-}q)CT_3^l \\ \vdots \end{pmatrix} \tag{12.121}$$

12.4.7 Non-homogeneous equation

It is pretty straight-forward to adapt our schemes for the non-homogeneous equation

$$\frac{\partial T}{\partial t} - \chi \frac{\partial^2 T}{\partial x^2} = h(x,t)\ . \tag{12.122}$$

The general implicit scheme takes the form

$$\begin{aligned} -qCT_{k-1}^{l+1} + (1{+}2qC)T_k^{l+1} - qCT_{k+1}^{l+1} \;=\; & (1{-}q)CT_{k-1}^l + [1-2(1{-}q)C]T_k^l + (1{-}q)CT_{k+1}^l \\ & + \delta t\, h(x_k, t_{l+1/2})\ , \end{aligned} \tag{12.123}$$

which is still a tridiagonal system as before.

There is some freedom in the choice of the time argument for h, and we could just as well have chosen $h(x_k, t_l)$ or $h(x_k, t_{l+1})$. The present choice [to evaluate $h(x,t)$ at the time $t_{l+1/2} \equiv t_l + \delta t/2$] has the advantage that the Crank–Nicholson scheme

$$-\frac{C}{2}T_{k-1}^{l+1} + (1{+}C)T_k^{l+1} - \frac{C}{2}T_{k+1}^{l+1} \;=\; \frac{C}{2}T_{k-1}^l + (1{-}C)T_k^l + \frac{C}{2}T_{k+1}^l + \delta t\, h(x_k, t_{l+1/2})\ , \tag{12.124}$$

will still be of second order in time.

12.4.8 Higher-order explicit schemes

The accuracy of the schemes used so far was second order in space and first or second order in time. Many people are content with this – but they should not, as it is relatively simple to use higher-order schemes and increase accuracy. One reason not to do so may be

[5] It can be shown that an $N-1$th order boundary scheme is consistent with an Nth order scheme for interior points. Thus, the Crank–Nicholson scheme together with a first-order boundary scheme will still be of second order in space and time.

the better stability properties of implicit methods, which are more difficult to implement in higher order. Another complication with higher-order methods are often the boundary conditions, although most of this can be dealt with elegantly using ghost zones (see 12.5.2). But for explicit time stepping with periodic boundary conditions there is definitively no excuse for not using high-order schemes.

As high-order methods are going to be our favourite method for solving hyperbolic equations (which we will in fact turn into parabolic ones), we leave the discussion for Section 12.5.2 below.

12.5 Hyperbolic problems

The standard example of a hyperbolic equation is the wave equation

$$\frac{\partial^2 f}{\partial t^2} = c^2 \frac{\partial^2 f}{\partial x^2} \; . \tag{12.125}$$

It describes waves moving with phase velocity c in both directions and has the general solution

$$f(x,t) = \varphi_{\rm r}(x-ct) + \varphi_{\rm l}(x+ct) \; . \tag{12.126}$$

The wave operator can be factored as

$$\left(\frac{\partial}{\partial t} - c\frac{\partial}{\partial x}\right)\left(\frac{\partial}{\partial t} + c\frac{\partial}{\partial x}\right) f = 0 \; , \tag{12.127}$$

and if we are only interested in signals propagating to the right, we can drop the first differential operator and get

$$\left(\frac{\partial}{\partial t} + c\frac{\partial}{\partial x}\right) f = 0 \; , \tag{12.128}$$

or

$$\frac{\partial f}{\partial t} = -u\frac{\partial f}{\partial x} \; , \tag{12.129}$$

where we have replaced the symbol c with u. This equation has the general solution

$$f(x,t) = \varphi_{\rm r}(x-ut) \; . \tag{12.130}$$

The advection equation (12.129) is *not* a hyperbolic equation, as it is only of first order. Nevertheless it will be our prototype for this section, as it has all the properties of hyperbolic systems that are relevant for us. Also, note that Eq. (12.127) implies that the wave equation can be written as a system of two coupled advection equations

$$\frac{\partial f}{\partial t} = u\frac{\partial g}{\partial x} \; , \tag{12.131}$$

$$\frac{\partial g}{\partial t} = u\frac{\partial f}{\partial x} \; . \tag{12.132}$$

12.5.1 Low-order schemes

A straight-forward discretization of the advection equation

$$\frac{\partial f}{\partial t} = -u\frac{\partial f}{\partial x} \,, \tag{12.133}$$

is

$$\frac{f_k^{l+1} - f_k^l}{\delta t} = -u\,\frac{f_{k+1}^l - f_{k-1}^l}{2\delta x} \,, \tag{12.134}$$

which is first-order accurate in time and second-order accurate in space. The scheme turns out to be consistent.

Von Neumann stability analysis gives

$$\frac{A-1}{\delta t} = -u\frac{i\,\sin k\delta x}{\delta x} \,, \qquad \text{or} \qquad A = 1 - i\,C_{\rm adv}\sin k\delta x \,, \tag{12.135}$$

where

$$C_{\rm adv} \equiv \frac{u\delta t}{\delta x} \tag{12.136}$$

is the *advective Courant number*; we will mostly omit the index 'adv', unless this can lead to confusion with the diffusive Courant number.

The modulus of the amplification factor A is thus

$$|A|^2 = 1 + C^2\sin^2 k\delta x \geq 1 \,, \tag{12.137}$$

and thus, unfortunately, this scheme is unconditionally unstable. As [NR77] put it:

> *"The resulting finite-difference approximation [. . .] is called the FTCS representation (Forward Time Centred Space) [. . .] It's a fine example of an algorithm that is easy to derive, takes little storage, and executes quickly. Too bad it doesn't work!"*

The instability is illustrated in Figure 12.1.

The Lax scheme

Replacing f_k^l by $(f_{k-1}^l + f_{k+1}^l)/2$ on the left-hand side of Eq. (12.134), we obtain the *Lax scheme*

$$\frac{f_k^{l+1} - \dfrac{f_{k-1}^l + f_{k+1}^l}{2}}{\delta t} = -u\frac{f_{k+1}^l - f_{k-1}^l}{2\delta x} \,, \tag{12.138}$$

or

$$f_k^{l+1} = \frac{f_{k-1}^l + f_{k+1}^l}{2} - C\,\frac{f_{k+1}^l - f_{k-1}^l}{2} \,. \tag{12.139}$$

The amplification factor is

$$A = \cos k\delta x - i\,C\sin k\delta x \,, \tag{12.140}$$

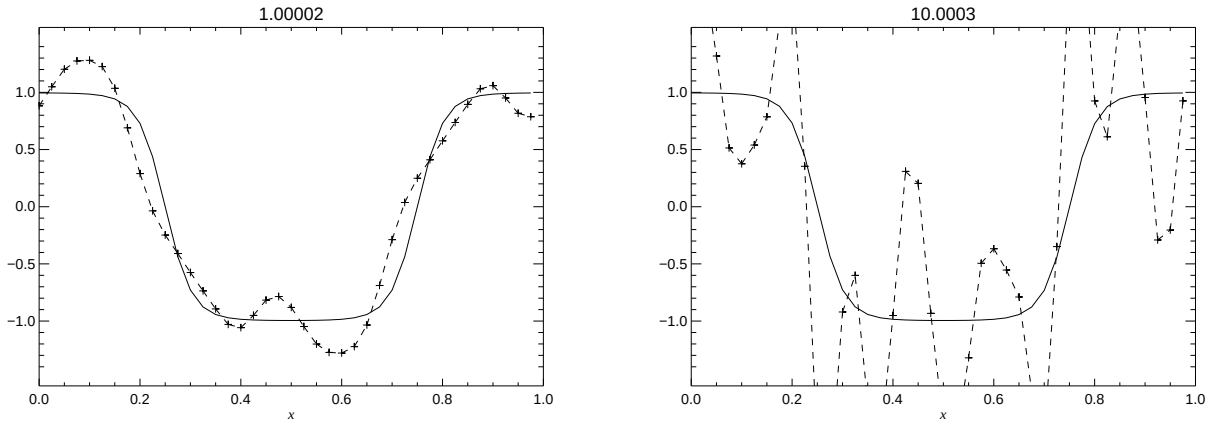


Figure 12.1: The $O\left(\delta t, \delta x^2\right)$ (first-order time step, second-order spatial derivatives) scheme applied to the advection problem (12.129) with $u = 1$ and periodic boundary conditions. The time step is extremely small ($\delta t = 0.0005$). The solid line shows the exact solution (identical to the initial profile), while the crosses and dashed line show the numerical solution. Left: t=1 (i.e. the pattern has travelled once through the interval $[0, 1]$. Right: t=10 (the pattern has travelled ten times through the interval).

and

$$|A|^2 \;=\; \cos^2 k\delta x + C^2 \sin^2 k\delta x \;=\; 1 + (C^2{-}1) \sin^2 k\delta x \;. \tag{12.141}$$

The Lax scheme is thus conditionally stable, and the stability criterion is the *Courant–Friedrichs–Lewy stability criterion* (or simply *Courant criterion*)

$$C \leq 1 \;, \tag{12.142}$$

which holds in a similar form for most or all explicit schemes.

Interpretation of the Courant criterion: the scheme is only stable if information (which travels at speed u) crosses no more than one cell per time step. This seems pretty intuitive, as during one step, our scheme, which extends just one point to the left and one to the right, does not have access to information from further away. On the other hand, we should remember that von Neumann stability analysis is base on the *global* Fourier modes and their evolution is governed by the *phase velocity*. Thus, the propagation of information (which propagates at the *group velocity*) does probably not give a consistent interpretation.

Why is the Lax scheme stable (provided δt satisfies the Courant condition), while (12.134) is not? The answer becomes clear if we write the Lax scheme in the form

$$\frac{f_k^{l+1} - f_k^l}{\delta t} = -u \frac{f_{k+1}^l - f_{k-1}^l}{2\delta x} + \frac{f_{k-1}^l - 2f_k^l + f_{k+1}^l}{2\delta t} \;. \tag{12.143}$$

This suggests that the Lax scheme is the scheme (12.134) applied to the equation

$$\frac{\partial f}{\partial t} \;=\; -u \frac{\partial f}{\partial x} + \frac{\delta x^2}{2\delta t} \frac{\partial^2 f}{\partial x^2} \tag{12.144}$$

$$\;=\; -u \frac{\partial f}{\partial x} + \frac{\delta x}{2C} \frac{\partial^2 f}{\partial x^2} \;. \tag{12.145}$$

Additional second-derivative terms like the one appearing here are called *numerical diffusivity* (or *numerical viscosity*) terms. They damp the largest wave numbers (that we will not get right anyway), but have relatively little impact on the small wave numbers (large scales) that we are interested in.

Is the Lax scheme consistent? Equation (12.144) tells us that it is generally not. However, one will typically us a time step that is slightly (by a factor of 1/2 or so) below the Courant threshold, thus keeping C constant when reducing the grid size δx. In this case, as Eq. (12.145) shows, the additional term tends to zero as δx does. Thus, in practise, the Lax scheme is (often) consistent, but when choosing very small time steps, it is not.

The upwind scheme

The advection equation only advects information in one direction (the positive x direction if $u > 0$). The Lax scheme, however, takes into account information from both neighbouring points, which is part of the reason why its numerical diffusivity is quite large. We can avoid this by discretizing

$$\frac{f_k^{l+1} - f_k^l}{\delta t} = \begin{cases} -|u| \, \dfrac{f_{k+1}^l - f_k^l}{\delta x} \,, & \text{for } u < 0 \\ -|u| \, \dfrac{f_k^l - f_{k-1}^l}{\delta x} \,, & \text{for } u > 0 \end{cases} \tag{12.146}$$

This is called the *upwind scheme* (because only information from the upwind or upstream direction is used); it is first-order accurate in time and space and is stable, provided that

$$C \leq 1 \,. \tag{12.147}$$

The upwind scheme can be written as

$$\frac{f_k^{l+1} - f_k^l}{\delta t} = -u \, \frac{f_{k+1}^l - f_{k-1}^l}{2\delta x} + |u| \, \frac{f_{k-1}^l - 2f_k^l + f_{k+1}^l}{2\delta x} \,. \tag{12.148}$$

It discretizes

$$\frac{\partial f}{\partial t} = -u \, \frac{\partial f}{\partial x} + \frac{|u| \, \delta x}{2} \, \frac{\partial^2 f}{\partial x^2} + O(\delta t) + O\left(\delta x^2\right) \,. \tag{12.149}$$

We thus again have numerical diffusivity in the scheme, but this time the scheme is consistent for any (admissible) values of δt and δx. Another advantage over the Lax scheme is that if $u = u(x)$, viscosity is only applied "where it is needed", i.e. there is little diffusion in regions where $|u|$ is small.

And yet, the upwind scheme is only first-order accurate in space (and time), and its numerical diffusivity is still relatively large (where $|u|$ is large). Higher-order methods are much better.

More low-order schemes

There are a number of schemes designed to be less diffusive than the Lax or the upwind scheme and some of them are of second order in both time and space (e.g. the Lax–Wendroff scheme or the staggered leapfrog scheme). We will not discuss them here (as we will use something better), and just refer to [NR77] here.

TVD schemes

Another class of schemes we mention only briefly are the *total variation diminishing* (TVD) schemes. These are schemes that use a nonlinear filter to minimize the total variation

$$V[f] \equiv \sum |f_i - f_{i-1}| \tag{12.150}$$

due to the small scales[6]. This suppresses the formation of Nyquist zigzags that are in many cases the source of instability.

These schemes give impressive results for simple test problems (and can be applied for real-life applications), but the nonlinear character of the filtering makes them somewhat dubious and very difficult to analyze.

Conservative schemes

Many schemes use the fact that advection-type equations can be written in the form

$$\frac{\partial f}{\partial t} = -\nabla(\mathbf{F}) \,, \tag{12.152}$$

where $\mathbf{F}$ is a *flux function*. In our example, $F = uf$ if u is constant. This representation is called *conservation form*, as it is related to conservation laws (e.g. for momentum and total energy), and the class of *conservative schemes* employs this form to obtain exact conservation of these quantities (up to round-off error). Note that the conservation form of equations can be quite unnatural compared to the *primitive equations* (continuity, Navier-Stokes, induction, . . .) that we are used to deal with.

While it may sound impressive that these schemes manage to exactly conserve energy and momentum, it must be kept in mind that these conservation properties just constrain the trajectory of the system to a 6^N-4-dimensional hypersurface in phase space if N is

[6] For the one-dimensional advection problem (12.129) [but with possibly space-and time-dependent advection velocity $u(x,t)$], the analytical total variation

$$V[f] \equiv \int \left|\frac{df}{dx}\right| dx \tag{12.151}$$

[for which Eq. (12.150) is a discretization] is known to be constant. This is because the effect of advection is to displace and stretch or compress the initial profile, which does not change the difference between successive minima and maxima.

However, in more than one dimension, or for equations more complicated than the plain advection equation, there is no analog for this property.

the number of grid points. To get the correct trajectory, one needs another 6^N-3 conservation properties, none of which is known. Thus, while conservative schemes are good at conserving a few known properties, they can be just as bad at getting the phase-space trajectory right as are other schemes of the same order.

And while non-conservative schemes allow checking the known conservation laws as a simple accuracy test, conservative schemes provide no such straight-forward quality measure.

12.5.2 Higher-order schemes

The basic idea of higher-order explicit schemes is quite simple: Treat time and spatial discretization completely separately. We thus use a high-order spatial discretization like Eq. (12.54) for the right-hand side. Once we know how to calculate the RHS, we can apply Runge–Kutta for time stepping.

The scheme we are going to use is third-order in time and sixth-order in space.

Spectral characteristics of finite-difference stencils

Important insight into the properties of a finite-difference scheme is obtained by applying it to a harmonic function

$$f(x) = e^{ikx} , \qquad \text{where again } 0 \le |k| \le k_{\rm Ny} . \tag{12.153}$$

Applying the exact first and second derivative operators to the harmonic function (12.153) would yield

$$\partial_x e^{ikx} = ike^{ikx} , \qquad \partial_x^2 e^{ikx} = -k^2 e^{ikx} , \tag{12.154}$$

thus the *spectral transfer functions*

$$H^{(1)}(k) \equiv e^{-ikx} D^{(1)} e^{ikx} , \tag{12.155}$$

$$H^{(2)}(k) \equiv e^{-ikx} D^{(2)} e^{ikx} \tag{12.156}$$

indicate the quality of the finite-difference approximations $D^{(1)}$, $D^{(2)}$: for exact derivatives $D^{(1)} = D, D^{(2)} = D^2$ one would get [7]

$$H^{(1)}(k) = ik , \qquad H^{(2)}(k) = -k^2 . \tag{12.157}$$

Finite-difference approximations $D^{(1)}$ and $D^{(2)}$ to the first and second order derivatives will give rise to deviations,

$$H^{(1)}(k) = ik\Theta , \qquad H^{(2)}(k) = -k^2\Theta^{(2)} , \tag{12.158}$$

[7] Such exact numerical derivative operators are indeed implemented by *spectral schemes* which apply a Fourier transform, multiply in Fourier space by ik or $-k^2$, and then transform back.

where Θ and $\Theta^{(2)}$ are very close to 1 for small wave numbers (large scales), but differ strongly from 1 near the Nyquist wave number.

For the second-order first derivative operator (12.49), we get

$$H^{(1)}(k) \;=\; ik\,\frac{\sin\kappa}{\kappa} \;=\; ik\left(1-\frac{\kappa^2}{6}+\ldots\right), \tag{12.159}$$

where $\kappa \equiv k\,\delta x$. For the fourth-order operator (12.50), we get

$$H^{(1)}(k) \;=\; ik\,\frac{8\sin\kappa-\sin 2\kappa}{6\kappa} \;=\; ik\left(1-\frac{\kappa^4}{30}+\ldots\right), \tag{12.160}$$

and for sixth order (12.51)

$$H^{(1)}(k) \;=\; ik\,\frac{45\sin\kappa-9\sin 2\kappa+\sin 3\kappa}{30\kappa} \;=\; ik\left(1-\frac{\kappa^6}{140}+\ldots\right). \tag{12.161}$$

Note that all of these expressions become zero at the Nyquist frequency. This is unavoidable for centred finite-difference operators on a non-staggered grid.

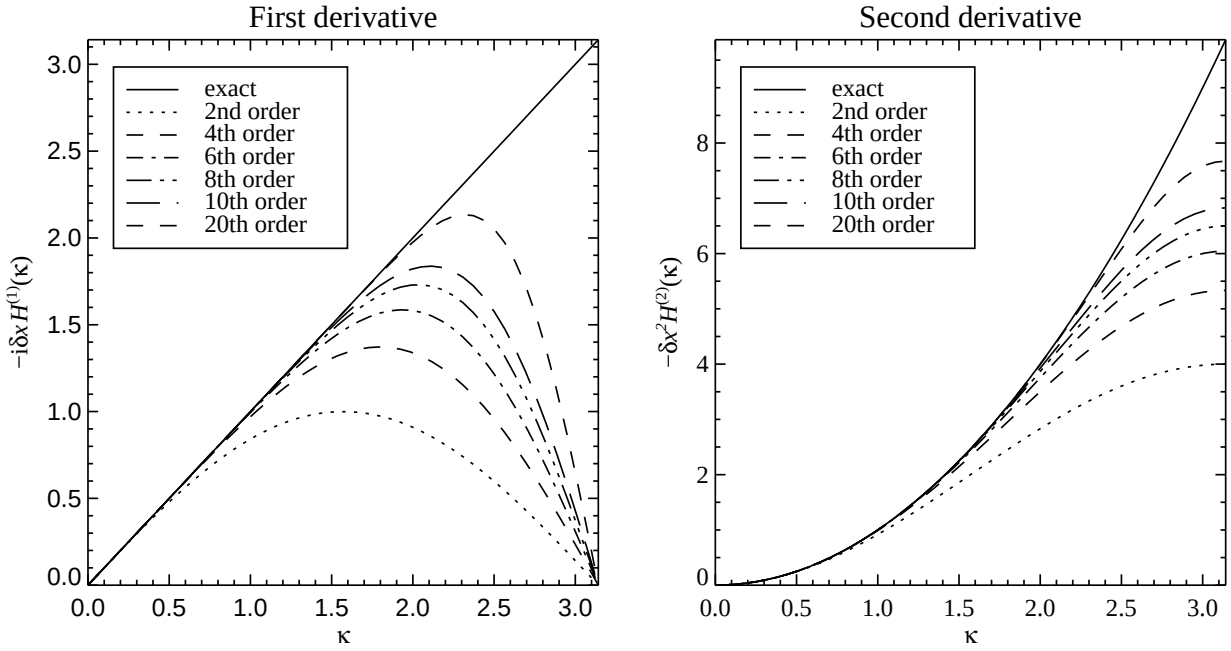


Figure 12.2: Spectral transfer functions $H(k\delta x) \equiv e^{-ik}De^{ik}$ as a function of $\kappa = k\delta x$ for centred finite-difference schemes of different orders. Left: transfer function for the first derivative operator, $D^{(1)}$, multiplied by $-i\,\delta x$. Right: spectral transfer function for the second derivative operator, $D^{(2)}$, multiplied by $-\delta x^2$. The solid lines show the transfer function of the exact derivative operator (which is reproduced by spectral schemes).

Similarly, for the second-order second derivative operator (12.52), we get

$$H^{(2)}(k) \;=\; -k^2\,2\frac{1-\cos\kappa}{\kappa^2} \;=\; -k^2\left(1-\frac{\kappa^2}{12}+\ldots\right), \tag{12.162}$$

for fourth order (12.53)

$$H^{(2)}(k) = -k^2 \, \frac{15 - 16\cos\kappa + \cos 2\kappa}{6\,\kappa^2} \;=\; -k^2 \left(1 - \frac{\kappa^4}{9} + \ldots\right) , \tag{12.163}$$

and for sixth order (12.54)

$$H^{(2)}(k) = -k^2 \, \frac{245 - 270\cos\kappa + 27\cos 2\kappa - 2\cos 3\kappa}{90\,\kappa^2} \;=\; -k^2 \left(1 - \frac{\kappa^6}{560} + \ldots\right) . \tag{12.164}$$

Figure 12.2 shows the spectral transfer functions for a number of schemes from order 2 up to 20. One can easily see how all schemes yield good approximations to the exact derivative for small k, but for intermediate wave numbers (say, half the Nyquist wavenumber $\kappa_{\mathrm{Ny}} = k_{\mathrm{Ny}}\delta x = \pi$) only higher orders reproduce the exact derivatives with sufficient accuracy.

Note: The fact that the numerical first derivative of a Nyquist signal is zero has several consequences. First, we note that for a Nyquist signal there is no propagation due to the advection term $uD^{(1)}$, thus signals with wave numbers κ close to π will quickly get out of phase with the larger-scale signals (see *phase error* in Sec. 12.5.2 below).

Second, an iterated first derivative $D^{(1)}D^{(1)}f$ will be zero for a Nyquist signal, and thus not have any damping effect, while the second derivative $D^{(2)}f$ gives rise to dissipative damping. As a consequence, any iterated first derivative term on the right-hand side of our partial differential equations *must* be re-written in terms of a second derivative. E.g. the heat-conduction term $\partial_x(\lambda\partial_x T)$ must be used in the form $\partial_x\lambda\,\partial_x T + \lambda\partial_x^2 T$. If this rule is not applied, calculations typically work reasonably well for some time, but Nyquist signals slowly grow in amplitude (due to boundary effects and nonlinearities) and eventually make the numerical solution unusable.

Stability

Under the advection equation, a harmonic profile

$$f(x,0) = e^{ikx} \tag{12.165}$$

evolves as

$$f(x,t) = e^{ik(x-ut)} = e^{-ikut} f(x,0) , \tag{12.166}$$

thus the exact amplification factor is

$$A_{\mathrm{exact}} = e^{-iku} . \tag{12.167}$$

The amplification factor of the discretized scheme will differ from this value:

$$A = e^{-iuk+\gamma+i\omega} , \qquad \gamma, \omega \in \mathbb{R} . \tag{12.168}$$

The quantity γ is a growth rate and gives rise to the *amplitude error*, while ω represents the *phase error*.

The question of stability boils down to the sign of γ. If $\gamma > 0$ for some modes, then the energy in these modes will grow and eventually dominate the solution and render it useless. Reducing the *modulus* of γ in this case will not remove the instability — it only increases the time for which it can be ignored.

If $\gamma < 0$, on the other hand, energy in the corresponding modes will decrease. This implies that there is some *numerical dissipation* at work, but normally this only affects the smaller scales. By decreasing the time step, both amplitude and phase error will be decreased, so if $\gamma \leq 0$ for all modes, one can control the errors by adjusting the time step δt.

Similar conclusions can be drawn for the diffusion term, which has only an amplitude error γ_{diff}. Here, however, instability will only occur if γ_{diff} overcomes the natural decay of the modes.

Table 12.1: Leading-order terms (in δt) of the growth rate γ and phase drift ω for time-stepping schemes of different order m. The quantities Θ and $\Theta^{(2)}$ measure the quality of the spatial schemes. Note that in the absence of diffusion $\gamma < 0$ (indicating stability) only for $m = 3, 4; 7, 8; 11, 12 \ldots$.

m	γ	ω	γ_{diff}
1	$\frac{(uk\Theta)^2}{2}\delta t$	$uk(1-\Theta) + \frac{(uk\Theta)^3}{3}\delta t^2$	$\nu k^2(1-\Theta^{(2)}) - \frac{\nu^2 k^4 {\Theta^{(2)}}^2}{2}\delta t$
2	$\frac{(uk\Theta)^4}{8}\delta t^3$	$uk(1-\Theta) - \frac{(uk\Theta)^3}{6}\delta t^2$	$\nu k^2(1-\Theta^{(2)}) + \frac{\nu^3 k^6 {\Theta^{(2)}}^3}{6}\delta t^2$
3	$-\frac{(uk\Theta)^4}{24}\delta t^3$	$uk(1-\Theta) - \frac{(uk\Theta)^5}{30}\delta t^4$	$\nu k^2(1-\Theta^{(2)}) - \frac{\nu^4 k^8 {\Theta^{(2)}}^4}{24}\delta t^3$
4	$-\frac{(uk\Theta)^6}{144}\delta t^5$	$uk(1-\Theta) + \frac{(uk\Theta)^5}{120}\delta t^4$	$\nu k^2(1-\Theta^{(2)}) + \frac{\nu^5 k^{10} {\Theta^{(2)}}^5}{120}\delta t^4$

Table 12.1 shows the leading order in δt of the amplitude and phase errors for time-stepping schemes of orders 1 to 4. Without artificial diffusivity, only the third- and fourth-order schemes are stable ($\gamma < 0$), provided that the time step is sufficiently small (see § 12.5.2 below). Although this result is formulated for the advection problem (12.129), exactly the same stability conditions hold in the case of linear sound waves

$$\partial_t \ln \varrho = -\partial_x v \tag{12.169}$$

$$\partial_t v = -c_s^2 \partial_x \ln \varrho \tag{12.170}$$

if the advection speed u is replaced by the speed of sound c_s. In the case of sound waves in a medium that moves at speed u, the relevant velocity is $\max(|u \pm c_s|)$.

To conclude, we can say that

> *For advection and similar problems the amplitude error, and thus the stability of the scheme, is determined by the time-stepping scheme, while the phase error is normally dominated by the spatial discretization.*

Artificial viscosity

When solving partial differential equations that are more realistic than our simple advection problem, even third- and fourth order time-stepping schemes require some amount of diffusivity/viscosity due to boundary effects, nonlinearities or just to minimize the consequences of the phase error. Like in the case discussed above, this viscosity will always tend to zero for $\delta x \to 0$. The recommended minimum value of viscosity for the $O(\delta t^3, \delta x^6)$ scheme is[8]

$$\nu = c_\nu \, U_{\max} \, \delta x \tag{12.171}$$

where $U_{\max}$ is the largest velocity in the problem (including propagation speeds of waves), and $c_\nu = 0.01 \,..\, 0.02$.

The length of the time step

Even the explicit schemes labelled as 'stable' are only stable if the time step δt satisfies a Courant condition of the form

$$\delta t \le c_{\text{adv}} \, \frac{\delta x}{u} \qquad \text{or} \qquad \delta t \le c_{\text{dif}} \, \frac{\delta x^2}{\nu} \; . \tag{12.172}$$

For $O(\delta t^3, \delta x^6)$ schemes, the stability boundary is $c_{\text{adv}} = 1.092$ and $c_{\text{dif}} = 0.4157$. For $O\left(\delta t^4, \delta x^6\right)$ schemes, we have $c_{\text{adv}} = 1.783$ and $c_{\text{dif}} = 0.4608$. In practise one should use a time step considerably smaller than the stability limit ($C = 0.5$ or smaller), since at the very limit nonlinearities and boundaries can destabilize the configuration.

Other propagation velocities (like the sound speed) will give rise to similar time step restrictions. The recommended time step for the $O(\delta t^3, \delta x^6)$ scheme is

$$\delta t = \min\left(0.4 \, \delta x / U_{\max}, \; 0.08 \, \delta x^2 / \nu_{\max}\right) \; , \tag{12.173}$$

where $U_{\max}$ is the largest velocity[9] in the problem and $\nu_{\max}$ the largest diffusivity.

[8] This dependence on velocity and grid spacing is reminiscent of the upwind scheme (Sec. 12.5.1), but for higher-order schemes we need much less diffusivity.

[9] Again, $U_{\max}$ includes propagations speeds of waves and also drift terms like $\partial_x \lambda$ in $\partial_x(\lambda \partial_x T) = \partial_x \lambda \, \partial_x T + \lambda \partial_x^2 T$.

Our standard scheme

This box summarizes the properties of the scheme we normally use to solve partial differential equations.

- 6th-order spatial derivative operators:

$$D^{(1)}f = \frac{-f_{k-3} + 9f_{k-2} - 45f_{k-1} + 45f_{k+1} - 9f_{k+2} + f_{k+3}}{60\,\delta x} \tag{12.174}$$

$$D^{(2)}f = \frac{2f_{k-3} - 27f_{k-2} + 270f_{k-1} - 490f_k + 270f_{k+1} - 27f_{k+2} + f_{k+3}}{180\,\delta x^2} \tag{12.175}$$

- Artificial viscosity:

$$\nu = c_\nu U_{\max} \delta x$$

with $c_\nu = 0.01 \ldots 0.02$ (or physical viscosity at least that large);

- 3rd-order Runge–Kutta time stepping (this is a memory-efficient version of third-order Runge–Kutta, although this is far from being obvious here):

$$\begin{array}{c|ccc} 0 & & & \\ \frac{8}{15} & \frac{8}{15} & & \\ \frac{2}{3} & \frac{1}{4} & \frac{5}{12} & \\ \hline & \frac{1}{4} & 0 & \frac{3}{4} \end{array} \tag{12.176}$$

applied to solve

$$\frac{df_k}{dt} = -uD^{(1)}f_k + \nu D^{(2)}f_k \,. \tag{12.177}$$

- Time step:

$$\delta t = \min\left(0.4\,\frac{\delta x}{U_{\max}},\ 0.08\,\frac{\delta x^2}{\nu_{\max}}\right) .$$

Boundary conditions

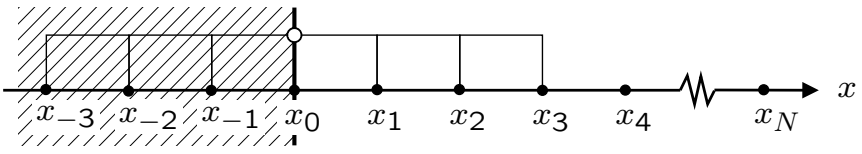


Figure 12.3: Sketch of ghost zones for a seven-point finite-difference stencil on a grid ranging from x_0 to x_N.

So far, our discussion was implicitly assuming that boundary conditions are periodic (or

that the interval in x is unbounded). In real life, one often has to use other boundary conditions. We will discuss this just briefly, restricting ourselves to boundary conditions implemented by setting *ghost zone* values. A ghost zone is a layer of fictitious points beyond the boundary which is introduced so that wide finite-difference stencils can be applied even close to the boundary. For our sixth-order (seven-point) stencil, we need three points on each side of the given point, thus we will need three ghost layers on each side if we want to be able to calculate derivatives in the very boundary points. This situation is depicted in Fig. 12.3.

When ghost zones are used, the boundary conditions provide a rule how to set the values in the ghost points. We just present four popular choices of boundary conditions that can be thus implemented:

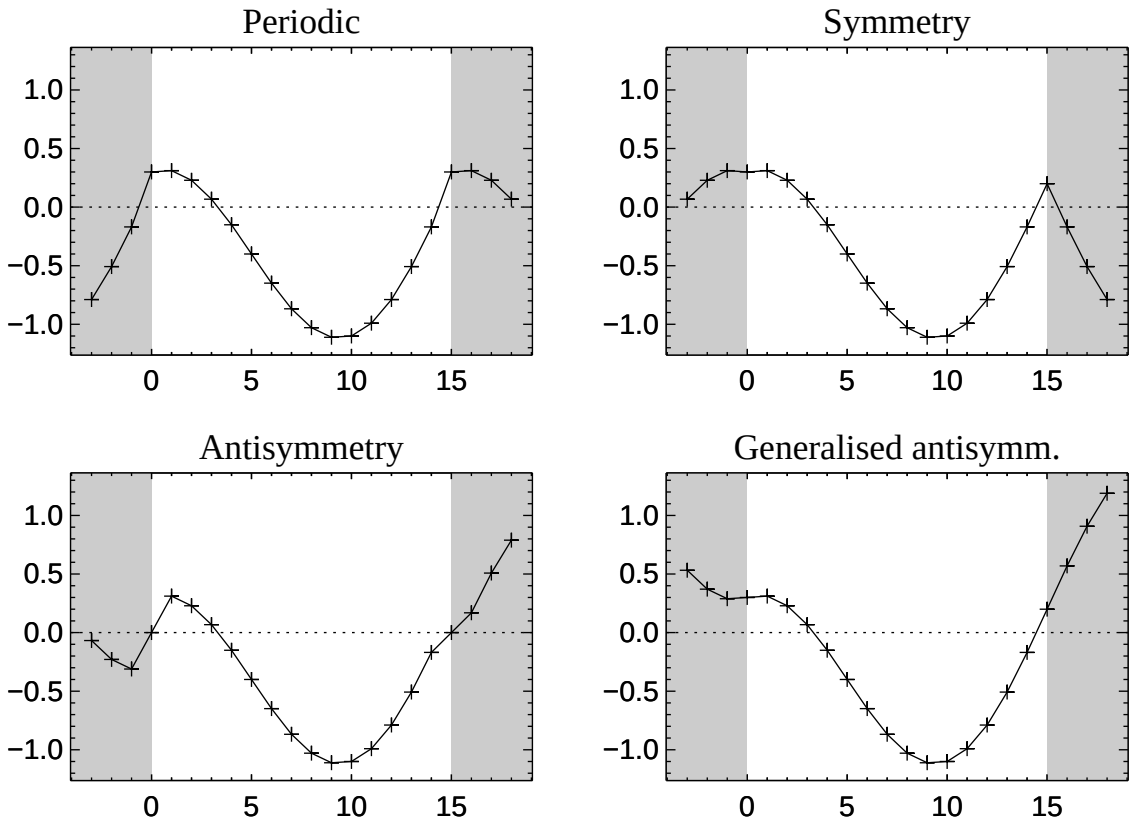


Figure 12.4: The four boundary conditions discussed in the text, applied to an arbitrarily chosen function. The shaded regions to the left and right are the 'ghost zones'.

1. Periodic boundary conditions

$$y_{-1} = y_{N-1}\,, \qquad y_{-2} = y_{N-2}\,, \qquad y_{-3} = y_{N-3}\,. \tag{12.178}$$

2. Symmetry ($y_0' = 0$)

$$y_{-1} = y_1\,, \qquad y_{-2} = y_2\,, \qquad y_{-3} = y_3\,. \tag{12.179}$$

3. Antisymmetry ($y_0 = 0$)

$$y_{-1} = -y_1 \,, \qquad y_{-2} = -y_2 \,, \qquad y_{-3} = -y_3 \,. \tag{12.180}$$

4. Generalized antisymmetry ($y_0'' = 0$)

$$y_{-1} = 2y_0 - y_1 \,, \qquad y_{-2} = 2y_0 - y_2 \,, \qquad y_{-3} = 2y_0 - y_3 \,. \tag{12.181}$$

Figure 12.4 illustrates these four boundary conditions.

Note that symmetry also implies that $y_0^{(3)} = y_0^{(5)} = \ldots = 0$, and similarly antisymmetry and generalized antisymmetry imply $y_0^{(2)} = y_0^{(4)} = \ldots = 0$. These additional conditions are often compatible with the physical problem (some heat-conduction problems are solved using symmetry conditions like this), but this will not generally be the case. If they are not, the overall order of the numerical scheme is reduced to second order in space, which sounds like a dramatic change for the worse. In practise, however, we find that these boundary conditions give quite good results (the coefficient in front of the $O(\delta x^2)$ error term must be small), and the resulting schemes have very good stability properties.

Application I: Sound waves

Sound waves are propagating pressure perturbations arising from the interaction of velocity, density and pressure.

Continuity equation:

$$\frac{\partial \varrho}{\partial t} + \nabla \cdot (\varrho \mathbf{v}) = 0 \,, \tag{12.182}$$

or

$$\frac{\partial \varrho}{\partial t} + \mathbf{v} \cdot \nabla \varrho = -\varrho \nabla \cdot \mathbf{v} \,, \tag{12.183}$$

which we can write as

$$\frac{D \ln \varrho}{Dt} = -\nabla \cdot \mathbf{v} \,, \tag{12.184}$$

where

$$\frac{D}{Dt} \equiv \frac{\partial}{\partial t} + \mathbf{v} \cdot \nabla \tag{12.185}$$

is the *advective derivative* (sometimes also called *convective* or *Lagrangian derivative*).

Equation of motion: The *Navier–Stokes equation*

$$\frac{D\mathbf{v}}{Dt} = -\frac{\nabla P}{\varrho} + \nu \left(\Delta \mathbf{v} + \frac{1}{3} \nabla \nabla \cdot \mathbf{v} \right) + \mathbf{f}_{\text{ext}} \tag{12.186}$$

describes momentum conservation (although this is not obvious from this form of the equation). The above form holds for a compressible fluid if the kinematics viscosity ν is constant.

Pressure term: If there is a unique relation between P and ϱ — for example an adiabatic, polytropic or isothermal equation of state — we can define the sound speed [10]

$$c_s^2 \equiv \frac{dP}{d\varrho} \,. \tag{12.189}$$

This allows us to rewrite the pressure term as follows

$$-\frac{1}{\varrho}\nabla P = -\frac{c_s^2}{\varrho}\nabla\varrho = -c_s^2 \nabla \ln \varrho \,. \tag{12.190}$$

Thus, in terms of logarithmic density, our equations become

$$\begin{aligned} \frac{\partial \ln \varrho}{\partial t} + \mathbf{v}\cdot\nabla \ln \varrho &= -\nabla\cdot\mathbf{v} \qquad (12.191)\\ \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v}\cdot\nabla)\mathbf{v} &= -c_s^2\nabla \ln \varrho + \nu\left(\Delta\,\mathbf{v} + \frac{1}{3}\nabla\nabla\cdot\mathbf{v}\right) + \mathbf{f}_{\rm ext} \,. \end{aligned}$$

One-dimensional case

For a one-dimensional flow $\mathbf{v} = [v(x,t), 0, 0]$, $\varrho = \varrho(x,t)$, Eqs. (12.191) and (12.192) simplify to

$$\partial_t \ln \varrho = -v\,\partial_x \ln \varrho - \partial_x v \,, \tag{12.192}$$

$$\partial_t v = -v\partial_x v - \frac{\partial_x P}{\varrho} + \frac{4}{3}\nu\partial_x^2 v \tag{12.193}$$

Sound waves

If we linearize equations (12.191), (12.192) using the ansatz

$$\ln \varrho = \ln \varrho_0 + \lambda \,, \tag{12.194}$$

$$\mathbf{v} = \mathbf{0} + \mathbf{u} \tag{12.195}$$

[10] For a perfect gas in the adiabatic case (entropy s = const), the equation of state is

$$P = K\varrho^\gamma \,, \tag{12.187}$$

where $\gamma \equiv c_p/c_v$ is the adiabatic index, i. e. the ratio of specific heat at constant pressure, c_p, to the specific heat at constant volume, c_v, and K is a constant related to the entropy s. For this case we obtain the familiar relation

$$c_s^2 = \left(\frac{dP}{d\varrho}\right)_s = \gamma\frac{P}{\varrho} = \gamma\frac{\mathcal{R}}{\mu_{\rm mol}}T \tag{12.188}$$

where $\mathcal{R}/\mu_{\rm mol}$ is the specific gas constant, T the temperature, and $(\partial/\partial)_s$ denotes the partial derivative for constant entropy s.

A *polytropic equation of state* looks like the adiabatic one (12.187), but with the adiabatic index replaced by an exponent Γ that is treated as a free parameter.

and assuming that $\lambda \ll 1$, $|\mathbf{u}| \ll c_s$, we obtain the system

$$\frac{\partial \lambda}{\partial t} = -\frac{\partial u}{\partial x}\,, \tag{12.196}$$

$$\frac{\partial u}{\partial t} = -c_s^2 \frac{\partial \lambda}{\partial x}\,. \tag{12.197}$$

This system has the general solution

$$\lambda = f(x - c_s t) + g(x + c_s t) \tag{12.198}$$

$$u = c_s\, f(x - c_s t) - c_s\, g(x + c_s t) \tag{12.199}$$

where $f(\cdot)$, $g(\cdot)$ are arbitrary functions.

Nonlinear sound waves are described by our equations (12.191) and (12.192) and the nonlinearities give rise to new phenomena like steepening of wave profiles ans shocks.

12.6 Appendix

12.6.1 Lab exercises

FFT, wave numbers and IDL: IDL's `FFT()` function is pretty straight-forward to use, apart from the second argument which may be a bit confusing. E.g. `FFT(f)` and `FFT(f,-1)` return the forward transform of f, while `FFT(f,1)` will return the backward transform (which applies a normalization factor and is really the inverse of the forward transform). The ±1 in the second argument represent the sign of the exponent $\pm 2\pi i \mathbf{k} \cdot \mathbf{x}$.

The trivial but somewhat tedious part to be still done by the user is to construct the appropriate grid of $\mathbf{k}$ values. Let $L_x \equiv N_x \delta x$ be the extent of the x interval (i.e. the period length in x). Then k changes in steps of $\delta k = 2\pi/L_x$. IDL treats k (in some sense) as starting at $k_{\min} = 0$ and increasing monotonically until $k_{\max} = 2\pi/\delta x$. However, k and $k - 2\pi/\delta x$ are equivalent and normally we want k to go from $-k_{\mathrm{Ny}}$ to $k_{\mathrm{Ny}} = \pi/\delta x$. This can be achieved by

```
dk = 2*!pi/Lx
k1 = (findgen(Nx)-Nx/2)*dk
k  = shift(k1, -Nx/2)
```

When plotting spectral information over k,

```
spect = abs(fft(f,-1))
plot, k, spect
```

the plot will be OK, apart from a gap around $k = 0$ and a spurious line connecting the leftmost and the rightmost point. This is because k is piecewise linear in two intervals and is discontinuous at the interface (`plot, k` to see this).

This (`k` and `spect` in the example) is the representation that is most useful for everything but plotting spectra (or anything else in Fourier space). E.g. filtering is easily done using k, and it often helps that we know that $k = 0$ is at position `k[0]`.

For plotting spectra, we use k_1, which is a continuous version of k. We then *must* shift the spectral information:

```
spect1 = shift(spect,Nx/2)
plot, k1, spect1
```

Note: The k and k_1 used here are *physical* wave numbers, and are thus not identical to the ones (unfortunately) introduced in the lecture. Using physical wave numbers, the Laplace equation $\Delta f = g$ becomes

$$-k^2 \tilde{f} = \tilde{g}$$

in Fourier space – without the factor $4\pi^2$.

Question 32 *Fourier transform*

Use IDL's `fft()` function to calculate the (discrete) Fourier transforms of the functions

$$\begin{aligned} f_0(x) &= 1\,, \\ f_1(x) &= \sin 2\pi x\,, \\ f_2(x) &= \cos 2\pi x\,, \\ f_3(x) &= \frac{e^{-(x-x_0)^2/(2w^2)}}{\sqrt{2\pi w^2}}\,, \qquad x_0 = \frac{2}{3}, w = 0.03\,, \\ f_4(x) &= \cos 2\pi x\,, \end{aligned}$$

on the interval $0 \le x < 1$.

(a) Plot modulus $|\tilde{f_n}(k)|$, real part $\Re \tilde{f_n}(k)$, imaginary part $\Im \tilde{f_n}(k)$ and phase $\arg \tilde{f_n}(k)$.

(b) Make sure that the normalization and range of k is correct. For one of the functions (not $f_0 \ldots$), change the interval length and verify that k is still correct.

(c) Apply a low-pass filter to f_4 to screen out any wave number $k > 50$, transform back, plot $f_4(x)$ and overplot the filtered f_4.

Question 33 *Spectral method in 1-d*

Use the spectral method to solve the one-dimensional stationary heat conduction equation

$$\frac{d^2T}{dx^2} = -q(x)$$

with

$$q(x) = \frac{e^{-(x-x_0)^2/(2w^2)}}{\sqrt{2\pi w^2}}$$

on the interval $0 \le x < 1$. Assume periodic boundary conditions $T(x) = T(1)$.

(a) Use the forward transform `fft(f,-1)` on the right hand side $-q$.

(b) Construct an array `k_2` that is equal to $1/k^2$ for $|k| \neq 0$ and is 0 for $k = 0$.

(c) Multiply $-\tilde{q}$ by `k_2` and transform back using `fft(f,1)`.

(d) Plot the solution $T(x)$. Does this match your expectations?

(e) Plot the second derivative of $T(x)$,

```
plot, x, wdderiv2(x,Temp)
```

and compare to $-q(x)$. Can you explain?

Question 34 *Tridiagonal solver*

(a) Download the tridiagonal solver in F90, and the corresponding test program and Makefile from:
`http://www.capca.ucalgary.ca/top/teaching/phys499+535/Section11/triadiagonal`

Hint: You can modify files in these directories and check the modifications in, but I will not like that (unless you really fixed a bug or added something really clever)! It is thus best if you check out the files into a place where you will not edit or compile, and copy them over to your playground.

(b) Try to understand the Makefile, compile and run the test. Try to understand the test.

(c) Use the solver to solve the system

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ -C/2 & 1+C & -C/2 & 0 & 0 & 0 \\ 0 & -C/2 & 1+C & -C/2 & 0 & 0 \\ 0 & 0 & -C/2 & 1+C & -C/2 & 0 \\ 0 & 0 & 0 & -C/2 & 1+C & -C/2 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \vec{T} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

with $C = 0.3$

(d) Repeat this in IDL, using
`www.capca.ucalgary.ca/top/teaching/phys499+535/Section11/tridiagonal/solve_tridiag.pro`.

Question 35 *Simple heat conduction with high-order method*

The routines for this problem can be downloaded from:

`www.capca.ucalgary.ca/top/teaching/phys499+535/Section11/heatcond`

Solve the heat conduction equation

$$\frac{\partial T}{\partial t} = \chi \frac{\partial^2 T}{\partial x^2}$$

on the periodic interval $0 \le x < 2\pi$, where thermal diffusivity $\chi = 1$ is constant. The initial condition is

$$T(x,0) = \sum_{n=1}^{8} (-1)^{n-1} \frac{\sin nx}{n}$$

Plot the temperature profile and the heat flux density $F = -\partial T/\partial x$ for about 5 interesting times.

Find the Courant time step (border between stable and unstable).

Hint: You can use IDL's |deriv()| or our |xder()| to calculate the first derivative.

Question 36 *Advection*

The routines for this problem can be downloaded from:

`http://www.capca.ucalgary.ca/top/teaching/phys499+535/Section11/advect`

Solve the advection equation

$$\frac{\partial f}{\partial t} = -u\frac{\partial f}{\partial x}$$

on the periodic interval $0 \le x < 1$, where the advection velocity $u = 1$ is constant. The initial condition is

$$f(x, 0) = \tanh(7 \sin x) \; .$$

(a) Run for 10 time units and compare the final profile to the original one.

(b) Try to understand what the different files do. What is new compared to the systems of ordinary differential equations we have solved earlier?

(c) Find the Courant time step.

(d) Set numerical diffusivity |visc| to zero; what happens? How does the Euler time-stepping method fare in the absence of diffusivity?

(e) Use different spatial derivative schemes and compare. Does the spectral scheme really give the exact result (what happens if you decrease the time step?)?

(f) Switch to 6th-order hyperdiffusivity and find an acceptable value of |visc|. How does the energy decrease compare to 2nd-order normal diffusivity?

(g) If you do not like the deformation of the signal, you can increase the number of points.

Question 37 *Sound waves*

The routines for this problem can be downloaded from:

`http://www.capca.ucalgary.ca/top/teaching/phys499+535/Section11/sound`

(a) Solve the equations for sound waves in 1-d:

$$\begin{aligned} \partial_t \ln \varrho &= -v\,\partial_x \ln \varrho - \partial_x v \; , \\ \partial_t v &= -v\partial_x v - c_s^2 \partial_x \ln \varrho + \frac{4}{3}\nu \partial_x^2 v \end{aligned}$$

on the (non-periodic) interval $0 \le x < 1$ for a Gaussian profile as initial condition.

(b) Adapt the initial condition such that the bump moves only to the right (and stick with that for the following questions).

(c) Increase the amplitude to `ampl` = 0.3. What changes? For how long can you run the simulation?

(d) How much viscosity do you need to stabilize the run for an amplitude `ampl` = 2?

(e) What do the boundary conditions represent physically? Modify the right boundary condition to represent the open end of a pipe (e.g. in a wind instrument).

(f) Why can we use periodic derivative operators (they use IDL's cyclic `shift` function) for this non-periodic problem?

Question 38 *Schrödinger equation*

Solve the time-dependent Schrödinger equation

$$i\hbar\frac{\partial\psi}{\partial t} = H\psi \ ,$$

where

$$H = -\frac{\hbar^2}{2\,m}\frac{\partial^2}{\partial x^2} + U(x) \ .$$

for periodic boundary conditions on the interval $0 \le x < 2\pi$. Use units where $\hbar = 1$ and $m = 1$.

The initial condition is a wave packet

$$\psi = \frac{1}{(\pi\, w^2)^{1/4}} e^{-\frac{1}{2}\left(\frac{x-x_{\rm peak}}{w}\right)^2 + i\frac{p_0 x}{\hbar}} \ ,$$

where $x_{\rm peak} = \pi$, $w = 0.3$, $p_0 = 15$.

The potential is

$$U(x) = 150\,\frac{1 + \tanh\dfrac{\sin(x-4) - 0.95}{0.002}}{2}$$

Use at least 200 points. Monitor the total probability $\int |\psi|^2\, dx$ and increase the number of points if it varies noticeably.

Work *collectively* on this problem and check in your contributions under `common/schroedinger`.

(a) Plot $|\psi|$, $|\psi|^2$, $\Re\psi$, and $\Im\psi$

(b) Setting the potential to zero, how does the motion of the wave packet change when you change p_0? What is the physical meaning of p_0?

(c) What changes if you use a narrower wave packet?

(d) [Revert to the original width and potential] Vary the height of the potential barrier and explain the resulting changes.