F10ND2/F11ND2/F71NT – Numerical Methods for PDEs

1 Introduction

1.1 Definitions

The F10ND2 and F11ND2 modules cover the following topics, Fin MSc students (F71NT) take just topics 1 and 2:

Section 1. Introduction and basic concepts

Section 2. Parabolic PDE's (e.g. heat equation).

Section 3. Hyperbolic PDE's (e.g. wave equation).

Section 4. Elliptic PDE's (e.g. Poisson equation).

For all types of equations we consider the *finite difference method*. In addition, *finite element methods* are considered in Section 4.

Definition *A Partial Differential Equation (PDE)* is one or more equations connecting partial derivatives of one or more unknown functions (or *dependent variables*). In addition the equation(s) can contain known functions. All functions are functions of two or more *independent variables*.

Examples

$$\begin{array}{lll} (1) & u(x,t): & u_t = 3u_{xx} - 2u\,\cos(x) + 5\exp(-3t)\sin(2x) \\ (2) & u(x,y): & u_{xx} + (1+u_x)u_{yy} = 0 \end{array}$$

In (1), $u \equiv u(x,t)$ is the dependent variable, x and t are the independent variables, and we use the standard notation that

$$u_{xx} = \frac{\partial^2 u}{\partial x^2}, \qquad \text{etc.}$$

Exercise: show that one exact solution of (1) is $u(x,t) = 5\exp(-3t)\sin(x)$.

Definition The *Order/Degree* of a PDE is the highest order partial derivative occurring in the equation. In the examples above, both PDEs are 2nd order.

Definition A PDE is *linear* if the dependent variable only occurs in linear combinations multiplied by known functions of the independent variables. Otherwise it is *nonlinear*. In the examples above, (1) is linear and (2) is nonlinear.

The most general *linear, first order* PDE has the form

$$au_x + bu_t = c + d\,u,$$

here $u = u(x,t)$ is the dependent variable, x and t are the independent variables, and a, b, c, and d are constants or depend on x and t only. We will meet this type of equation in Section 3 since it is always of *hyperbolic* type. When we get to 2nd order equations we have more possibilities:

Classification It is useful to classify general 2nd order linear PDEs of 2 independent variables into *elliptic, parabolic*, or *hyperbolic* PDEs. If the PDE can be written as

$$au_{xx} + bu_{xy} + cu_{yy} + du_x + eu_y + fu + g = 0$$

where a, b, etc. are functions only of x and y, then we have the following

- if $b^2 - 4ac < 0$ then the equation is elliptic
- if $b^2 - 4ac = 0$ then the equation is parabolic
- if $b^2 - 4ac > 0$ then the equation is hyperbolic

(These definitions can be generalized to higher number of dimensions (independent variables) and other orders).

Examples

- elliptic:

  $$\boxed{\text{Poisson's equation: } \nabla^2 u = g(x,y)}$$

  $$\boxed{\text{Laplace's equation: } \nabla^2 u = 0.}$$

  where $u \equiv u(x,y), \quad \nabla^2 u = u_{xx} + u_{yy}.$

  Generally this type of equation is associated with equilibrium (steady-state) problems, e.g. steady flow of an incompressible fluid.

- parabolic:

  $$\boxed{\text{Heat or diffusion equation: } u_t = \kappa u_{xx}}$$

  where $\kappa > 0$. Here for example $u(x,t)$ is the temperature in a thin insulated rod, or the concentration of a certain chemical in a thin tube.

  $$\boxed{\text{Black-Scholes equation: } v_t + rsv_s + \frac{1}{2}\sigma^2 s^2 v_{ss} = rv}$$

  where $v(s,t)$ is the value of a share option, s is the share price, r is the interest rate, and σ is the share "volatility".

- hyperbolic:

  $$\boxed{\text{Wave equation: } u_{tt} = u_{xx}}$$

  Here $u(x,t)$ could be for example the water height in a narrow channel or the air pressure in a thin tube.

Exercise: Check that the examples given above of elliptic, parabolic, hyperbolic equations satisfy the definitions of such types.

Note that equations with variable coefficients can change type in different regions of the domain of the independent variables, for example

$$xu_{xx} + u_{tt} = 0$$

Exercise: Use the definitions above to determine what type this equation is, in which regions of the (x,t) plane.

1.2 Numerical methods for PDEs

We postpone until later a discussion of the extra information we need to solve the PDE, such as initial conditions and boundary conditions.

Although we can sometimes write down some *exact* solutions for some PDE's, in general this is not possible. In this case we need to have a way of finding *approximate* numerical solutions for a given equation. In this module we will generally stick to simple examples of PDEs which do have exact solutions, which has the great advantage that we can check whether our numerical methods and our computer programs are correct before we progress to more difficult cases.

We can group the main numerical techniques for PDEs as follows

1. **Finite Difference (FD) Methods.** We will use these throughout most of this course. In this type of method we forget about trying to find the values of the dependent variable, say u, for *all* values of the independent variables, say x and t. Instead we try to find approximate solutions of the problem at a *discrete* set of points in the (x, t) plane, normally a rectangular grid of points.

2. **Finite Element (FE) Methods.** We will touch on these briefly when looking at elliptic problems. In this method we first divide up the (x, t) plane into small *finite elements*, usually triangles or quadrilaterals, and then approximate solutions of the problem on each of these elements by simple linear or polynomial functions of the independent variables.

3. **Spectral Methods.** We do not treat these methods in this course due to lack of time, but they form an important class of techniques. Here we approximate the solutions of the problems by a truncated expansion in the eigenfunctions of some linear operator, for example a truncated Fourier Series.

The advantages and disadvantages of these three approaches can be briefly summarised as follows.

1 **FD Methods.** Simple to construct and to analyse. Need some modification in regions where the solution is changing rapidly. More difficult to apply when the solution region is not rectangular.

2 **FE Methods.** More difficult to set up than the FD method, but much better in irregular shaped domains. Not so easy to analyse at a simple level, but some rigorous results can be found with the aid of variational calculus. Widely used in engineering.

3 **Spectral Methods.** These give the highest accuracy when applied to problems in rectangular domains with smooth solutions. Not so useful in irregular domains or when the solutions has discontinuities such as shocks.

1.3 Introduction to the Finite Difference method

We will need to approximate partial derivatives such as

$$u_x \equiv \frac{\partial u(x,t)}{\partial x}, \quad u_t \equiv \frac{\partial u(x,t)}{\partial t}, \quad u_{xx} \equiv \frac{\partial^2 u(x,t)}{\partial x^2}.$$

The continuous variables x and t are *discretized*, so that we only consider $u(x,t)$ evaluated at the intersections (node points) of the grid lines parallel to the x and t axes. We use the

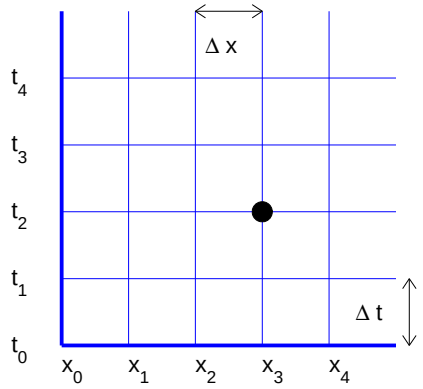


Figure 1: Finite difference mesh or grid

notation $x_j = x_0 + j\Delta x$, $t_n = t_0 + n\Delta t$, where (x_0, t_0) is the origin for the grid (often $(0,0)$), and Δx, Δt are the (constant) grid spacings in the x and t directions respectively. We will write $u_j^n = u(x_j, t_n)$ for the *exact* solution of the PDE at the grid point (x_j, t_n), and w_j^n for the *approximate* solution generated by the FD method at the same point.

The next step is to introduce some *Finite Difference operators*. Consider a function $f(x)$ of a single variable, then by definition

$$\frac{df}{dx} = \lim_{\Delta x \to 0} \left\{ \frac{f(x+\Delta x) - f(x)}{\Delta x} \right\}.$$

So for small Δx,

$$\frac{df}{dx} \approx \frac{f(x+\Delta x) - f(x)}{\Delta x}.$$

We now define the *Forward Difference* operator F_x such that $F_x f(x) = f(x+\Delta x) - f(x)$. Hence

$$\frac{df}{dx} \approx \frac{F_x f(x)}{\Delta x}.$$

The same reasoning goes through if f depends on the variable t also, so

$$\boxed{u_x = \frac{\partial u(x,t)}{\partial x} \approx \frac{F_x u(x,t)}{\Delta x} = \frac{u(x+\Delta x, t) - u(x,t)}{\Delta x}, \quad u_t = \frac{\partial u(x,t)}{\partial t} \approx \frac{F_t u(x,t)}{\Delta t},}$$

where we define F_t in an analogous way, $F_t f(x,t) = f(x, t+\Delta t) - f(x,t)$.

Exercise: Take $u(x,t) = \exp(-t)\sin(x)$. Find the *exact* value of $\partial u(x,t)/\partial x$ and $\partial u(x,t)/\partial t$ at the point $(x,t) = (0.5, 0.1)$ and calculate the approximate values given by the above using $\Delta x, \Delta t = 0.1, 0.01, \ldots$. Define the *error* as the difference between the exact and approximate results and consider what happened to the errors as $\Delta x, \Delta t \to 0$.

We can similarly define the corresponding *Backwards difference* operators $B_x u(x,t) = u(x,t) - u(x-\Delta x, t)$, $B_t u(x,t) = u(x,t) - u(x, t-\Delta t)$, so that

$$\boxed{\frac{\partial u(x,t)}{\partial x} \approx \frac{B_x u(x,t)}{\Delta x} = \frac{u(x,t) - u(x-\Delta x, t)}{\Delta x},}$$

$$\boxed{\frac{\partial u(x,t)}{\partial t} \approx \frac{B_t u(x,t)}{\Delta t} = \frac{u(x,t) - u(x, t-\Delta t)}{\Delta t}.}$$

(This is equivalent to replacing Δx by $-\Delta x$ in the definitions of F_x, etc.)
If we average the two approximations we have so far for u_x and u_t we introduce the *Central difference* operators $D_x u(x,t) = u(x+\Delta x, t) - u(x-\Delta x, t)$, $D_t u(x,t) = u(x, t+\Delta t) - u(x, t-\Delta t)$

$$\boxed{\frac{\partial u(x,t)}{\partial x} \approx \frac{D_x u(x,t)}{2\Delta x} = \frac{u(x+\Delta x, t) - u(x-\Delta x, t)}{2\Delta x},}$$

$$\boxed{\frac{\partial u(x,t)}{\partial t} \approx \frac{D_t u(x,t)}{2\Delta t} = \frac{u(x, t+\Delta t) - u(x, t-\Delta t)}{2\Delta t}.}$$

Going back to functions of one variable, these approximations have the simple geometric interpretation as various approximations to the tangent to the curve $y = f(x)$, see Fig. 2.

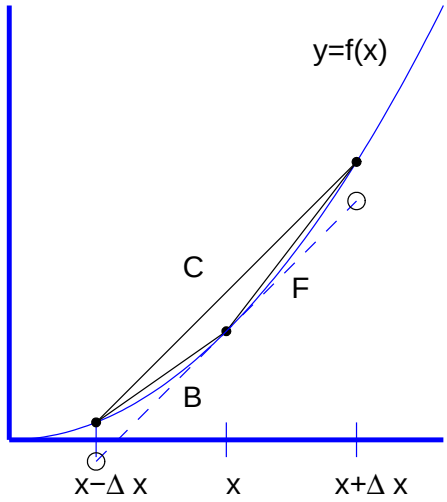


Figure 2: Approximations to o- - -o, the tangent to $y = f(x)$ at x, ·—· with D/F/B is Central/Forward/Backward difference approx.

Exercise: which of these three approximations do you expect to be the best approximation to the tangent? Check your guess by reworking the exercise above for $u(x,t) = \exp(-t)\sin(x)$ with Backward and Central Differences.

There is one further finite difference operator we need, a *second* central difference operator δ^2 to approximate *second* derivatives. (When there is danger of confusion we refer to D as the *first* central difference operator). Derivation of this one requires some FD manipulations. First

define another FD operator

$$\delta f(x) = f(x + \frac{1}{2}\Delta x) - f(x - \frac{1}{2}\Delta x)$$

So $\delta f(x)/\Delta x$ is like a central difference approximation to $df(x)/dx$ but using a *half* step size.

$$\text{so} \quad \frac{df(x)}{dx} \approx \frac{\delta f(x)}{\Delta x}, \text{ and } \frac{d^2 f(x)}{dx^2} = \frac{d}{dx}\left(\frac{df(x)}{dx}\right) \approx \frac{\delta}{\Delta x}\left(\frac{\delta f(x)}{\Delta x}\right) = \frac{\delta^2 f(x)}{\Delta x^2}.$$

Now

$$\begin{aligned}\delta^2 f(x) &= \delta\left(\delta f(x)\right) = \delta\left(f(x + \frac{1}{2}\Delta x) - f(x - \frac{1}{2}\Delta x)\right) \\ &= (f(x + \Delta x) - f(x)) - (f(x) - f(x - \Delta x)) = f(x + \Delta x) - 2f(x) + f(x - \Delta x)\end{aligned}$$

Combining these results we have

$$\frac{d^2 f(x)}{dx^2} \approx \frac{f(x + \Delta x) - 2f(x) + f(x - \Delta x)}{\Delta x^2}.$$

Where the numerator can be written as $\delta_x^2 f(x) = f(x+\Delta x) - 2f(x) + f(x-\Delta x)$. So for partial derivatives we have

$$\boxed{\frac{\partial^2 f(x,t)}{\partial x^2} \approx \frac{\delta_x^2 f(x,t)}{\Delta x^2} = \frac{f(x + \Delta x, t) - 2f(x,t) + f(x - \Delta x, t)}{\Delta x^2},}$$

$$\boxed{\frac{\partial^2 f(x,t)}{\partial t^2} \approx \frac{\delta_t^2 f(x,t)}{\Delta t^2} = \frac{f(x, t + \Delta t) - 2f(x,t) + f(x, t - \Delta t)}{\Delta t^2}.}$$

Exercise: With $u(x,t) = \exp(-t)\sin(x)$ again, find the *exact* value of $\partial^2 u(x,t)/\partial x^2$ and $\partial^2 u(x,t)/\partial t^2$ at the point $(x,t) = (0.5, 0.1)$ and calculate the approximate values given by the above 2nd central difference approximations using $\Delta x, \Delta t = 0.1, 0.01, \ldots$.

These four Finite Difference operators will be the key to all our work for the next few weeks. We will see a different justification for their use in the next section. Before leaving the topic we mention some other related results

We can establish various relations between the Finite Difference operators with a bit of algebra, for example $F_x B_x = B_x F_x = \delta_x^2$. For example

$$\begin{aligned}F_x B_x f(x) &= F_x(B_x f(x)) = F_x(f(x) - f(x - \Delta x)) = \\ &= (f(x + \Delta x) - f(x)) - (f(x) - f(x - \Delta x) = \delta_x^2 f(x)\end{aligned}$$

In general the FD operators commute with each other.

If we need to approximate a mixed derivative, for example $\partial^2 u(x,t)/\partial x \partial t$, we have seen that

$$\frac{F_x}{\Delta x} \approx \frac{\partial}{\partial x}, \frac{F_t}{\Delta t} \approx \frac{\partial}{\partial t},$$

so some possibilities are

$$\frac{\partial^2 u(x,t)}{\partial x \partial t} \approx \frac{F_x}{\Delta x}\frac{F_t}{\Delta t}u(x,t),\ \frac{B_x}{\Delta x}\frac{B_t}{\Delta t}u(x,t),\ \frac{B_x}{\Delta x}\frac{F_t}{\Delta t}u(x,t),\ \frac{D_x}{2\Delta x}\frac{D_t}{2\Delta t}u(x,t),\ \text{etc.}$$

Exercise: repeat the previous exercise for u_{xt}.

1.4 Taylor series and difference operators

We can use Taylor series expansions in one of the variables to see how well Finite Difference (FD) approximations work. Consider the approximation $(u(x+\Delta x,t)-u(x,t))/\Delta x$ to $\partial u/\partial x$.

$$\begin{aligned} F_x u(x_j,t_n) &= u(x_j+\Delta x,t_n)-u(x_j,t_n) \\ &= \left(u+\Delta x u_x+\frac{1}{2}\Delta x^2 u_{xx}+\frac{1}{3!}\Delta x^3 u_{xxx}+\dots\right)\Big|_{(x_j,t_n)} - u(x_j,t_n) \\ &= \left(\Delta x u_x+\frac{1}{2}\Delta x^2 u_{xx}+\dots\right)\Big|_{(x_j,t_n)} \\ &\approx \Delta x\, u_x(x_j,t_n) \text{ when } \Delta x \text{ is sufficiently small} \end{aligned}$$

So

$$\frac{F_x}{\Delta x}u(x_j,t_n) = u_x(x_j,t_n)+O(\Delta x)$$

where $O(\Delta x)$ means a quantity which $\to K\Delta x$ for some constant K, so $\to 0$ as $\Delta x\to 0$.
Similarly

$$\begin{aligned} D_x u &= u(x_j+\Delta x,t_n)-u(x_j-\Delta x,t_n) \\ &= \left(u+\Delta x\, u_x+\frac{1}{2}\Delta x^2\, u_{xx}+\frac{1}{3!}\Delta x^3\, u_{xxx}+\frac{1}{4!}\Delta x^4\, u_{xxxx}+O(\Delta x^5)\right)\Big|_{(x_j,t_n)} \\ &\quad -\left(u-\Delta x\, u_x+\frac{1}{2}\Delta x^2\, u_{xx}-\frac{1}{3!}\Delta x^3\, u_{xxx}+\frac{1}{4!}\Delta x^4\, u_{xxxx}+O(\Delta x^5)\right)\Big|_{(x_j,t_n)} \\ \Rightarrow \frac{D_x u}{2\Delta x} &= \left(u_x+\frac{\Delta x^2}{6}u_{xxx}+O(\Delta x^4)\right)\Big|_{(x_j,t_n)} \end{aligned}$$

So

$$\frac{D_x}{2\Delta x}u(x_j,t_n) = u_x(x_j,t_n)+O(\Delta x^2).$$

This is *more accurate* because $\Delta x^2\to 0$ faster than Δx as $\Delta x\to 0$.
In a similar way we can show that

$$\delta_x^2 u(x_j,t_n) = \Delta x^2 u_{xx}(x_j,t_n)+O(\Delta x^4)$$

so

$$\frac{\delta_x^2}{\Delta x^2}u(x_j,t_n) = u_{xx}(x_j,t_n)+O(\Delta x^2) \text{ when } \Delta x \text{ is small.}$$

By expanding first in one variable and then in the other(s), we can analyse approximations involving terms like $u(x_j+\Delta x,t_n+\Delta t)$.
Exercise: Show that

$$\frac{F_x F_t}{\Delta x\Delta t}u(x_j,t_n) = u_{xt}(x_j,t_n)+O(\Delta x,\Delta t).$$

and

$$\frac{D_x D_t}{4\Delta x\Delta t}u(x_j,t_n) = u_{xt}(x_j,t_n)+O(\Delta x^2,\Delta t^2).$$

so that again Central differences are better.
We now have enough background to start looking at PDEs, starting with parabolic problems.

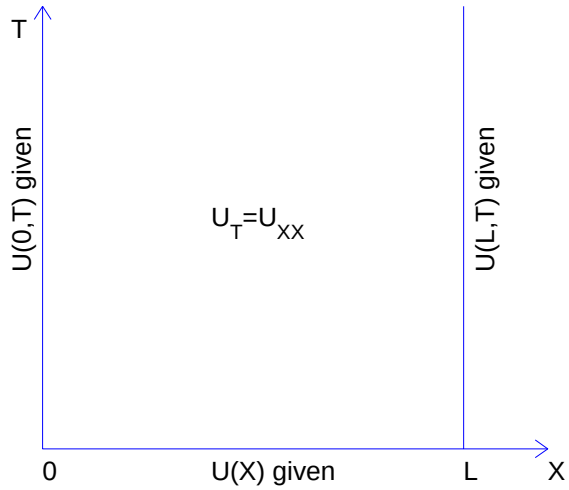
2 Parabolic PDE's

2.1 Introduction

The most practical approach is to study a concrete example. Hence we start by looking at the **heat** or **diffusion** equation

$$\frac{\partial U}{\partial T} = K\frac{\partial^2 U}{\partial X^2}, \tag{2.1}$$

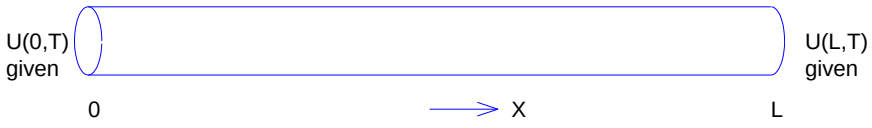
where $U = U(X,T)$ is measured in some units (say °C) and the physical time and space variables $T > 0$, $X \in (0,L)$ are also measured in appropriate units. K is a constant which depends on the material, called the *Thermal Conductivity*. We specify Dirichlet *boundary conditions* (BCs) at $X = 0, L$, i.e. $U(0,T)$ and $U(L,T)$ given for all $T > 0$, and *initial conditions* (ICs) $U(X,0)$ given for $X \in (0,L)$.



Need to find the solution in the open box $T > 0, X \in (0,L)$

The heat equation models many things, for example

- the temperature U in a long thin insulated bar with fixed temperatures at each end



- The diffusion of a chemical in a solid or stationary fluid (in this case U is the concentration of the chemical).

Since the dimensions and units used in each application can be different, it is useful, before we do anything else, to transform to a "standard problem" in dimensionless form. First define a new variable $x = X/L$ so that the equation becomes

$$\frac{\partial U}{\partial T} = \frac{K}{L^2}\frac{\partial^2 U}{\partial x^2}, \text{ for } x \in (0,1).$$

Now define a new variable $t = KT/L^2$ so that the equation becomes

$$\frac{\partial U}{\partial t} = \frac{\partial^2 U}{\partial x^2}.$$

Finally define $u(x,t) = U(x,t)/U_0$, where U_0 is some typical constant in the problem (for example the maximum temperature of the bar), to get finally

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, \tag{2.2}$$

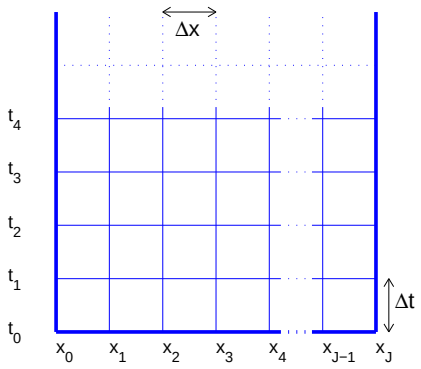
where now $u \in (0,1)$. From now on we will usually work with the equation in the form (2.2) with $x \in (0,1)$. We can always return to the physical version (2.1) by reversing the transformations above. In these dimensionless variable we will define the initial conditions and boundary conditions as

$$u(x,0) = F(x), \quad u(0,t) = \alpha(t), \quad u(1,t) = \beta(t),$$

for some given functions $F(x), \alpha(t), \beta(t)$.

2.2 A simple Finite Difference method for the heat equation

As described in Section 1.3, we divide up the (x,t) plane using a rectangular grid of mesh points separated by $\Delta x, \Delta t$ respectively.



We take $\Delta x = 1/J$ so that $x_0 = 0, x_J = 1, x_j = x_0 + j\Delta x$. We wish to approximate the equation (2.2) at the mesh points (nodes) so that

$$u(x_j, t_n) = u_j^n \approx w_j^n$$

where w_j^n is our numerical *approximation* to the *exact* solution $u(x_j,t_n) = u_j^n$ at (x_j,t_n).
We consider in this section the simplest approximation to the PDE, using forward differences for u_t and central differences for u_{xx}.

$$u_t(x_j,t_n) \approx \frac{F_t}{\Delta t} u_j^n, \quad u_{xx}(x_j,t_n) \approx \frac{\delta_x^2}{\Delta x^2} u_j^n$$

So we have from (2.2) that

$$\frac{F_t}{\Delta t}u_j^n \approx \frac{\delta_x^2}{\Delta x^2}u_j^n,$$

at all interior points $(x_j, t_n), 0 < j < J, n > 0$. We *define* w_j^n to satisfy this equation with equality.

$$\frac{F_t}{\Delta t}w_j^n = \frac{\delta_x^2}{\Delta x^2}w_j^n$$

so

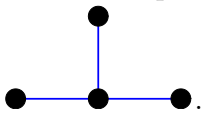
$$\frac{w_j^{n+1} - w_j^n}{\Delta t} = \frac{w_{j-1}^n - 2w_j^n + w_{j+1}^n}{\Delta x^2}$$

Multiplying through by Δt, defining $r = \Delta t/\Delta x^2$, and re-arranging, we get

$$\text{FTCS scheme: } w_j^{n+1} = rw_{j-1}^n + (1-2r)w_j^n + rw_{j+1}^n \tag{2.3}$$

Here FTCS stands for Forward Time, Central Space difference approximation. We also need the following IC and BCs:

$$\text{IC: } w_j^0 = F(x_j), \quad j = 0, \ldots, J, \qquad \text{BC: } \left.\begin{array}{l} w_0^n = \alpha(t_n) \\ w_J^n = \beta(t_n) \end{array}\right\} \quad n > 0$$

With these conditions we can use (2.3) to find the other w_j^n for $0 < j < J$, $n > 0$ as described below. The key to the process is to understand that the FTCS scheme (2.3) gives an equation connecting one value of w at the time level $n+1$ with three values at the time level n. Graphically we can represent this as a "computational molecule" which looks like this:  .

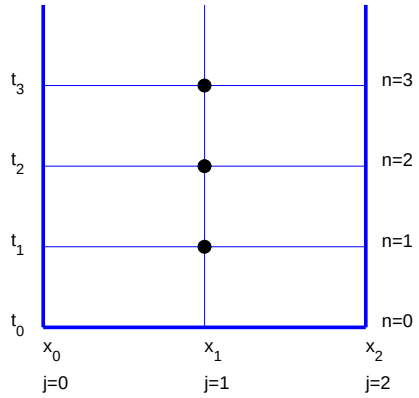
If we know all the values of w_j^n for $0 \le j \le J$ at a time level n, we can use (2.3) to calculate w_j^{n+1} for $j = 1, 2, \ldots, J-1$ in turn. The scheme is *explicit* - we don't have to solve any coupled algebraic equations to get from t_n to $t_{n+1} = t_n + \Delta t$. The method will become clear once we have worked through some examples, but we summarise the main steps here

1. Choose J and r then calculate $\Delta x = 1/J, \Delta t = r\Delta x^2$.
2. Calculate w_j^0 and set $n = 0$.
3. Use (2.3) to get $w_j^{n+1}, j = 1 \ldots J-1$, and get w_0^{n+1}, w_J^{n+1} from the boundary conditions.
4. Set $n \to n+1$ and repeat from Step 3 until t_n is "big enough".

Example

Use the FTCS scheme (2.3) to find the approximate solution to (2.2) with BCs $u(0,t) = u(1,t) = 0$, IC $u(x,0) = \sin(\pi x)$, when $r = 0.4$ and $J = 2, 4$ respectively. Calculate at least two time steps. Using the fact that an exact solution of this problem is $u(x,t) = \exp(-t\pi^2)\sin(\pi x)$, calculate the maximum error at each time step.

J=2



We have $\Delta x = 1/2 = 0.5$, $\Delta t = r\Delta x^2 = 0.1$, and we need to find the solution at the points $\bullet$ in the figure above. Along the bottom row $t = t_0 = 0$ we have $x_j = 0, 0.5, 1$, and from the ICs

$$w_0^0 = \sin(\pi x_0) = 0, \quad w_1^0 = \sin(\pi x_1) = 1, \quad w_2^0 = \sin(\pi x_2) = 0,$$

Now take $n = 0$ in (2.3) with $j = 1$. We have

$$w_1^1 = rw_0^0 + (1-2r)w_1^0 + rw_2^0 = 0 + 0.2 \times 1 + 0 = 0.2$$

The two boundary conditions give

$$w_0^1 = 0, \quad w_2^1 = 0.$$

Now take $n = 1$ in (2.3) with $j = 1$. We have

$$w_1^2 = rw_0^1 + (1-2r)w_1^1 + rw_2^1 = 0 + 0.2 \times 0.2 + 0 = 0.04$$

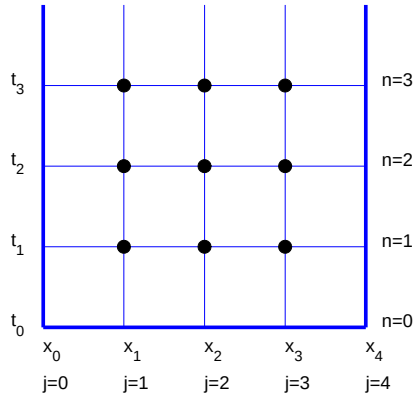
The two boundary conditions give

$$w_0^2 = 0, \quad w_2^2 = 0.$$

So for $t = 0.1, 0.2$ we have the numerical values $w_1^1 = 0.2, w_1^2 = 0.04$. The corresponding exact values are $\exp(-\pi^2 t_n)\sin(\pi x_1) = \exp(-0.1\pi^2), \exp(-0.2\pi^2) = 0.37, 0.14$ to 2D. So the accuracy is not too great with this large value of Δx.

Exercise: carry out more time steps in this calculation. Can you guess at a general solution for w_1^n?

J=4



We have $\Delta x = 1/4 = 0.25$, $\Delta t = r\Delta x^2 = 0.025$, and we need to find the solution at the points • in the figure above. Along the bottom row $t = t_0 = 0$ we have $x_j = 0, 0.25, 0.5, 0.75, 1$, and from the ICs

$$w_0^0 = \sin(\pi x_0) = 0, \quad w_1^0 = \sin(\pi x_1) = 1/\sqrt{2}, \quad w_2^0 = \sin(\pi x_2) = 1,$$
$$w_3^0 = \sin(\pi x_3) = 1/\sqrt{2}, \quad w_4^0 = \sin(\pi x_4) = 0.$$

Now take $n = 0$ in (2.3) with $j = 1$. We have

$$w_1^1 = rw_0^0 + (1-2r)w_1^0 + rw_2^0 = 0 + 0.2 \times \frac{1}{\sqrt{2}} + 0.4 \times 1 = 0.5414$$
$$w_2^1 = rw_1^0 + (1-2r)w_2^0 + rw_3^0 = 0.4 \times \frac{1}{\sqrt{2}} + 0.2 \times 1 + 0.4 \times \frac{1}{\sqrt{2}} = 0.7657$$
$$w_3^1 = rw_2^0 + (1-2r)w_3^0 + rw_4^0 = 0.4 \times 1 + 0.2 \times \frac{1}{\sqrt{2}} + 0 = 0.5414$$

The two boundary conditions give

$$w_0^1 = 0, \quad w_4^1 = 0.$$

Now take $n = 1$ in (2.3) with $j = 1$. We have

$$w_1^2 = rw_0^1 + (1-2r)w_1^1 + rw_2^1 = 0 + 0.2 \times 0.5414 + 0.4 \times 0.7657 = 0.4146$$
$$w_2^2 = rw_1^1 + (1-2r)w_2^1 + rw_3^1 = 0.4 \times 0.5414 + 0.2 \times 0.7657 + 0.4 \times 0.5414 = 0.5863$$
$$w_3^2 = rw_2^1 + (1-2r)w_3^1 + rw_4^1 = 0.4 \times 0.7657 + 0.2 \times 0.5414 + 0 = 0.4146$$

The two boundary conditions give

$$w_0^2 = 0, \quad w_4^2 = 0.$$

So for $t = 0.025, 0.05$ we have following numerical values for w_j^1, w_j^2 to 4D

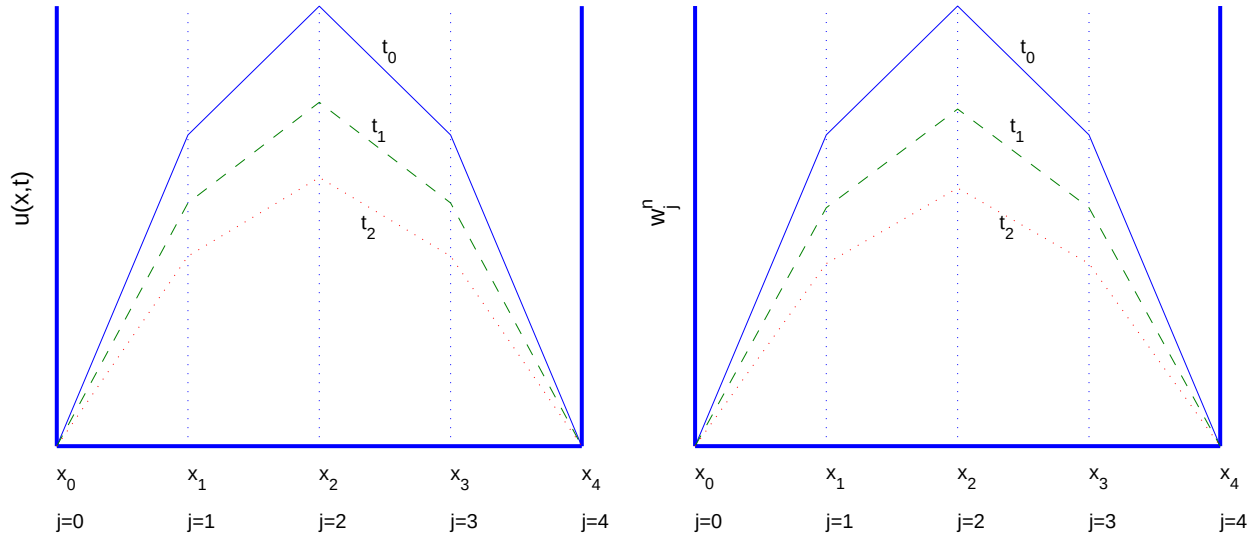
$$w_j^1 = [0, 0.5414, 0.7657, 0.5414, 0]$$
$$w_j^2 = [0, 0.4146, 0.5863, 0.4146, 0]$$

The corresponding exact values are $u_j^1, u_j^2 = \exp(-\pi^2 t_n)\sin(\pi x_j) = \exp(-0.025\pi^2)\sin(\pi x_j)$, $\exp(-0.05\pi^2)\sin(\pi x_j)$, given below to 4D

$$u_j^1 = [0, 0.5525, 0.7813, 0.5525, 0]$$
$$u_j^2 = [0, 0.4317, 0.6105, 0.4317, 0]$$

So the accuracy is better with this smaller value of Δx (but note we are now working at much smaller values of t_n also).

We plot below the exact solution in the $J = 4$ case and the corresponding numerical approximation. It can be seen that qualitatively at least the agreement is good.



Exercise: carry out more time steps in this calculation, and repeat the above calculations for $r = 1.0$.

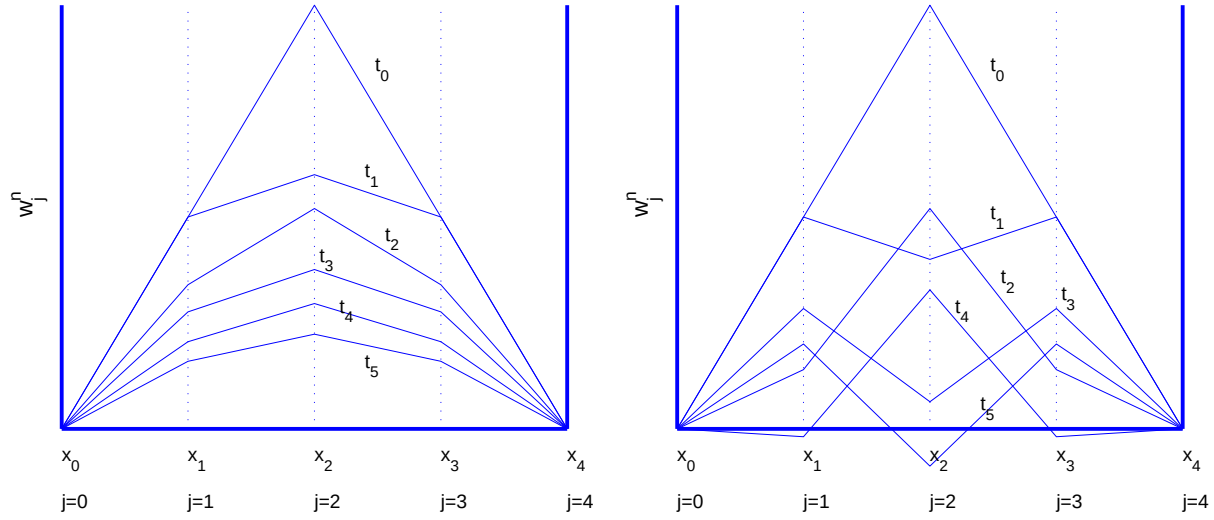
Another example we can look at, with $J = 4$, is the "triangular" initial conditions

$$F(x) = \begin{cases} 2x, & x \leq 0.5 \\ 2(1-x), & x > 0.5 \end{cases}$$

(This choice of an IC with a discontinuous derivative, rather than a $\sin(\pi x)$, results in problems becoming apparent at smaller times. Similar things happen for $\sin(\pi x)$ but take longer to manifest themselves). With $r = 0.4$ (left) and $r = 0.6$ (right) we get the two graphs shown below. Note that when $r = 0.4$ the numerical solution decreases smoothly towards zero, in line with what we would expect in the physical model. But when $r = 0.6$ the solution develops spatial oscillations which are increasing in amplitude. If run for further times (try it!), these oscillations become unbounded, i.e. $w_j^n \to \infty$ as $n \to \infty$, in contrast to the exact solution which can be shown to satisfy $\to 0$ as $t \to \infty$. We would regard this as bad. Even at small times we find the temperature is becoming negative at some points, physically contradicting the laws of thermodynamics!

Technically if this sort of bad behaviour occurs we say the scheme *unstable*. We study one way of analysing this behaviour over the next few lectures.

Exercise: repeat the above calculations for $r = 1.0$ and $w_0 = [0, 0.5, 1, 0.5, 0]$. What goes wrong? Why? We will answer this question in a few lectures' time. First we consider something called the Local Truncation Error (LTE) of the scheme (2.3).



2.3 Local Truncation Error (LTE) analysis

Given an approximation for the PDE (2.2), how do we know it is any good? The first step in analysing any scheme is to examine the Local Truncation Error (LTE). To do this we first write the PDE in the operator form

$$Lu = 0, \quad \text{where } L = \frac{\partial}{\partial t} - \frac{\partial^2}{\partial x^2}$$

We then replace the partial derivatives by the FD approximations we are using, to get a FD approximation to L, which we will call L_Δ. For the FTCS scheme we have

$$L_\Delta = \frac{F_t}{\Delta t} - \frac{\delta_x^2}{\Delta x^2}$$

The numerical solution w_j^n will satisfy the equation

$$L_\Delta w_j^n = 0 = \frac{w_j^{n+1} - w_j^n}{\Delta t} - \frac{w_{j-1}^n - 2w_j^n + w_{j+1}^n}{\Delta x^2} \tag{2.4}$$

Note that we do **not** multiply this equation by Δt at this stage, in contrast to our derivation of (2.3).

Definition: Local Truncation Error

Let L be a differential operator and let L_Δ be the corresponding difference operator approximating L. The LTE of the approximation is given by the leading terms of the Taylor expansion of $L_\Delta u$, where u satisfies $Lu = 0$. In other words, the LTE is found by substituting the *exact solution* of the PDE into the finite difference equation, then Taylor expanding and cancelling as many terms as possible.

Warning: do not multiply or divide L_Δ by any factors such as Δt or Δx when working out the LTE

Example: Calculate the LTE of the FTCS scheme (2.3).
In (2.4), we replace w_j^n by $u(x_j, t_n)$, etc., and expand all the terms in a Taylor series around the point (x_j, t_n).

$$\begin{aligned}
\text{LTE} &= L_\Delta u(x_j, t_n) \\
&= \frac{u(x_j, t_n + \Delta t) - u(x_j, t_n)}{\Delta t} - \frac{u(x_j - \Delta x, t_n) - 2u(x_j, t_n) + u(x_j + \Delta x, t_n)}{\Delta x^2} \\
&= \frac{1}{\Delta t}\left(u + \Delta t u_t + \frac{1}{2}\Delta t^2 u_{tt} + \cdots - u\right)\Big|_{x_j,t_k} - \\
&\quad - \frac{1}{\Delta x^2}\left(u - \Delta x u_x + \frac{1}{2}\Delta x^2 u_{xx} - \frac{1}{3!}\Delta x^3 u_{xxx} + \frac{1}{4!}\Delta x^4 u_{xxxx} - 2u + \right. \\
&\qquad \left. + u + \Delta x u_x + \frac{1}{2}\Delta x^2 u_{xx} + \frac{1}{3!}\Delta x^3 u_{xxx} + \frac{1}{4!}\Delta x^4 u_{xxxx}\right)\Big|_{x_j,t_k} \\
&= u_t + \frac{1}{2}\Delta t u_{tt} - u_{xx} - \frac{1}{12}\Delta x^2 u_{xxxx} + \ldots
\end{aligned}$$

where all the u and derivatives are evaluated at (x_j, t_n). Since u satisfies the heat equation, $u_t - u_{xx} = 0$, we are left with

$$\text{LTE} = \frac{\Delta t}{2} u_{tt} - \frac{\Delta x^2}{12} u_{xxxx} + O(\Delta t^2, \Delta x^4)$$

By using the fact that

$$u_{tt} = \frac{\partial}{\partial t} u_t = \frac{\partial}{\partial t} u_{xx} = \frac{\partial^2}{\partial x^2} u_t = \frac{\partial^2}{\partial x^2} u_{xx} = u_{xxxx}$$

we can also write the LTE as

$$\text{LTE} = \left(\frac{\Delta t}{2} - \frac{\Delta x^2}{12}\right) u_{xxxx} + O(\Delta t^2, \Delta x^4) = \frac{\Delta x^2}{2}\left(r - \frac{1}{6}\right) u_{xxxx} + O(\Delta t^2, \Delta x^4)$$

where $r = \Delta t / \Delta x^2$. From this form we see that the $O(\Delta x^2)$ terms vanish if $r = 1/6$, and the LTE becomes $O(\Delta t^2, \Delta x^4)$.
The first term on the right of the LTE for the FTCS scheme is referred to as the *leading term* of the local truncation error (LTE): for this scheme it is *first order accurate in time* and *2nd order in space.*
Definition: consistency/order If the LTE of a scheme $\to 0$ as $\Delta x, \Delta t \to 0$, the scheme is said to be *consistent.* This is the minimum requirement for any numerical scheme. Furthermore, if the LTE is of order $O(\Delta x^p, \Delta t^q)$, the scheme is said to be of order p in space and q in time. If r is fixed so we can write the LTE as $O(\Delta x^p)$ as in the above expression for the FTCS scheme, we say more generally that the scheme is pth order. So the FTCS scheme for the heat equation is 2nd order if $r \neq \frac{1}{6}$, and 4th order if $r = \frac{1}{6}$.
Exercise: Check that all the 3rd order terms in the LTE for the FTCS scheme vanish.

2.4 Matrix version of FTCS scheme

We have seen that if $\Delta x, \Delta t$ are small, then the LTE for the FTCS scheme $\to 0$ as $\Delta x, \Delta t \to 0$. This is saying that in some sense the error is small after one time step, starting with the exact solution. However this is not enough to guarantee that the scheme gives good results after a large number of time steps. The examples we looked at in previous lectures suggest that problems occur in the FTCS scheme if $r > 0.5$. If we run a program which shows oscillations becoming unbounded in magnitude, i.e. $w_j^n \to \infty$ as $n \to \infty$, in contrast to the exact solution which can be shown to satisfy $\to 0$ as $t \to \infty$, we regard this as bad. Even at small times we find the temperature is becoming negative at some points, physically contradicting the laws of thermodynamics!

Technically if this sort of bad behaviour occurs we say the scheme *unstable*. We study one way of analysing this behaviour in this section. To do this we write the FTCS scheme in matrix form.

Set

$$\mathbf{w}^n = \begin{pmatrix} w_1^n \\ w_2^n \\ \vdots \\ w_{J-1}^n \end{pmatrix},$$

the vector of values of the numerical solution at the internal spatial grid points at time level t_n. We know from the initial conditions that

$$\mathbf{w}^0 = \mathbf{u}^0 = \begin{pmatrix} F(x_1) \\ F(x_2) \\ \vdots \\ F(x_{J-1}) \end{pmatrix},$$

so $\mathbf{w}^0$ is known, also $w_0^n = \alpha(t_n), w_J^n = \beta(t_n)$. The scheme (2.3) is

$$w_j^{n+1} = rw_{j-1}^n + (1-2r)w_j^n + rw_{j+1}^n, \quad j = 1, \ldots, J-1.$$

This can be condensed to

$$\mathbf{w}^{n+1} = S\mathbf{w}^n + \mathbf{b}^n, \tag{2.5}$$

where

$$S = \begin{pmatrix} 1-2r & r & 0 & \ldots & & 0 \\ r & 1-2r & r & 0 & \ddots & \ddots \\ 0 & r & 1-2r & r & 0 & \ddots \\ \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ \ddots & \ddots & 0 & r & 1-2r & r \\ 0 & \ddots & \ddots & 0 & r & 1-2r \end{pmatrix}, \qquad \mathbf{b}^n = \begin{pmatrix} r\alpha(t_n) \\ 0 \\ \vdots \\ \vdots \\ 0 \\ r\beta(t_n) \end{pmatrix}.$$

Note that the matrix version of the FTCS scheme (2.5) includes both the PDE and the BCs. However we have chosen to calculate and store in $\mathbf{w}$ only the internal points, in contrast to the

matlab program presented earlier. Hence j just goes from 1 to $J-1$ instead of from 0 to J (maths, includes boundary points) or from 1 to `J+1` (Matlab, includes boundary points). This form is useful for analysis but not so useful for a computer program since we would have to add the boundary values before plotting the results.

Warning: Using the scheme in the form (2.5) for actual calculation will only be efficient in Matlab if you declare the matrix S to be *sparse*. A sparse matrix is one which has most elements zero. This way Matlab will not spend any time multiplying together terms which will give zero.

From (2.5) we have

$$\begin{aligned}\mathbf{w}^{n+1} &= S\mathbf{w}^n + \mathbf{b}^n \\ &= S(S\mathbf{w}^{n-1} + \mathbf{b}^{n-1}) + \mathbf{b}^n \\ &= S^2\mathbf{w}^{n-1} + S\mathbf{b}^{n-1} + \mathbf{b}^n.\end{aligned}$$

we repeat back to $\mathbf{w}^0$ or use induction to get

$$\mathbf{w}^{n+1} = S^{n+1}\mathbf{w}^0 + S^n\mathbf{b}^0 + S^{n-1}\mathbf{b}^1 + \cdots + S\mathbf{b}^{n-1} + \mathbf{b}^n, \tag{2.6}$$

where

$$S^n = \underbrace{S \times S \times S \times \cdots \times S}_{n \text{ times}}.$$

We want to find out what happens to the solution for large n. We first simplify (2.6) by setting $\alpha(t) = \beta(t) = 0$ (as in our numerical examples earlier) – this makes the calculations below easier, but a similar result can be found in the more general case. Then we have $\mathbf{b}^n = 0$ for all n so (2.6) becomes

$$\mathbf{w}^{n+1} = S^{n+1}\mathbf{w}^0.$$

Note that the superscript on the matrix S means S raised to the $n+1$th power, but superscript on the $\mathbf{w}$ refers to the time level. What does the solution $\mathbf{w}^{n+1}$ look like? We need to know something about the eigenvalues of S to answer this.

2.4.1 Some facts about Eigenvalues

(i) If λ is an eigenvalue of S with $|\lambda| > 1$ and $\mathbf{e}$ is the corresponding eigenvector ($S\mathbf{e} = \lambda\mathbf{e}$), then

$$||S^n\mathbf{e}|| = ||\lambda^n\mathbf{e}|| \to \infty,$$

where $||\cdot||$ is the *Euclidean norm* $||x|| = (|x_1|^2 + |x_2|^2 + \cdots + |x_n|^2)^{\frac{1}{2}}$.

(ii) If all the eigenvalues of S satisfy $\lambda \leq 1$ in modulus, then $||S^n\mathbf{z}|| \to 0$, ($\lambda < 1$) or remain bounded (some $\lambda = 1$) as $n \to \infty$ for all vectors $\mathbf{z}$. Note: for this we require S to be symmetric.

In summary, the solution to (2.5) (and hence to (2.3)) is only well-behaved when all the eigenvalues of S are ≤ 1 in modulus.

(iii) Because S is tri-diagonal with constant coefficients, it is possible to derive a nice formula for its eigenvalues

$$\begin{aligned}\lambda_s &= 1 - 4r\sin^2\left(\frac{s\pi}{2J}\right), \quad s = 1, \ldots, J-1, \\ &= 1 - 4r\sin^2\left(\frac{s\pi}{2}\Delta x\right) \quad \text{since } J\Delta x = 1.\end{aligned}$$

For stability we require

$$-1 \leq 1 - 4r\sin^2\left(\frac{s\pi}{2J}\right) \leq 1 \quad \text{as } \Delta x \to 0.$$

The right-hand inequality is always true, so we just need to satisfy the left-hand one. We have

$$\begin{aligned} & 1 - 4r\sin^2\left(\frac{s\pi}{2J}\right) \geq -1 \\ & 1 - 4r\sin^2\left(\frac{(J-1)\pi}{2J}\right) \geq -1 \quad \text{(worst case)} \\ \text{i.e.} \quad & 1 - 4r\sin^2\left(\frac{\pi}{2}(1-\Delta x)\right) \quad \text{as } \Delta x \to 0 \\ \text{so} \quad & 2r\sin^2\left(\frac{\pi}{2}(1-\Delta x)\right) \leq 1 \\ \text{finally} \quad & r \leq \frac{1}{2\sin^2\left(\frac{\pi}{2}(1-\Delta x)\right)} \end{aligned}$$

Hence we need

$$r \leq \frac{1}{2},$$

since Δx can be arbitrary small. So we expect $r \leq \frac{1}{2}$ to give a "good" solution and $r > \frac{1}{2}$ to give a poor solution – which is what we find in practice.

Remarks

(i) This way of analysing the stability of a scheme is not easily generalized since it involves finding the eigenvalues of the corresponding S-matrix.

(ii) The condition $|\lambda_s| \leq 1$ only guarantees stability because S is symmetric (true in general for parabolic equations but not for hyperbolics).

We now look at a different way of determining whether an approximation scheme is stable or not - the *Fourier* method or *von Neumann* method.

2.5 Fourier or von Neumann method

We saw in the simple $J = 2$ case that we could find an exact solution of the numerical scheme in the form

$$w_1^n = (1 - 2r)^n.$$

Exercise: for what values of r does the above solution become unstable?

We would like to find a more general solution of the numerical scheme in the $J > 2$ case. We turn back to the PDE for guidance. We have already seen that the function

$$u(x,t) = a\exp(-\pi^2 t)\sin(\pi x)$$

satisfies the heat equation with zero boundary conditions ($\alpha = \beta = 0$). More generally, a solution satisfying the same PDE and BCs is

$$u(x,t) = a_k \exp(-\pi^2 k^2 t)\sin(k\pi x), \quad k = 1, 2, \ldots, J$$

for any choice of constant a_k.
Exercise: Check this!
In fact the most general solution we can write down for the heat equation with zero boundary conditions is a linear sum of such terms

$$u(x,t) = \sum_{k=1}^{J} a_k \exp(-\pi^2 k^2 t)\sin(k\pi x),$$

which you will recognise as the Fourier series solution for the heat equation.
It is easier to proceed if we write $\sin(k\pi x) = \left(\exp(ik\pi x) - \exp(-ik\pi x)\right)/2i$, where $i^2 = -1$, and write the Fourier series in *complex form*

$$u(x,t) = \sum_{k=-J}^{J} c_k \exp(-\pi^2 k^2 t)\exp(i\pi k x).$$

Consider now a single term in this series at $(x_j, t_n) = (j\Delta x, n\Delta t)$.

$$\exp(-\pi^2 k^2 t_n)\exp(i\pi k x_j) = \exp(-\pi^2 k^2 n\Delta t)\exp(i\pi k j\Delta x).$$

We can write this as

$$\exp(-\pi^2 k^2 \Delta t)^n \exp(i\pi k j\Delta x) = \xi^n \exp(i\omega j)$$

where $\xi = \exp(-\pi^2 k^2 \Delta t)$ and $\omega = \pi k \Delta x$, where k can take any value between $-J$ and J.
The first part of this looks like the solution of the numerical problem with $J = 2$ we discussed earlier. The essential step in the von Neumann method is to assume a typical component of the *numerical* solution has the form

$$\boxed{w_j^n = \xi^n \exp(i\omega j)} \tag{2.7}$$

for some constants ξ and ω. The term ξ is called the *amplification factor*.
Lets try this in the FTCS scheme. If we insert this ansatz into (2.3) we get

$$\xi^{n+1}\exp i\omega j = r\xi^n \exp(i\omega(j-1)) + (1-2r)\xi^n \exp(i\omega j) + r\xi^n \exp(i\omega(j+1))$$

cancelling through by $\xi^n \exp(i\omega j)$ we get

$$\begin{aligned}\xi &= r\exp(-i\omega) + (1-2r) + r\exp(i\omega)\\ &= 1 - 2r + 2r\cos(\omega) = 1 - 2r(1-\cos(\omega))\end{aligned}$$

Now $1-\cos(\omega) = 1-(1-2\sin^2\left(\frac{1}{2}\omega\right)) = 2\sin^2\left(\frac{1}{2}\omega\right)$, so we have finally

$$\boxed{\xi = 1 - 4r\sin^2\left(\frac{1}{2}\omega\right)} \tag{2.8}$$

If ξ satisfies (2.8) then $w_j^n = c_k\xi^n\exp(i\omega j)$ is a solution to the FTCS scheme (2.3). What does this solution look like?

(i) If $|\xi| > 1$ then $|w_j^n| = |c_k\xi^n| \to \infty$ as $n \to \infty$, so the solution blows up and the scheme is said to be *unstable*.

(ii) If $|\xi| \leq 1$ then the solution doesn't blow up and the scheme is said to be *von Neumann stable*.

Let's see what this means for the FTCS scheme. The requirement that $|\xi| \leq 1$ for stability becomes

$$-1 \leq 1 - 4r\sin^2\left(\frac{1}{2}\omega\right) \leq 1.$$

This is very similar to what we got from the matrix analysis - the r.h. inequality is always true and the left-hand one leads to

$$\boxed{r \leq \frac{1}{2\sin^2\left(\frac{1}{2}\omega\right)} = \frac{1}{2}}$$

the last step following from the fact that the worst case for the inequality is $\omega = \pi$. This is the same result that we got from the matrix analysis, but the von Neumann method is much easier to apply.

Note that the result $r \leq \frac{1}{2}$ corresponds to $\Delta t \leq \frac{1}{2}\Delta x^2$, which fits in with the assumption $\Delta t = O(\Delta x^2)$ which we made when looking at the LTE.

2.5.1 Summary - von Neumann stability analysis

- Substitute a solution of the form

  $$w_j^n = \xi^n\exp(i\omega j)$$

  into the difference scheme, simplify by cancelling common factors, and solve for ξ in terms of ω, r, etc.

- Determine if the amplification factor ξ has modulus ≤ 1 for all values of $|\omega| \leq \pi$. If this is so for all values of r we have *unconditional stability*.

- If $|\xi| \leq 1$ for some range of r, we say the scheme is *von Neumann stable* for r in the stated range, otherwise the scheme is *unstable*.

Notes:

- von Neumann stability is always necessary but may not be a sufficient condition for stability (for example for difference schemes involving 3 or more time levels). It can't deal easily with nonzero boundary conditions. In practice the method often gives useful results even when its application is not fully justified.

- If the exact solution of the PDE increases (exponentially) with time, the von Neumann stability condition that we need is

$$|\xi| \leq 1 + K\Delta t$$

for some positive K in the limit of small Δt.

Now that we have established that the FTCS scheme is not useful for $\Delta t > \frac{1}{2}\Delta x^2$, we look at other schemes which have better stability properties. We can group a whole class of schemes, including the FTCS scheme, into the name "θ-method".

2.6 The θ-method for $u_t = u_{xx}$

Recall that we derived the FTCS scheme for the heat equation by using a Forwards difference approximation in time and a Central difference approximation in space applied at the point (x_j, t_n). If instead we work at the forward point (x_j, t_{n+1}) and use a Backwards difference approximation in time and again a Central difference approximation in space, we get the BTCS scheme:

$$\frac{w_j^{n+1} - w_j^n}{\Delta t} = \frac{w_{j-1}^{n+1} - 2w_j^{n+1} + w_{j+1}^{n+1}}{\Delta x^2}$$

If we multiply through by Δt, define $r = \Delta t/\Delta x^2$ as before, and collect all $n+1$ terms on the left, we get

$$-rw_{j-1}^{n+1} + (1+2r)w_j^{n+1} - rw_{j+1}^{n+1} = w_j^n$$

Note that in contrast to the FTCS scheme, we now have three unknowns in this equation, the three values of w at the higher time level. This equation holds for $j = 1, 2, \ldots, J-1$, so in total we get $J-1$ equations for the $J-1$ unknowns $w_j^{n+1}, j = 1, 2, \ldots, J-1$. In matrix form this becomes

$$S\mathbf{w}^{n+1} = \mathbf{w}^n + \mathbf{b^{n+1}}$$

where

$$S = \begin{pmatrix} 1+2r & -r & 0 & \ldots & \\ -r & 1+2r & -r & 0 & \ldots \\ 0 & -r & 1+2r & -r & 0 \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ & \ddots & 0 & -r & 1+2r \end{pmatrix}, \quad \mathbf{w}^n = \begin{pmatrix} w_1^n \\ w_2^n \\ \vdots \\ \vdots \\ w_{J-1}^n \end{pmatrix}, \mathbf{b}^{n+1} = \begin{pmatrix} r\alpha(t_{n+1}) \\ 0 \\ \vdots \\ 0 \\ r\beta(t_{n+1}) \end{pmatrix}$$

We say this scheme is *implicit* because we need to solve a set of simultaneous equations for each time level.

Example: Solve $u_t = u_{xx}$ using the BTCS scheme with B.C. $u(0,t) = u(1,t) = 0, t > 0$, I.C. $u(x) = \sin(\pi x)$, and $r = 0.4, J = 4$.

Solution: The I.C.s tell us that

$$\mathbf{w}^0 = \left[0, 1/\sqrt{2}, 1, 1/\sqrt{2}, 0\right].$$

At $n = 1$ the B.C.s tell us that $w_0^1 = w_4^1 = 0$. Setting $n = 0$ in the BTCS scheme we get $(j = 1, 2, 3)$

$$\begin{aligned}(1+0.8)w_1^1 - 0.4w_2^1 &= w_1^0 + 0.4w_0^1\\ -0.4w_1^1 + (1+0.8)w_2^1 - 0.4w_3^1 &= w_2^0\\ -0.4w_2^1 + (1+0.8)w_3^1 &= w_3^0 + 0.4w_4^1\end{aligned}$$

or

$$\begin{aligned}1.8w_1^1 - 0.4w_2^1 &= \frac{1}{\sqrt{2}} + 0.4\times 0\\ -0.4w_1^1 + 1.8w_2^1 - 0.4w_3^1 &= 1\\ -0.4w_2^1 + 1.8w_3^1 &= \frac{1}{\sqrt{2}} + 0.4\times 0\end{aligned}$$

or

$$\begin{pmatrix} 1.8 & -0.4 & 0\\ -0.4 & 1.8 & -0.4\\ 0 & -0.4 & 1.8\end{pmatrix}\begin{pmatrix} w_1^1\\ w_2^1\\ w_3^1\end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}}\\ 1\\ \frac{1}{\sqrt{2}}\end{pmatrix}$$

Solving this by Gauss elimination gives

$$\mathbf{w}^1 = [0, 0.5729, 0.8102, 0.5729, 0]$$

to 4 significant figures. We now repeat this process to get $\mathbf{w}^2, \mathbf{w}^3$, etc. The exact result is

$$\mathbf{u}^1 = \exp(-\pi^2\Delta t)\sin(\pi x_j) = [0, 0.5525, 0.7813, 0.5525, 0]$$

A natural generalization of the FTCS and BTCS schemes is to take a weighted average of the two

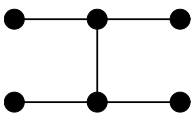
$$\frac{w_j^{n+1} - w_j^n}{\Delta t} = (1-\theta)\frac{w_{j-1}^n - 2w_j^n + w_{j+1}^n}{\Delta x^2} + \theta\frac{w_{j-1}^{n+1} - 2w_j^{n+1} + w_{j+1}^{n+1}}{\Delta x^2}.$$

This is the so-called θ method for $u_t = u_{xx}$, with $\theta \in [0, 1]$. Note that

(i) if $\theta = 0$ we get the FTCS scheme.

(ii) if $\theta = 1$ we get the BTCS scheme.

(iii) if $\theta > 0$ the scheme is implicit

(iv) If $\theta \in (0, 1)$, the computational molecule is  .

Another special value of θ is $\theta = 1/2$, which gives the so-called *Crank-Nicolson method*:

$$\frac{w_j^{n+1} - w_j^n}{\Delta t} = \frac{1}{2}\frac{w_{j-1}^n - 2w_j^n + w_{j+1}^n}{\Delta x^2} + \frac{1}{2}\frac{w_{j-1}^{n+1} - 2w_j^{n+1} + w_{j+1}^{n+1}}{\Delta x^2}.$$

Multiplying through by Δt and re-arranging in the usual way we get

$$-\frac{r}{2}w_{j-1}^{n+1} + (1+r)w_j^{n+1} - \frac{r}{2}w_{j+1}^{n+1} = \frac{r}{2}w_{j-1}^n + (1-r)w_j^n + \frac{r}{2}w_{j+1}^n.$$

2.6.1 LTE analysis of the θ-method

The PDE $u_t = u_{xx}$ is approximated on a uniform grid of size $\Delta x = 1/N$ in space and Δt in time, with approximate solution

$$w_j^n \approx u(x_j, t_n) \ , \quad x_j = x_0 + j\Delta x \ , t_n = n\Delta t,$$

where $u(x,t)$ is an exact solution of the PDE. The θ-method scheme can be written as

$$\frac{w_j^{n+1} - w_j^n}{\Delta t} = (1-\theta)\frac{\delta_x^2}{\Delta x^2} w_j^n + \theta \frac{\delta_x^2}{\Delta x^2} w_j^{n+1}$$

where $\delta_x^2 w_j^n = w_{j-1}^n - 2w_j^n + w_{j+1}^n$. The scheme can be written as

$$L_\Delta w_j^n \equiv \frac{F_t}{\Delta t} w_j^n - (1-\theta)\frac{\delta_x^2}{\Delta x^2} w_j^n - \theta \frac{\delta_x^2}{\Delta x^2} w_j^{n+1} = 0. \tag{2.9}$$

The LTE is found in the usual way by plugging in an exact solution $u(x_j, t_n)$ in place of the approximate solution w_j^n (for all j, n) into $L_\Delta w_j^n$, Taylor expanding about $(x,t) = (x_j, t_n)$ and eliminating terms using the PDE.

Remember, do not multiply or divide (2.9) by Δt or Δx when working out the LTE.

Recall that when applied to a smooth enough function $u(x,t)$ we can write

$$\begin{aligned} F_t u(x_j, t_n) &= u(x_j, t_n + \Delta t) - u(x_j, t_n) \\ &= \left[\Delta t \frac{\partial}{\partial t} + \frac{\Delta t^2}{2!}\frac{\partial^2}{\partial t^2} + \frac{\Delta t^3}{3!}\frac{\partial^3}{\partial t^3} + O(\Delta t^4) \right] u(x_j, t_n) \\ &= \left[\Delta t \, u_t + \frac{\Delta t^2}{2!} u_{tt} + \frac{\Delta t^3}{3!} u_{ttt} \right]_{(x_j, t_n)} + O(\Delta t^4). \end{aligned}$$

At line 2 above we have expanded in Taylor series, using the operator form of the series, and cancelled terms. Expanding the 2nd central space difference term $\delta_x^2 u(x_j, t_n)$ gives

$$\begin{aligned} \delta_x^2 u(x_j, t_n) &= u(x_j + \Delta x, t_n) - 2u(x_j, t_n) + u(x_j - \Delta x, t_n) \\ &= \left[\Delta x^2 \frac{\partial^2}{\partial x^2} + \frac{2\Delta x^4}{4!}\frac{\partial^4}{\partial x^4} + O(\Delta x^6) \right] u(x_j, t_n) \\ &= \left[\Delta x^2 \, u_{xx} + \frac{\Delta x^4}{12} u_{xxxx} \right]_{(x_j, t_n)} + O(\Delta x^6). \end{aligned}$$

(We have again missed out the details of the Taylor series expansion). The similar term $\delta_x^2 u(x_j, t_{n+1})$ (time level $n+1$) becomes

$$\begin{aligned} \delta_x^2 u(x_j, t_n + \Delta t) &= u(x_j + \Delta x, t_n + \Delta t) - 2u(x_j, t_n + \Delta t) + u(x_j - \Delta x, t_n + \Delta t) \\ &= \left[\Delta x^2 \, u_{xx} + \frac{\Delta x^4}{12} u_{xxxx} + O(\Delta x^6) \right]_{(x_j, t_n + \Delta t)}. \end{aligned}$$

Note that this term is evaluated at time $t = t_n + \Delta t$ and so it must also be expanded in Δt. That is

$$\begin{aligned}
\delta_x^2 u(x_j, t_n+\Delta t) &= \left[\Delta x^2\, u_{xx} + \frac{\Delta x^4}{12} u_{xxxx} + O(\Delta x^6)\right]_{(x_j, t_n+\Delta t)} \\
&= \left[1 + \Delta t \frac{\partial}{\partial t} + \frac{\Delta t^2}{2!}\frac{\partial^2}{\partial t^2} + \dots\right]\left[\Delta x^2\, u_{xx} + \frac{\Delta x^4}{12} u_{xxxx} + O(\Delta x^6)\right]_{(x_j, t_n)} \\
&= \left[\Delta x^2\, u_{xx} + \Delta t \Delta x^2\, u_{xxt} + \frac{\Delta x^4}{12} u_{xxxx}\right]_{(x_j, t_n)} + O(\Delta t \Delta x^4,\, \Delta t^2 \Delta x^2,\, \Delta x^6)
\end{aligned}$$

We now substitute each of the three terms into (2.9) and collect terms together to get

$$\text{LTE} = (u_t - u_{xx}) + \Delta t \left(\frac{1}{2} u_{tt} - \theta u_{txx}\right) - \frac{\Delta x^2}{12} u_{xxxx} + O(\Delta t^2,\, \Delta t \Delta x^2,\, \Delta x^4).$$

Since u is a solution of the PDE, this eliminates $u_t - u_{xx}$. Also, differentiating the PDE once with respect to t gives

$$u_{tt} = u_{txx} = u_{xxxx}$$

and hence

$$\text{LTE} = \left(\Delta t \left(\frac{1}{2} - \theta\right) - \frac{\Delta x^2}{12}\right) u_{xxxx} + O(\Delta t^2,\, \Delta t \Delta x^2,\, \Delta x^4).$$

If we do nothing special with the choice of $\theta, \Delta t, \Delta x$ then the LTE is $O(\Delta t,\, \Delta x^2)$. This is often described as *first order accurate in time and second order in space*. If we make the restriction that $\Delta t = O(\Delta x^2)$, then the LTE is $O(\Delta x^2)$ and the scheme is simply *second order accurate*. When $\theta = 1/2$ the $O(\Delta t)$ term is eliminated and the scheme is *second order accurate in time and space*.

Higher accuracy can be obtained by choosing parameters such that

$$\Delta t \left(\frac{1}{2} - \theta\right) - \frac{\Delta x^2}{12} = 0.$$

This can be achieved by choosing Δt, Δx and θ to satisfy

$$\theta = \frac{1}{2} - \frac{1}{12r} \quad \text{where } r = \frac{\Delta t}{\Delta x^2}\,.$$

In fact when we satisfy this condition and choose $\Delta t = O(\Delta x^2)$, the LTE is $O(\Delta x^4)$ and so the scheme is *fourth order accurate*. When $\theta = 0$ we have the FTCS scheme and the choice $r = 1/6$ used before satisfies the above requirements.

2.6.2 Stability analysis of the θ method

We now carry out a von Neumann analysis of the θ method

$$-\theta r w_{m-1}^{n+1} + (1+2\theta r) w_m^{n+1} - \theta r w_{m+1}^{n+1} = (1-\theta) r w_{m-1}^n + (1-2(1-\theta)r) w_m^n + (1-\theta) r w_{m+1}^n.$$

Substitute $w_m^n = \xi^n e^{im\omega}$ and simplify in the usual way

$$\begin{aligned}
&-\theta r e^{i(m-1)\omega}\xi^{n+1} + (1+2\theta r)e^{im\omega}\xi^{n+1} - \theta r e^{i(m+1)\omega}\xi^{n+1} = \\
&\quad (1-\theta) r e^{i(m-1)\omega}\xi^n + (1-2(1-\theta)r)e^{im\omega}\xi^n + (1-\theta) r e^{i(m+1)\omega}\xi^n \\
&\Rightarrow -\theta r e^{-i\omega}\xi + (1+2\theta r)\xi - \theta r e^{i\omega}\xi = (1-\theta) r e^{-i\omega} + (1-2(1-\theta)r) + (1-\theta) r e^{i\omega} \\
&\quad \text{(taking out factors } e^{im\omega} \text{ and } \xi^n) \\
&\Rightarrow \xi - \xi\theta r(e^{i\omega} - 2 + e^{-i\omega}) = 1 + (1-\theta) r (e^{i\omega} - 2 + e^{-i\omega})
\end{aligned}$$

Now

$$e^{i\omega} - 2 + e^{-i\omega} = -2(1-\cos(\omega)) = -4\sin^2(\omega/2)$$

so the above becomes

$$\begin{aligned}
\xi + 4\xi\theta \sin^2(\omega/2) r &= 1 - 4(1-\theta) r \sin^2(\omega/2) \\
\Rightarrow \xi &= \frac{1 - 4(1-\theta) r \sin^2(\omega/2)}{1 + 4\theta r \sin^2(\omega/2)}
\end{aligned}$$

We need $|\xi| \le 1$ for stability for all $\omega \in [-\pi, \pi]$. Since ξ is clearly real in this case this means we require $-1 \le \xi \le 1$. Now

$$\xi = \frac{1 + 4\theta r \sin^2(\omega/2) - 4r\sin^2(\omega/2)}{1 + 4\theta r \sin^2(\omega/2)} = 1 - \frac{4r\sin^2(\omega/2)}{1 + 4\theta r \sin^2(\omega/2)}$$

so ξ is clearly less than $+1$. Now consider the inequality $\xi \ge -1$. This is (on multiplying through by the denominator)

$$\begin{aligned}
-1 - 4\theta r \sin^2(\omega/2) &\le 1 - 4(1-\theta) r \sin^2(\omega/2) \\
&\Rightarrow 2(1-2\theta) r \sin^2(\omega/2) \le 1
\end{aligned}$$

(i) If $\theta \ge 1/2$, this last inequality will clearly hold for *all* r.

(ii) If $\theta < 1/2$, we need in the worst case ($\omega = \pi$) that

$$r \le \frac{1}{2(1-2\theta)}.$$

for stability.

We can check that when $\theta = 0$, we recover the familiar $r \le 1/2$ result for the FTCS scheme. In summary, the θ-scheme is stable

(i) For *all* r, if $\theta \ge 1/2$.

(ii) For

$$r \le \frac{1}{2(1-2\theta)}, \quad \text{if } \theta < 1/2.$$

2.6.3 Matrix form of the θ method

We can write the θ method in much the same form as the BTCS scheme,

$$S\mathbf{w}^{n+1} = M\mathbf{w}^n + (1-\theta)\mathbf{b}^n + \theta\mathbf{b}^{n+1}$$

where

$$S = \begin{pmatrix} 1+2\theta r & -\theta r & 0 & \dots & \\ -\theta r & 1+2\theta r & -\theta r & 0 & \dots \\ 0 & -\theta r & 1+2\theta r & -\theta r & 0 \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ & \ddots & 0 & -\theta r & 1+2\theta r \end{pmatrix}, \quad \mathbf{w}^n = \begin{pmatrix} w_1^n \\ w_2^n \\ \vdots \\ \vdots \\ w_{j-1}^n \end{pmatrix}, \mathbf{b}^n = \begin{pmatrix} r\alpha(t_n) \\ 0 \\ \vdots \\ 0 \\ r\beta(t_n) \end{pmatrix},$$

$$M = \begin{pmatrix} 1-2(1-\theta)r & (1-\theta)r & 0 & \dots & \\ (1-\theta)r & 1-2(1-\theta)r & (1-\theta)r & 0 & \dots \\ 0 & (1-\theta)r & 1-2(1-\theta)r & (1-\theta)r & 0 \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ & \ddots & 0 & (1-\theta)r & 1-2(1-\theta)r \end{pmatrix}.$$

with similar definitions for $\mathbf{w}^{n+1}$ and $\mathbf{b}^{n+1}$. So supposing we know $\mathbf{w}^n$, then $\mathbf{w}^{n+1}$ is computed by

(i) Set $\mathbf{q} = M\mathbf{w}^n + (1-\theta)\mathbf{b}^n + \theta\mathbf{b}^{n+1}$.

(ii) Solve $S\mathbf{v} = \mathbf{q}$ for $\mathbf{v}$.

(iii) Set $\mathbf{w}^{n+1} = \mathbf{v}$

The matrix S is tridiagonal (if $\theta > 0$), so solving (ii) is fairly quick and easy. This is still more work than solving the explicit FTCS scheme ($\theta = 0$) but not much more.

2.6.4 Summary

Scheme	Order	type	stability
$\theta = 0$	$O(\Delta t, \Delta x^2)$	explicit	$r \le 1/2$
$0 < \theta < 1/2$	$O(\Delta t, \Delta x^2)$	implicit	$r \le 1/2(1-2\theta)$
$\theta = 1/2$	$O(\Delta t^2, \Delta x^2)$	implicit	stable for *all* r
$1/2 < \theta \le 1$	$O(\Delta t, \Delta x^2)$	implicit	stable for *all* r

(Note the accuracy of some schemes can be increased by choosing a special value for r).

Although the FTCS scheme is easy to apply (because it is explicit), the time step is constrained by stability requirements to be $\Delta t \le 1/2\Delta x^2$, and since Δx is small, this implies Δt is very small. The θ-method for $\theta > 1/2$ allows a larger time step for stability (but not too large, otherwise the LTE gets big), and hence can require less overall computing. The Crank-Nicolson scheme has the added advantage of a higher order LTE.

2.6.5 Extension to more general PDEs

We can follow a similar approach to a more general PDE. For example consider the equation

$$u_t = u_{xx} + c_1 u_x + c_2 u$$

The corresponding FTCS scheme is

$$\begin{aligned}\frac{F_t}{\Delta t} w_j^n &= \frac{\delta_x^2}{\Delta x^2} w_j^n + c_1 \frac{D_x}{2\Delta x} w_j^n + c_2 w_j^n \\ \Rightarrow w_j^{n+1} &= (r - c_1 p) w_{j-1}^n + (1 - 2r + c_2 \Delta t) w_j^n + (r + c_1 p) w_{j+1}^n,\end{aligned}$$

where $r = \Delta t / \Delta x^2$, $p = \Delta t / 2\Delta x$.
The corresponding θ-method is

$$\begin{aligned}&\frac{F_t}{\Delta t} w_j^n = (1-\theta)\left(\frac{\delta_x^2}{\Delta x^2} w_j^n + c_1 \frac{D_x}{2\Delta x} w_j^n + c_2 w_j^n\right) + \theta\left(\frac{\delta_x^2}{\Delta x^2} w_j^{n+1} + c_1 \frac{D_x}{2\Delta x} w_j^{n+1} + c_2 w_j^{n+1}\right) \\ \Rightarrow &-\theta(r - c_1 p) w_{j-1}^{n+1} + (1 + 2\theta r - \theta \Delta t c_2) w_j^{n+1} - \theta(r + c_1 p) w_{j+1}^{n+1} = \\ &\quad (1-\theta)(r - c_1 p) w_{j-1}^n + (1 + (\Delta t c_2 - 2r)(1-\theta)) w_j^n + (1-\theta)(r + c_1 p) w_{j+1}^n.\end{aligned}$$

A von Neumann stability analysis shows that the extra term makes no difference to the results already obtained for the heat equation.

2.7 Other Boundary Conditions

So far we have dealt with Dirichlet BCs of the form

$$u(0,t) = \alpha(t), \qquad u(1,t) = \beta(t),$$

where $\alpha(t), \beta(t)$, are known functions of t. Other possible boundary conditions are Neumann (conditions on u_x at the boundary) or mixed (a mixture of Dirichlet and Neumann).

Suppose for example we have the following condition at $x = 0$ (similar calculations result if we have the condition at $x = 1$, although we will need to replace the Forwards Difference approximation at $x = 0$ by a Backwards Difference approximation at $x = 1$.)

$$u_x(0,t) = \gamma(t).$$

In the finite difference scheme, previously from $u(0,t_n) = \alpha(t_n)$ we knew that $w_0^n = \alpha(t_n)$. Now we do not have a value for $u(0,t)$ and need a way of calculating w_0^n.

2.7.1 Forward Difference approximation to u_x

The simplest (but least accurate) way to do this is by approximating $u_x(0,t)$ by a Forward Difference in space at $(0,t_n)$

$$u_x(0,t) \approx \frac{F_x}{\Delta x} u(0,t_n)$$

so the BC $u_x(0,t) = \gamma(t)$ becomes

$$\frac{F_x}{\Delta x} w_0^n = \frac{w_1^n - w_0^n}{\Delta x} = \gamma(t_n) \equiv \gamma_n$$

giving

$$w_1^n - w_0^n = \Delta x\, \gamma_n \qquad (*)$$

How we use this extra equation depends on out choice of scheme, in particular whether it is explicit or implicit.

If we are using an explicit scheme such as the FTCS scheme, at $j = 1$ we have

$$w_1^{n+1} = r w_0^n + (1 - 2r) w_1^n + r w_2^n.$$

As soon as we have w_1^{n+1} we can use (*) to calculate w_0^{n+1}.

$$w_0^{n+1} = w_1^{n+1} - \Delta x\, \gamma_{n+1}.$$

If we are working with an implicit scheme, there are two possible ways to proceed. The first is to add (*) to the set of Finite Difference equations which approximate the PDE. The second is to eliminate w_0^n directly by modifying the FD equations. For example, suppose we are using the Crank-Nicolson scheme. We can add (*) at $n+1$ to the set of equations, with a new unknown w_0^{n+1}, so that

$$\mathbf{w}^{n+1} = \left[w_0^{n+1}, w_1^{n+1}, \ldots, w_{j-1}^{n+1}\right]'.$$

so S now has an extra row and column, and the first two rows of S now look like

$$\begin{pmatrix} -1 & 1 & 0 & & \\ -\frac{1}{2}r & 1+r & -\frac{1}{2}r & 0 & \ldots \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ \ddots & \ddots & \ddots & \ddots & \ddots \end{pmatrix}$$

with

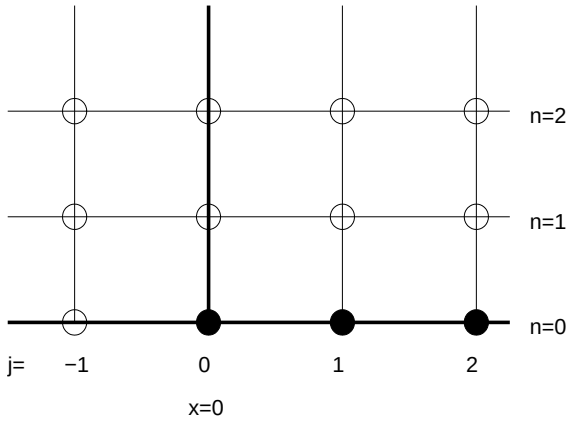
$$\mathbf{b}^{n+1} = [\Delta x \gamma_{n+1}, 0, \ldots,]$$

Alternatively we can eliminate w_0^{n+1} from (*) and the approximation to the PDE at $j = 1$, but this is not so useful since presumably we will still need to calculate w_0^{n+1}.

Although straightforward to apply, the FD method has the disadvantage that the approximation is only $O(\Delta x)$ accurate and hence spoils the $O(\Delta x^2)$ accuracy we are getting from the PDE scheme. A better method is the following:

2.7.2 Central Difference approximation to u_x

This method is also known as the fictitious point scheme, since the first step is to introduce a fictitious point $x = -\Delta x$ $(j = -1)$ *outside* the region $[0, 1]$ (see figure).



We now approximate $u_x(0,t)$ by a Central Difference in space at $(0,t_n)$

$$u_x(0,t) \approx \frac{D_x}{2\Delta x} u(0,t_n)$$

so the BC $u_x(0,t) = \gamma(t)$ becomes

$$\frac{D_x}{2\Delta x} w_0^n = \frac{w_1^n - w_{-1}^n}{2\Delta x} = \gamma_n$$

with accuracy $O(\Delta x^2)$. So now we have

$$w_1^n - w_{-1}^n = 2\Delta x\, \gamma_n \qquad (**)$$

The other essential step is to write down the PDE approximation *at* $(0,t_n)$ and eliminate the value w_{-1}^n between these two equations. For example, for the FTCS scheme, we get

$$\begin{aligned} w_0^{n+1} &= r w_{-1}^n + (1-2r) w_0^n + r w_1^n \\ &= r(w_1^n - 2\Delta x\, \gamma_n) + (1-2r) w_0^n + r w_1^n \\ &= (1-2r) w_0^n + 2r w_1^n - 2r\Delta x\, \gamma_n \end{aligned}$$

We now have an extra explicit equation for w_0^{n+1} involving only the points $(0,t_n)$ and $(\Delta x, t_n)$. When using (**) with an implicit scheme like the θ-method, we need to apply it at both n and $n+1$ to get a scheme connecting the values of w at $(0,t_n), (\Delta x, t_n), (0,t_{n+1}), (\Delta x, t_{n+1})$. For example, with the Crank-Nicolson Scheme we combine (**) at t_n and at t_{n+1} with the approximation to the PDE at $(0,t_n)$

$$-\frac{1}{2} r w_{-1}^{n+1} + (1+r) w_0^{n+1} - \frac{1}{2} r w_1^{n+1} = \frac{1}{2} r w_{-1}^n + (1-r) w_0^n + \frac{1}{2} r w_1^n$$

to get

$$\begin{aligned} -\frac{1}{2} r (w_1^{n+1} - 2\Delta x \gamma_{n+1}) + (1+r) w_0^{n+1} - \frac{1}{2} r w_1^{n+1} &= \frac{1}{2} r (w_1^n - 2\Delta x \gamma_n) + (1-r) w_0^n + \frac{1}{2} r w_1^n \\ \Rightarrow (1+r) w_0^{n+1} - r w_1^{n+1} &= (1-r) w_0^n + r w_1^n - r\Delta x (\gamma_n + \gamma_{n+1}) \end{aligned}$$

2.8 Multilevel schemes for $u_t = u_{xx}$

We now move to a class of schemes for the heat equation which involve more than 2 time levels. These are called *multilevel schemes*. The first of these is:

2.8.1 The Richardson scheme

In this scheme we approximate u_t by the Central Difference operator in time

$$u_t \approx \frac{D_t}{2\Delta t} u$$

so with the usual approximation in space at t_n we have

$$\frac{D_t}{2\Delta t} w_j^n \equiv \frac{w_j^{n+1} - w_j^{n-1}}{2\Delta t} = \frac{\delta_x^2}{\Delta x^2} w_j^n \equiv \frac{w_{j-1}^n - 2w_j^n + w_{j+1}^n}{\Delta x^2}$$

giving the Richardson scheme

$$w_j^{n+1} = w_j^{n-1} + 2r(w_{j-1}^n - 2w_j^n + w_{j+1}^n), \quad n \geq 1, j = 1, \ldots, J-1 \tag{2.10}$$

An examination of the LTE shows this scheme is in general $O(\Delta t^2, \Delta x^2)$. The scheme gives the solution at the $n+1$th time level in terms of the solutions at the nth and $n-1$th time levels. In order to start this off, we will need to calculate the $n = 1$ time level using a different scheme (for example the θ-method).

Stability Analysis of (2.10)

Set $w_j^n = \xi^n e^{i\omega j}$ in the usual way to get eventually a *quadratic* for ξ

$$\xi^2 + 8r \sin^2\left(\frac{1}{2}\omega\right) \xi - 1 = 0$$

This will have *two* solutions ξ_1, ξ_2. For stability we need *both* of them to be less than 1 in modulus, i.e (2.10) is only stable for values of r for which $|\xi_1|, |\xi_2| \leq 1$.

A useful trick is to note that if we write the quadratic as $(\xi - \xi_1)(\xi - \xi_2) = \xi^2 + b\xi + c$, then $b = -(\xi_1 + \xi_2), c = \xi_1\xi_2$. So in this case, since $|c| = 1$, and $|b| \neq 2$, we must have $|\xi_1||\xi_2| = 1$, and either (i) both ξ_i are complex with $|\xi_i| = 1$, or (ii) both ξ_i are real and distinct with one $|\xi_i| < 1$ and the other > 1. Since solving the quadratic gives

$$\xi_{1,2} = \xi_\pm = -4r \sin^2 \frac{\omega}{2} \pm \sqrt{1 + 16r^2 \sin^4 \frac{\omega}{2}}$$

then if both roots are real, one will have $|\xi| > 1$. In fact

$$\xi_- \quad = \quad -p - \sqrt{1+p^2}, \quad p = 4r \sin^2 \frac{\omega}{2} \geq 0$$

so clearly $|\xi_-| > 1$ when $p > 0$, and hence (2.10) is *unstable* for *all* r.

So as a practical scheme, the Richardson method is useless, but it can be modified to make the following useful scheme:

2.8.2 The Du Fort-Frankel scheme

This is similar to Richardson's method, but the term w_j^n on the right-hand side is replaced by the average of its two nearest neighbours in time: $(w_j^{n+1} + w_j^{n-1})/2$. So the approximate operator L_Δ is

$$L_\Delta w_j^n = \frac{w_j^{n+1} - w_j^{n-1}}{2\Delta t} - \left(\frac{w_{j-1}^n - w_j^{n-1} - w_j^{n+1} + w_{j+1}^n}{\Delta x^2} \right) \tag{2.11}$$

The scheme for $u_t = u_{xx}$ can therefore be arranged as

$$w_j^{n+1} = \frac{1-2r}{1+2r}\, w_j^{n-1} + \frac{2r}{1+2r}\, \left(w_{j-1}^n + w_{j+1}^n \right), \tag{2.12}$$

where (as usual) $r = \Delta t/\Delta x^2$. So this scheme is *explicit* but needs to have $\boldsymbol{w}^1$ supplied by a different scheme.

Exercise: Show that the LTE of (2.11) is

$$\text{LTE } = \left(r^2 - \frac{1}{12} \right) u_{tt}\, \Delta x^2 + \frac{\Delta t^2}{6} u_{ttt} + O(\Delta t^4, \Delta x^4, r^4 \Delta x^6)$$

i.e. it is **second order in time and space**. However, note that we have **implicitly assumed** that $\Delta t = O(\Delta x^2)$ in order to deduce second order accuracy, because of the r^2 factor in the first term of the LTE. If for example we set $\Delta t = \Delta x$, then $r = 1/\Delta x$ and the leading term of the LTE is u_{tt}. Hence the LTE doesn't tend to zero as Δt, $\Delta x \to 0$.
Note also that if we set $\Delta t = r\Delta x^2$ with $r = \sqrt{1/12}$ then the method is **fourth order accurate**.

Exercise: Show that the amplification factor in the von Neumann stability analysis of the Du Fort-Frankel scheme (2.12) satisfies the quadratic equation

$$(1+2r)\xi^2 - 4r\xi \cos\omega + 2r - 1 = 0,$$

where ω is the frequency parameter.

The roots of the quadratic are

$$\xi_\pm = \frac{2r\cos\omega \pm \sqrt{1 - 4r^2 \sin^2\omega}}{1+2r}.$$

We shall examine the scheme's stability by considering two cases separately.

(i) $4r^2 \sin^2\omega \le 1$, so both roots are real. It then follows that

$$\xi_+ = \frac{2r\cos\omega + \sqrt{1 - 4r^2\sin^2\omega}}{1+2r} \le \frac{2r\cos\omega + 1}{1+2r} \le 1,$$

since $\cos\omega \le 1$ for all ω, and we also have (since $\cos\omega \ge -1$ for all ω)

$$\xi_+ \ge \frac{-2r + \sqrt{1 - 4r^2\sin^2\omega}}{1+2r} \ge \frac{-2r}{1+2r} > -1,$$

i.e. $-1 < \xi_+ \le 1$. It can similarly be shown that $-1 \le \xi_- < 1$, so in this case both roots satisfy $|\xi| \le 1$.

(ii) $4r^2 \sin^2 \omega > 1$, so both roots of the quadratic are complex, i.e. $\xi_\pm = \alpha \pm i\beta$, where

$$\alpha = \frac{2r\cos\omega}{1+2r}, \quad \beta = \frac{\sqrt{4r^2\sin^2\omega - 1}}{1+2r}.$$

This means that $|\xi_+| = |\xi_-|$. But the product of the two roots is $(2r-1)/(1+2r)$ and so both roots must satisfy

$$|\xi|^2 = \frac{|2r-1|}{1+2r} \leq 1$$

for all $r > 0$, and so $|\xi| \leq 1$ for any r.

Thus we have shown in both cases, that if ξ is a root of the quadratic then $|\xi| \leq 1$ for all r and all $\omega \in [-\pi, \pi]$. Hence the Du Fort-Frankel scheme (2.12) is **unconditionally stable** (stable for all $r > 0$).

Comparison of methods:

Scheme	Comments	Stability limit	LTE
Du Fort-Frankel	explicit, but need to use a different scheme to get $\boldsymbol{w}^1$	OK $\forall\, r$	$O(r^2\Delta x^2, \Delta x^2, \Delta t^2)$. Second order if $\Delta t = O(\Delta x^2)$, fourth order if $\Delta t = \Delta x^2/\sqrt{12}$
Crank-Nicolson ($\theta = 1/2$)	implicit: need to solve a tridiagonal system (not too bad)	OK $\forall\, r$	$O(\Delta x^2, \Delta t^2)$. Second order in space and time separately
FTCS ($\theta = 0$)	explicit	OK if $r \leq 1/2$	$O(\Delta x^2, \Delta t)$. Second order if $\Delta t = O(\Delta x^2)$, fourth order if $\Delta t = \Delta x^2/6$

2.9 Convergence

Suppose we use the difference operator L_Δ to approximate $L = \partial/\partial t - \partial^2/\partial x^2$, and that u is the exact solution ($Lu = 0$) and w_j^n the corresponding approximate solution ($L_\Delta w_j^n = 0$). The approximation will only be useful if w_j^n tends to u as Δx, $\Delta t \to 0$. Roughly speaking this is what we mean by *convergence*.

Fix $x^* \in (0,1)$ and $t^* > 0$. The exact solution here is $u(x^*, t^*)$. The approximate solution is w_j^n, where $j\Delta x = x^*$, $n\Delta t = t^*$. We want to see what happens to the approximate solution when we let Δx, $\Delta t \to 0$ and j, $n \to \infty$ in such a way that $j\Delta x$ and $n\Delta t$ remain fixed at x^* and t^* respectively. i.e. we choose $\Delta x = x^*/j$, $\Delta t = t^*/n$ and let j, $n \to \infty$. We can then write $w_j^n = w_{x^*/\Delta x}^{t^*/\Delta t}$, and in this notation we have the following definition.

Definition (Convergence)
The approximate solution w_j^n converges to the exact solution u at (x^*, t^*) if

$$\left| u(x^*, t^*) - w_{x^*/\Delta x}^{t^*/\Delta t} \right| \to 0$$

as Δx, $\Delta t \to 0$.

We have only space here to mention one result on convergence. We say that a scheme is *consistent* if its LTE $\to 0$ as Δx, $\Delta t \to 0$, and is *stable* if w_j^n remains bounded as $n \to \infty$ for fixed $\Delta t, \Delta x$. There is a nice theorem that covers all the analysis.

Lax Equivalence Theorem

Given a properly posed linear initial value problem and a finite difference approximation of it that satisfies the *consistency* condition (i.e. its LTE $\to 0$ as Δx, $\Delta t \to 0$) and the *stability* condition, then the scheme *converges*.
In fact the full result is actually stronger than this: if the scheme is consistent then

$$\text{stability} \iff \text{convergence}.$$

Therefore it is enough to establish stability and consistency to get convergence. On the other hand, instability rules out convergence.

2.10 Approximations to more general parabolic PDEs

- **Reaction-Diffusion equations**

  The general form is

  $$u_t = \kappa u_{xx} + f(x, t, u)$$

  where the first term on the rhs is the diffusion term and the other is the reaction term. We approximate the u_t and the u_{xx} term in the usual way (note that r becomes κr and stability results apply to this new value, i.e. the FTCS scheme is unstable for $\kappa r > 1/2$. The reaction term is approximated by $f(x_j, t_n, w_j^n)$ at time level n. If $f(x, t, u)$ is nonlinear in u, and if we use an implicit scheme, then we will end up with a set of *nonlinear* equations for w_j^{n+1} at each time level.

- **Linear equations with Varying coefficients**

  A typical equation is

  $$u_t = A(x, t)u_{xx} + B(x, t)u_x + C(x, t)u$$

  Again we approximate the terms in the usual way, except that we replace $A(x, t)$ by $A_j^n = A(x_j, t_n)$, etc.

- **The Black-Scholes equation**

  This is a specific case of the linear equation with variable coefficients. The BS equation describes the value of an *option* to buy shares at time T at the price E. If $S(T)$ is the value of the share price at $t = T$, and if $S(T) > E$, buy them (exercise the option), if $S(T) \leq E$, don't buy (no profit). What is the value of this option $V(t, S)$ at $t = 0$?

  It satisfies the Black-Scholes PDE

  $$V_t + \rho S V_s + \frac{1}{2}\sigma^2 S^2 V_{ss} - \rho V = 0, \quad t \in [0, T]$$

  Where ρ is the interest rate, σ is the share volatility, and S is the share price. The boundary conditions are $V(0, t) = 0$ and $\lim_{S\to\infty} V(S, t)/S = 1$, since $(V \sim S - E)$. The final condition is $V(S, T) = \max(S - E, 0)$, E given. We know $S = S_0$ at $t = 0$ (i.e. now) and want to work out $V(S_0, 0)$. We approximate $V(S_j, t_n) \approx W_j^n$.

  Approximating for the terms in the usual way we get for example for the FTCS scheme:

  $$\frac{W_j^{n+1} - W_j^n}{\Delta t} + \rho S_j \frac{W_{j+1}^n - W_{j-1}^n}{2\Delta S} + \frac{1}{2}\sigma^2 S_j^2 \frac{W_{j-1}^n - 2W_j^n + W_{j+1}^n}{\Delta S^2} - \rho W_j^n = 0$$

  However the method of solutions is a little different, we solve this starting at $t = T$ and working *backwards* in time to get to $t = 0$. Then we see if the computed value $V(S_0, 0)$ is higher or lower than the price being asked for the option.

2.11 More space dimensions

So far we have looked at the PDE $u_t = u_{xx}$, plus variations, but always with time and one space-like dimension such as x as the independent variables. When more than one space dimensions are involved (usually the case in real life), we have to deal with equations such as $u_t = u_{xx} + u_{yy}$ or $u_t = u_{xx} + u_{yy} + u_{zz}$ with solutions $u(x, y, t)$ or $u(x, y, z, t)$ and approximate solutions $w^n_{j,k} \approx u(x_j, y_k, t_n)$ and $w^n_{j,k,\ell} \approx u(x_j, y_k, z_\ell, t_n)$. Now the space grid is 2D or 3D, but we use the same principles to develop a numerical scheme.

For example, let us develop a FTCS scheme for the 2D heat equation $u_t = u_{xx} + u_{yy}$ defined in a rectangle of size $J\Delta x \times K\Delta y$. We approximate u_t and u_{xx} as before, and u_{yy} as for u_{xx} using δ^2_y.

$$\begin{aligned} \frac{F_t}{\Delta t} w^n_{j,k} &= \frac{w^{n+1}_{j,k} - w^n_{j,k}}{\Delta t} = \left(\frac{\delta^2_x}{\Delta x^2} + \frac{\delta^2_y}{\Delta y^2} \right) w^n_{j,k} \\ &= \frac{w^n_{j-1,k} - 2w^n_{j,k} + w^n_{j+1,k}}{\Delta x^2} + \frac{w^n_{j,k-1} - 2w^n_{j,k} + w^n_{j,k+1}}{\Delta y^2} \end{aligned}$$

giving the scheme

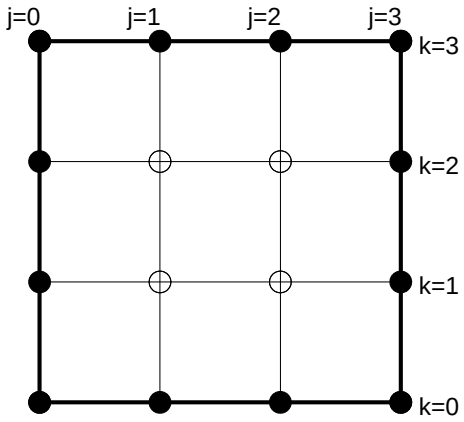
$$w^{n+1}_{j,k} = w^n_{j,k} + r_x(w^n_{j-1,k} - 2w^n_{j,k} + w^n_{j+1,k}) + r_y(w^n_{j,k-1} - 2w^n_{j,k} + w^n_{j,k+1}) \tag{2.13}$$

with $r_x = \Delta t/\Delta x^2$, $r_y = \Delta t/\Delta y^2$. At each time level n we have a $(J + 1) \times (K + 1)$ grid of values $w^n_{j,k}, j = 0, \ldots, J; k = 0, \ldots, K$. With the boundary values given on the sides of the rectangle, applying (2.13) at each interior point gives the values at the next time level $n + 1$.

2.11.1 Numerical example: the FTCS scheme for the 2D heat equation

Suppose we wish to solve $u_t = u_{xx} + u_{yy}$ on the unit square with $\Delta x = \Delta y = 1/3$, initial conditions $u(x, y, 0) = \sin(\pi x/2)\sin(\pi y)$, and boundary conditions $u(0, y, t) = u(x, 1, t) = u(x, 0, t) = 0$, $u(1, y, t) = \sin(\pi y)$. Carry out two steps of the scheme with $r = 0.25$.

The grid for this scheme at each time level looks like this:



We have the initial conditions $w^0_{j,k}$ given by the formula $\sin(\pi j/6)\sin(\pi k/3), j = 0, \ldots, 3; k = 0, \ldots, 3$. The BCs give $w_{0,j} = w_{j,0} = w_{j,3} = 0, w_{3,j} = \sin(\pi j/3)$, so the values of $w^0_{j,k}$ look like this:

$k \setminus j$	0	1	2	3
0	0	0	0	0
1	0	0.4330	0.7500	0.8660
2	0	0.4330	0.7500	0.8660
3	0	0	0	0

We now apply (2.13) for $j = 1, 2; k = 1, 2, n = 0$ to get the values of $w^1_{j,k}$

$k \setminus j$	0	1	2	3
0	0	0	0	0
1	0	0.2958	0.5123	0.8660
2	0	0.2958	0.5123	0.8660
3	0	0	0	0

We now apply (2.13) for $j = 1, 2; k = 1, 2, n = 1$ to get the values of $w^2_{j,k}$

$k \setminus j$	0	1	2	3
0	0	0	0	0
1	0	0.2020	0.4185	0.8660
2	0	0.2020	0.4185	0.8660
3	0	0	0	0

(all results to 4D). Note that the solution is symmetric about the line $y = 1/2$, which we could expect since the PDE, the initial conditions and the boundary conditions satisfy this symmetry. We now example the stability of this scheme.

2.11.2 Stability of the FTCS scheme for the 2D heat equation

We investigate the von Neumann stability of the scheme (2.13) by following the steps below.

1. Substitute $w^n_{j,k} = \xi^n \exp(ij\alpha) \exp(ik\beta)$ into the scheme (2.13).
2. Divide through by $\xi^n \exp(ij\alpha) \exp(ik\beta)$ and rearrange to get an expression for ξ.
3. Find conditions on the mesh ratios that guarantee $|\xi| \leq 1$ for all $(\alpha, \beta) \in [-\pi, \pi]^2$.

Step 1 gives

$$\begin{aligned}(\xi^{n+1} - \xi^n)\, e^{ij\alpha}\, e^{ik\beta} &= r_x\, \xi^n\, e^{ik\beta} \left[e^{i(j-1)\alpha} - 2e^{ij\alpha} + e^{i(j+1)\alpha} \right] \\ &\quad + r_y\, \xi^n\, e^{ij\alpha} \left[e^{i(k-1)\beta} - 2e^{ik\beta} + e^{i(k+1)\beta} \right],\end{aligned}$$

and simplifying (step 2) gives

$$\begin{aligned}\xi - 1 &= r_x \left(e^{-i\alpha} - 2 + e^{i\alpha} \right) + r_y \left(e^{-i\beta} - 2 + e^{i\beta} \right) \\ &= -4r_x \sin^2 \frac{\alpha}{2} - 4r_y \sin^2 \frac{\beta}{2}.\end{aligned}$$

Note that the second central difference operators δ_x^2 and δ_y^2 become $(2\cos\alpha - 2) = -4\sin^2\frac{\alpha}{2}$ and $(2\cos\beta - 2) = -4\sin^2\frac{\beta}{2}$ after simplification.

For step 3 we first look at the case of a square spatial mesh, i.e. when $\Delta x = \Delta y$, so that ξ is given by

$$\xi = 1 - 4r\,\sin^2\frac{\alpha}{2} - 4r\,\sin^2\frac{\beta}{2} \quad \text{where } r = r_x = r_y.$$

The amplification factor ξ is real, and so $|\xi| \le 1$ is equivalent to $-1 \le \xi \le 1$. Hence we need to find conditions on r that guarantee that the minimum value of ξ is greater than -1 and the maximum value is less than 1 for all α, $\beta \in [-\pi, \pi]$. The max and min values of ξ occur at the maximum and minimum values of the sine functions (because $r \ge 0$), i.e. at $(\alpha, \beta) = (\pm\pi, \pm\pi)$ and $(\alpha, \beta) = (0, 0)$ respectively. So

$$1 - 8r \le \xi \le 1 \quad \text{for all } \alpha,\, \beta \in [-\pi, \pi].$$

For stability we therefore require $1 - 8r \ge -1$, i.e. the scheme is only stable when $r \le 1/4$.

If the space grid is not square then the stability analysis is almost the same but now

$$\xi = 1 - 4r_x\,\sin^2\frac{\alpha}{2} - 4r_y\,\sin^2\frac{\beta}{2}$$

and

$$\max_{\alpha,\beta}\xi = 1, \quad \min_{\alpha,\beta}\xi = 1 - 4r_x - 4r_y$$

so that the scheme is stable if and only if

$$0 \le r_x + r_y \le \frac{1}{2}.$$

Exercise: Verify this and show that it gives the square mesh limit when $\Delta x = \Delta y$.

When $r_x = r_y, \Delta x = \Delta y$, the stability limit is twice as bad as for the 1D case, and there is much more work at each time level since we have M^2 equations to calculate at each time level. It is thus even more important than before to develop schemes which are more efficient than the FTCS scheme.

In 1D we saw that implicit schemes were more stable than the explicit FTCS scheme, and hence could be used with a larger value of Δt. This suggests trying to develop implicit schemes in 2D. Our first attempt will be a 2D version of the θ-method.

The 2D θ-method

It is straightforward to develop a 2D version of this scheme in the same way as for the FTCS scheme. For simplicity we work with $\theta = \frac{1}{2}$, but the same principles apply for general values of $\theta > 0$. Write

$$\begin{aligned}
\frac{F_t}{\Delta t}w_{j,k}^n &= \frac{w_{j,k}^{n+1} - w_{j,k}^n}{\Delta t} = \frac{1}{2}\frac{\delta_x^2}{\Delta x^2}\left(w_{j,k}^n + w_{j,k}^{n+1}\right) + \frac{1}{2}\frac{\delta_y^2}{\Delta y^2}\left(w_{j,k}^n + w_{j,k}^{n+1}\right) \\
&= \frac{w_{j-1,k}^n - 2w_{j,k}^n + w_{j+1,k}^n}{2\Delta x^2} + \frac{w_{j,k-1}^n - 2w_{j,k}^n + w_{j,k+1}^n}{2\Delta y^2} + \\
&\quad + \frac{w_{j-1,k}^{n+1} - 2w_{j,k}^{n+1} + w_{j+1,k}^{n+1}}{2\Delta x^2} + \frac{w_{j,k-1}^{n+1} - 2w_{j,k}^{n+1} + w_{j,k+1}^{n+1}}{2\Delta y^2}
\end{aligned}$$

taking all the unknown $w_{j,k}^{n+1}$ to the left gives the scheme

$$w_{j,k}^{n+1} - \frac{1}{2}r_x\delta_x^2 w_{j,k}^{n+1} - \frac{1}{2}r_y\delta_y^2 w_{j,k}^{n+1} = w_{j,k}^n + \frac{1}{2}r_x\delta_x^2 w_{j,k}^n + \frac{1}{2}r_y\delta_y^2 w_{j,k}^n, \tag{2.14}$$

or in full

$$w_{j,k}^{n+1} - \frac{1}{2}r_x(w_{j-1,k}^{n+1} - 2w_{j,k}^{n+1} + w_{j+1,k}^{n+1}) - \frac{1}{2}r_y(w_{j,k-1}^{n+1} - 2w_{j,k}^{n+1} + w_{j,k+1}^{n+1}) =$$
$$w_{j,k}^n + \frac{1}{2}r_x(w_{j-1,k}^n - 2w_{j,k}^n + w_{j+1,k}^n) + \frac{1}{2}r_y(w_{j,k-1}^n - 2w_{j,k}^n + w_{j,k+1}^n).$$

The problem with (2.14) is that it has 5 unknowns, and if we write down (2.14) at each interior point $(x_j, y_k), j = 1, \ldots, J-1; k = 1, \ldots, K-1$, we end up with $(J-1) \times (K-1)$ equations for the $(J-1) \times (K-1)$ unknowns $w_{j,k}^{n+1}, j = 1, \ldots, J-1; k = 1, \ldots, K-1$. Although sparse, the matrix of coefficients no longer has a simple tri-diagonal structure as in 1D.

Ignoring sparseness, and writing $(J-1)\times(K-1) \approx MJ$, we recall that a system of N equations generally requires $\frac{1}{3}N^3$ floating point operations to solve it using Gaussian elimination. In this case then we will have approx $\frac{1}{3}J^3K^3$ operations, and even with small values for J and K like 10 we will have around 3×10^5 operations at *each* time step to carry out.

We need another idea to be able to develop a more practical scheme in 2D, the so-called ADI method.

2.12 Alternating direction implicit (ADI) schemes

To help us here it is useful to develop a more compact notation, the **Exponential operator notation**.

We start out by observing that when u is *well enough behaved*, the Taylor expansion of $u(x_j + \Delta x, y_k, t_n)$ can be written as

$$\begin{aligned} u(x_j+\Delta x, y_k, t_n) &= \left[u + \Delta x\frac{\partial u}{\partial x} + \frac{\Delta x^2}{2!}\frac{\partial^2 u}{\partial x^2} + \frac{\Delta x^3}{3!}\frac{\partial^3 u}{\partial x^3} + \ldots\right]_{(x_j,y_k,t_n)} \\ &= \left[1 + \Delta x\frac{\partial}{\partial x} + \frac{\Delta x^2}{2!}\frac{\partial^2}{\partial x^2} + \frac{\Delta x^3}{3!}\frac{\partial^3}{\partial x^3} + \ldots\right] u\mid_{(x_j,y_k,t_n)} \\ &= \exp\left(\Delta x\frac{\partial}{\partial x}\right) u\mid_{(x_j,y_k,t_n)}. \end{aligned}$$

Similarly (missing out the steps in between) we can write

$$u(x_j, y_k + \Delta y, t_n) = \exp\left(\Delta y\frac{\partial}{\partial y}\right) u\mid_{(x_j,y_k,t_n)}$$

and

$$u(x_j, y_k, t_n + \Delta t) = \exp\left(\Delta t\frac{\partial}{\partial t}\right) u\mid_{(x_j,y_k,t_n)}$$

We now use the exponential operator notation to derive a different implicit scheme for $u_t = u_{xx} + u_{yy}$. Suppose that $u(x, y, t)$ is a smooth solution of the PDE. Taylor expanding

$u(x, y, t_n + \Delta t)$ about (x, y, t_n) gives

$$u(x, y, t_n+\Delta t) = \exp\left(\Delta t \frac{\partial}{\partial t}\right) u\,|_{(x,y,t_n)}\,.$$

Use $e^a = e^{a/2} \cdot e^{a/2}$ to rewrite this as

$$u|_{t=t_n+\Delta t} = \exp\left(\frac{\Delta t}{2}\frac{\partial}{\partial t}\right) \exp\left(\frac{\Delta t}{2}\frac{\partial}{\partial t}\right) u|_{t=t_n}$$

and hence

$$\underbrace{\exp\left(-\frac{\Delta t}{2}\frac{\partial}{\partial t}\right) u|_{t=t_n+\Delta t}}_{\textbf{new time-level}} = \underbrace{\exp\left(\frac{\Delta t}{2}\frac{\partial}{\partial t}\right) u|_{t=t_n}}_{\textbf{old time-level}}. \tag{2.15}$$

We now use the fact that u solves the PDE to write

$$\begin{aligned}\exp\left(\pm\frac{\Delta t}{2}\frac{\partial}{\partial t}\right) u &= \exp\left(\pm\frac{\Delta t}{2}\left[\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right]\right) u \\ &= \exp\left(\pm\frac{\Delta t}{2}\frac{\partial^2}{\partial x^2}\right)\exp\left(\pm\frac{\Delta t}{2}\frac{\partial^2}{\partial y^2}\right) u\end{aligned}$$

(using $e^{a+b} = e^a \cdot e^b$). Plugging this into (2.15) gives

$$\exp\left(-\frac{\Delta t}{2}\frac{\partial^2}{\partial x^2}\right)\exp\left(-\frac{\Delta t}{2}\frac{\partial^2}{\partial y^2}\right) u|_{t=t_n+\Delta t} = \exp\left(\frac{\Delta t}{2}\frac{\partial^2}{\partial x^2}\right)\exp\left(\frac{\Delta t}{2}\frac{\partial^2}{\partial y^2}\right) u|_{t=t_n}\,.$$

We now chop the exponentials at first order ($e^{\pm q} \approx 1 \pm q$) to get

$$\left(1-\frac{\Delta t}{2}\frac{\partial^2}{\partial x^2}\right)\left(1-\frac{\Delta t}{2}\frac{\partial^2}{\partial y^2}\right) u|_{t=t_n+\Delta t} \approx \left(1+\frac{\Delta t}{2}\frac{\partial^2}{\partial x^2}\right)\left(1+\frac{\Delta t}{2}\frac{\partial^2}{\partial y^2}\right) u|_{t=t_n}\,,$$

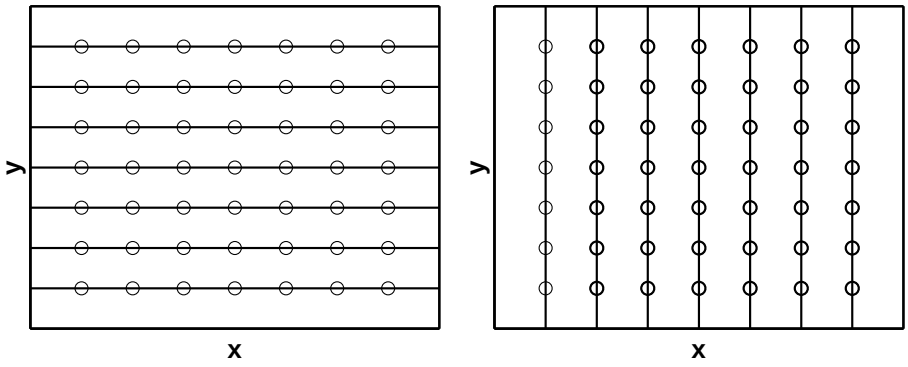
and finally use second central differences to approximate the space derivatives at $x = x_j$, $y = y_k$ to end up with the scheme

$$\left(1-\frac{r_x}{2}\delta_x^2\right)\left(1-\frac{r_y}{2}\delta_y^2\right) w_{j,k}^{n+1} = \left(1+\frac{r_x}{2}\delta_x^2\right)\left(1+\frac{r_y}{2}\delta_y^2\right) w_{j,k}^{n}, \tag{2.16}$$

where r_x, r_y defined as before.

This just looks like another (slow) implicit scheme. But it is actually **much easier and quicker to use than it looks**. It splits into two stages involving an intermediate quantity $v_{j,k}$:

$$\left.\begin{aligned}\textbf{Stage 1:}\quad \left(1-\frac{r_x}{2}\delta_x^2\right) v_{j,k} &= \left(1+\frac{r_y}{2}\delta_y^2\right) w_{j,k}^{n} \\ \textbf{Stage 2:}\quad \left(1-\frac{r_y}{2}\delta_y^2\right) w_{j,k}^{n+1} &= \left(1+\frac{r_x}{2}\delta_x^2\right) v_{j,k}\end{aligned}\right\} \tag{2.17}$$



Stage 1 (on left), Stage 2 (on right)

Stage 1 involves solving a tridiagonal system of equations along each row of the solution in the $x-$direction (which can be done quickly); **Stage 2** is similar, but in the $y-$direction (see figure)

The name ADI comes from this idea of alternately solving along the $x-$direction and $y-$direction.

We need to verify that the two stages of (2.17) are equivalent to the full scheme (2.16). Applying $\left(1-\frac{1}{2}r_x\delta_x^2\right)$ to Stage 2 gives

$$\begin{aligned}\left(1-\frac{1}{2}r_x\delta_x^2\right)\left(1-\frac{1}{2}r_y\delta_y^2\right)w_{j,k}^{n+1} &= \left(1-\frac{1}{2}r_x\delta_x^2\right)\left(1+\frac{1}{2}r_x\delta_x^2\right)v_{j,k}\\ &= \left(1+\frac{1}{2}r_x\delta_x^2\right)\left(1-\frac{1}{2}r_x\delta_x^2\right)v_{j,k} \quad \text{(difference operators commute)}\\ &= \left(1+\frac{1}{2}r_x\delta_x^2\right)\left(1+\frac{1}{2}r_y\delta_y^2\right)w_{j,k}^{n} \quad \text{by Stage 1.}\end{aligned}$$

So it does reproduce (2.16).

Why is all this faster than the 2D θ-method? Because the matrices at each step are triadiagonal, and because they involve only J or K unknowns, they can be solved very efficiently in $O(J)$ or $O(K)$ operations. So although we have K or J separate equations in J or K unknowns, the total number of operations required for *each* time step are just of order $O(JK)$. Compare this with the $O(J^3K^3)$ estimate above, and you will see that the ADI method implies a large saving in computer time.

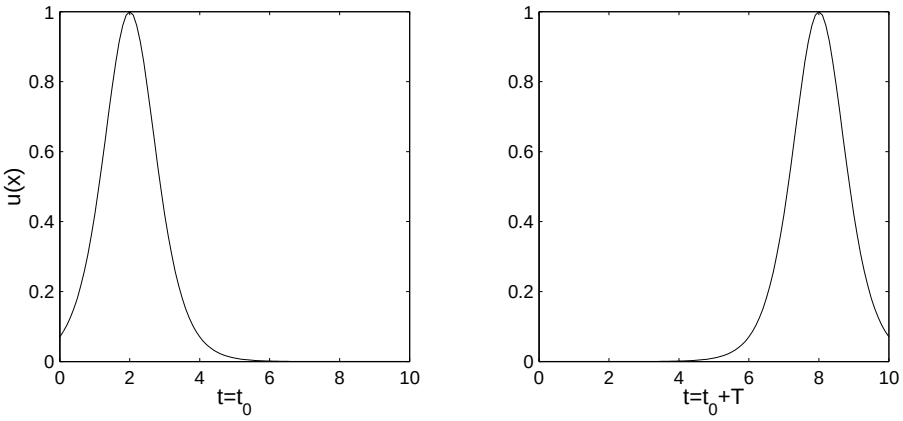
Stability

Exercise: Verify that if the mesh is square (i.e $\Delta x = \Delta y$ and so $r_x = r_y = r$) then the ADI scheme is **stable for all** r.

3 Hyperbolic PDEs

3.1 Introduction

Hyperbolic PDEs describe wave propagation problems, for example waves in water, gas, plasmas, traffic flow, etc. In a number of simple cases the wave moves along with unchanged form



In this case the wave moves from left to right, with a certain wave speed, and the wave form is not damped in time. The simplest hyperbolic equation exhibiting solutions of this sort is the first order *advection equation* for $u(x,t)$

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0, \qquad a \text{ const.} \tag{3.1}$$

$$\text{with IC: } u(x,0) = F(x) \tag{3.2}$$

This equation is particularly simple but it is of interest since it displays much of the phenomena which is specific to hyperbolic PDEs. Hence it is a useful test for schemes for solving hyperbolic PDEs (analogous to the study of $y' = \lambda y$ in the numerical study of ODEs). Furthermore (3.1) is easily generalised to more complicated examples such as

- advection equation with variable coefficient

$$\frac{\partial u}{\partial t} + a(x)\frac{\partial u}{\partial x} = 0$$

- The wave equation

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2}$$

- higher dimensions

$$\frac{\partial u}{\partial t} + a_x\frac{\partial u}{\partial x} + a_y\frac{\partial u}{\partial y} = 0$$

- nonlinear equations, for example Burger's equation

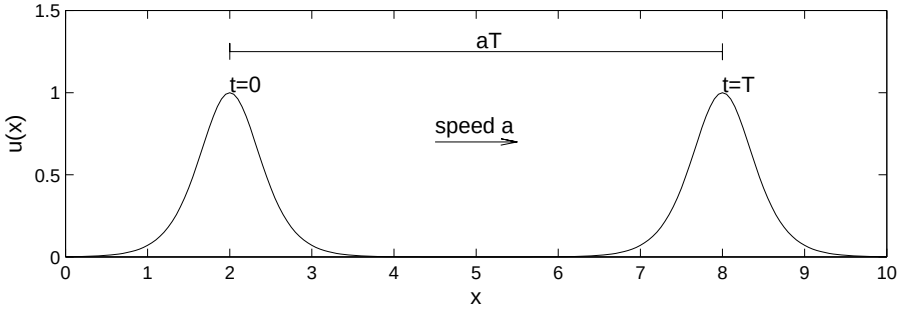
$$\frac{\partial u}{\partial t} + \frac{1}{2}\frac{\partial (u^2)}{\partial x} = 0$$

Return to (3.1) and consider the pure initial value problem, i.e. assume (3.1) holds for all x. Then the exact solution is

$$u(x,t) = F(x - at)$$

Exercise: check this!

For example with a pulse-like initial condition, if $a > 0$, we have something like the figure below



If $a < 0$ the wave moves to the left instead.

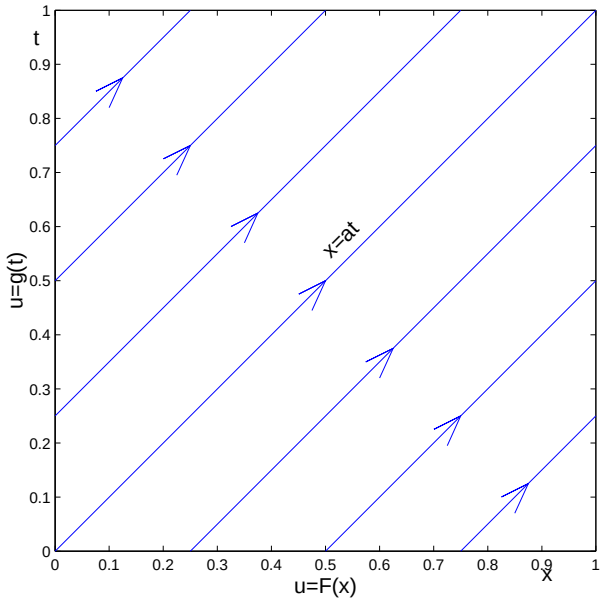
More typically (3.1) is defined on a finite interval, e.g. $\in (0,1)$. In that case we have to supply only *one* boundary condition.

$$\text{if } \left\{ \begin{array}{l} a > 0 \\ a < 0 \end{array} \right\} \text{ we specify } \left\{ \begin{array}{l} u(0,t) = g(t) \\ u(1,t) = g(t) \end{array} \right.$$

For example, if $a > 0$, from the initial condition $u(x,0) = F(x)$ and the left-hand boundary condition $u(0,t) = g(t)$ we have the exact solution

$$u(x,t) = \left\{ \begin{array}{ll} g(t - x/a), & x < at) \\ F(x - at), & x > at \end{array} \right.$$

The solution is constant along each *characteristic line* with slope $dx/dt = a$.

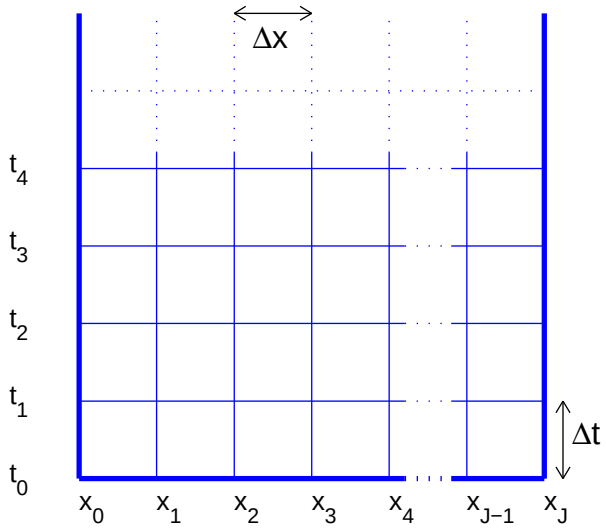


We see that the wave direction (i.e. sign(a)) is important for the analytic solution of the advection equation. We need to consider how this "feeds" into our numerical finite difference schemes. In particular

- Is there a preferred direction in our numerical scheme, in particular is there better stability/convergence if we include some of the underlying "physics of the problem" in our solution?
- In addition to stability we want our scheme to propagate the wave at approximately the correct speed – how do we analyse this in the numerical scheme?

3.2 Some simple numerical schemes for $u_t + au_x = 0$

We use the usual uniform grid in x and t, i.e. fixed values of Δx and Δt.



with $x_j = x_0 + j\Delta x$, $t_n = n\Delta t$, and the approximate solution $u(x_j, t_n) \approx w_j^n$.
Once again we need to approximate derivatives, this time u_t and u_x. We chose to approximate u_t by forward differences

$$u_t \approx \frac{F_t}{\Delta t} w_j^n = \frac{w_j^{n+1} - w_j^n}{\Delta t},$$

since this should lead to an explicit scheme.
There are various possibilities for approximating u_x

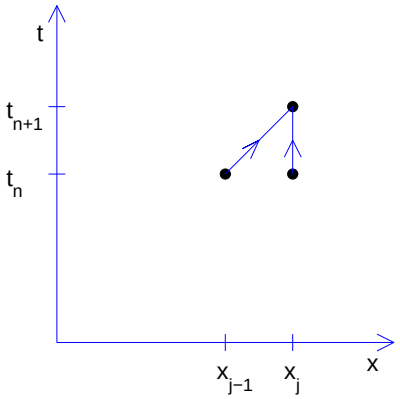
$$u_x|_{(x_j,t_n)} \approx \begin{cases} (w_j^n - w_{j-1}^n)/\Delta x, & \text{backwards diff.} \\ (w_{j+1}^n - w_{j-1}^n)/2\Delta x, & \text{central diff.} \\ (w_{j+1}^n - w_j^n)/\Delta x, & \text{forwards diff.} \end{cases}$$

We first consider a backwards difference approximation for u_x, which will lead to a FTBS scheme. We approximate $u_t + au_x = 0$ by

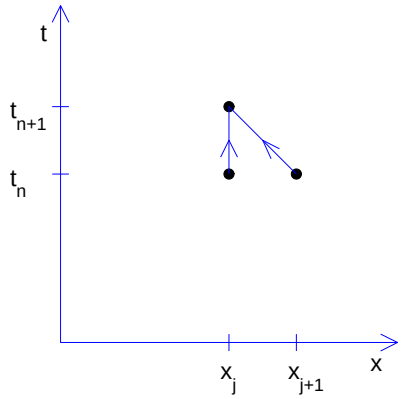
$$\begin{aligned} &\frac{w_j^{n+1} - w_j^n}{\Delta t} + a\frac{w_j^n - w_{j-1}^n}{\Delta x} = 0 \\ &\text{or } w_j^{n+1} = (1-p)w_j^n + pw_{j-1}^n \\ &\text{where } p = \frac{a\Delta t}{\Delta x} = \text{CFL number (Courant-Friedrichs-Lewy, 1928)} \end{aligned} \tag{3.3}$$

Alternatively we can use a forward difference approximation for u_x to get a FTFS scheme

$$w_j^{n+1} = (1+p)w_j^n - pw_{j+1}^n \tag{3.4}$$



FTBS scheme, info "travels" to right (like PDE with $a > 0$?).



FTFS scheme, info "travels" to left (like PDE with $a < 0$?).

The figures suggest that the applicability of the schemes will depend on sign(a). We will try to analyse this by studying the LTE and the stability of the schemes

LTE of FTBS and FTFS schemes

For the FTBS scheme we have

$$L_\Delta w_j^n = \frac{w_j^{n+1} - w_j^n}{\Delta t} + a\frac{w_j^n - w_{j-1}^n}{\Delta x},$$

and the LTE is defined as $L_\Delta u(x_j, t_n)$, so

$$\begin{aligned} \text{LTE} &= \frac{u(x_j, t_{n+1}) - u(x_j, t_n)}{\Delta t} + a\frac{u(x_j, t_n) - u(x_{j-1}, t_n)}{\Delta x} \\ &= \left(u_t + \frac{1}{2}\Delta t u_{tt} + O(\Delta t^2) + a\left[u_x - \frac{1}{2}\Delta x u_{xx} + O(\Delta x^2) \right] \right) \Big|_{(x_j, t_n)} \\ &= \underbrace{(u_t + au_x)}_{=0 \text{ by PDE}} + \underbrace{\frac{1}{2}(\Delta t\, u_{tt} - a\Delta x\, u_{xx})}_{\neq 0 \text{ in general}} + O(\Delta t^2, \Delta x^2), \end{aligned}$$

so the main component of the LTE is the second term which shows the FTBS scheme is in general 1st order accurate in time and space.

$$\text{Note: } u_t + au_x = 0 \Rightarrow u_{tt} = -au_{xt} = -a(u_t)_x = a^2 u_{xx}$$

$$\text{so} \quad \frac{1}{2}\Delta t\, u_{tt} - \frac{1}{2}a\Delta x\, u_{xx} = \frac{1}{2}a[a\Delta t - \Delta x]u_{xx}$$
$$= 0 \quad \text{iff } p = 1 \quad (p = a\Delta t/\Delta x).$$

But if $p = 1$ then (3.3) is $w_j^{n+1} = w_{j-1}^n$, which is *exact* (recall that the exact solution is $u(x,t) = \phi(x - at)$ for some function ϕ). This is reflected in the LTE in that if $p = 1$, the LTE $= 0$.

Exercise: We have shown that only the first order terms in the LTE vanish if $p = 1$ – show that this happens for the higher order terms also.

Exercise: Show that (3.4) is also 1st order accurate, unless $p = -1$ ($\Delta x = -a\Delta t \Rightarrow a < 0$), in which case it is exact.

Hence the LTE of both schemes $\to 0$ as $\Delta x, \Delta t \to 0$, so they are both consistent. The Lax equivalence theorem, Consistency + Stability $\Leftrightarrow$ Convergence, also holds in the hyperbolic case, so we need to determine whether these schemes are stable to prove convergence.

Stability of (3.3)

Set $w_j^n = \xi^n e^{i\omega j}$ in (3.3)

$$\Rightarrow \xi^{n+1} e^{i\omega j} = (1-p)\xi^n e^{i\omega j} + p\xi^n e^{i\omega(j-1)}$$

cancelling $\xi^n e^{i\omega j}$ as usual gives

$$\xi = (1-p) + pe^{-i\omega}.$$

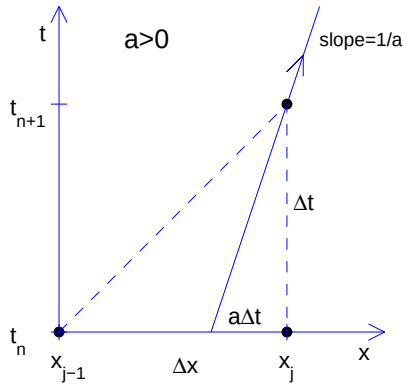
Note that ξ is now complex, a complication we will return to later. We need $|\xi| \le 1$ for von Neumann stability, for all $\omega \in [-\pi, \pi]$. Now $e^{-i\omega} = \cos\omega - i\sin\omega$, so

$$\xi = (1 - p + p\cos\omega) + p(-i\sin\omega)$$
$$\text{so} \quad |\xi|^2 = (1 - p + p\cos\omega)^2 + p^2\sin^2\omega$$
$$= (1-p)^2 + 2(1-p)p\cos\omega + p^2\cos^2\omega + p^2\sin^2\omega$$
$$= 1 - 2p(1-p)(1-\cos\omega)$$
$$= 1 - 4p(1-p)\sin^2(\omega/2)$$

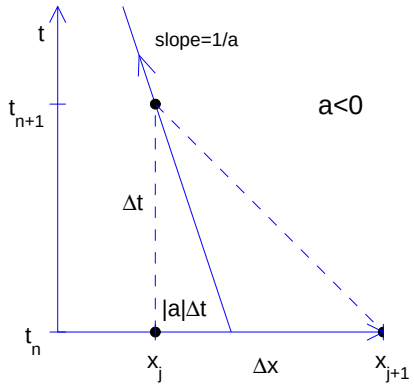
For $|\xi| \le 1$ we need $p(1-p) \ge 0$ which leads to the condition $p \in [0,1]$. Since $p = a\Delta t/\Delta x$, this means (3.3) is stable iff $a > 0$ and $\Delta t \le \Delta x/a$. Note in particular that if $a < 0$, (3.3) is *unstable* for *all* Δt.

Exercise: Show that (3.4) is stable $\Leftrightarrow$ $p \in [-1, 0]$, i.e. (3.4) is stable when $a < 0$ and $\Delta t \le \Delta x/|a|$, and it is unstable if $a > 0$.

Now we have an proper understanding of the ideas we discussed earlier: physically the exact solution travels a distance $a\Delta t$ in time Δt. Meanwhile the numerical scheme is influenced by a distance Δx in time Δt. The stability requirement means that the characteristic line of the exact solution passing through (x_j, t_n) must lie with the "computational molecule" of the numerical solution.

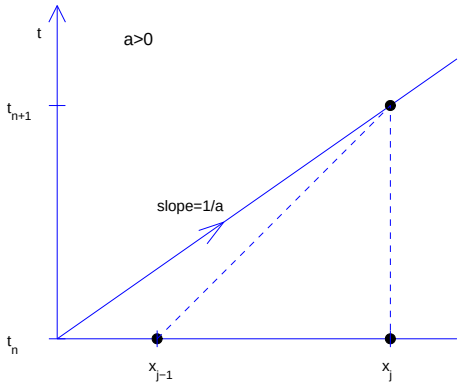


$\Delta x \geq a\Delta t$ for stability.



$\Delta x \geq |a|\Delta t$ for stability.

If $\Delta x < a\Delta t, a > 0$ we have a diagram like the one on the right, with a similar diagram for the case $a < 0$. The characteristic line lies outside the computational molecule and the scheme is not feeding information from the correct range of x



FTCS scheme

What happens if we use a central difference for u_x? We might expect that the central difference will give higher order accuracy, but does the "prefered direction" in hyperbolic problems impose greater restrictions? It is easily checked that the FTCS scheme gives

$$w_j^{n+1} = w_j^n - \frac{1}{2}p(w_{j+1}^n - w_{j-1}^n).$$

If we carry out a stability analysis in the usual way we get

$$\xi = 1 - \frac{1}{2}p(e^{i\omega} - e^{-i\omega}) = 1 - ip\sin(\omega).$$

Hence

$$\begin{aligned} |\xi|^2 &= 1 + p^2\sin^2(\omega) \\ &> 1 \quad \text{for all } \omega \neq 0, \pm\pi \text{ for any } p \neq 0, \end{aligned}$$

so the scheme is *completely unstable*, no matter what value of a and p are used - equally bad for $a > 0$ and $a < 0$.

Summary

If $a > 0$ we can use the FTBS scheme (3.3) and if $a < 0$ we can use the FTFS scheme (3.4). Both these schemes require $|p| \leq 1$ for stability and are first order in space and time. They are

called *upwind schemes* - information in the scheme flows in the same direction as that for the PDE (a *downwind* scheme seeks information in the wrong direction).

Complex ξ and relative phase errors

Why do we get complex values for ξ in these problems? Return to our Fourier approach, and assume we have a Fourier mode $\exp(i\alpha x)$ as initial conditions at time $t = 0$. Then at time t the wave will have travelled a distance x/a (we assume $a > 0$ here), so the solution is

$$u(x,t) = \exp\left(i\alpha(x - at)\right)$$

In discrete form this is

$$u(x_j, t_n) = u_j^n = \exp\left(i\alpha(j\Delta x - an\Delta t)\right) = \exp\left(-i\alpha an\Delta t\right)\exp\left(i\alpha j\Delta x\right) = \xi^n e^{i\omega j}$$

where $\omega = \alpha\Delta x$ and $\xi = \exp\left(-i\alpha a\Delta t\right)$. Hence we *should* see a complex phase shift of $\exp\left(-i\alpha a\Delta t\right)$ in one time step. Ideally we want $|\xi| = 1$.

We can write $\xi = |\xi|e^{i\arg\xi} = |\xi|e^{i\phi}$ say. For stability we require $|\xi| \leq 1$, and to get the wave travelling at the correct speed we require $\phi \approx -\alpha a\Delta t = -\omega a\Delta t/\Delta x = -\omega p$, as $\Delta x, \Delta t \to 0$.

Lets see how this works for the FTBS scheme (3.3). We have $\xi = (1 - p + p\cos\omega) + i(-p\sin\omega)$. So

$$\begin{aligned}
\phi &= \tan^{-1}\left(\frac{-p\sin\omega}{1 - p + p\cos\omega}\right) \\
&= -\tan^{-1}\left(\frac{p\sin\omega}{1 - p + p\cos\omega}\right) \\
&\approx -p\omega\left[1 - \frac{1}{6}(1-p)(1-2p)\omega^2 + \ldots\right]
\end{aligned}$$

where we have used the expansions $\sin x = x - x^3/3! + \ldots$, $\cos x = 1 - x^2/2! + \ldots$, and $\tan^{-1} x = x - x^3/3 + \ldots$.

Exercise: Check the details of this result.

Note in expanding for small ω, we are taking into account that large frequencies cannot be represented accurately on the finite difference mesh.

The ratio between the exact value $\phi = -p\omega$ and the value given above is gives us the *relative phase error*. More precisely, the relative phase error is obtained by subtracting the exact result 1 from the term multiplying $-p\omega$, so in the example above the relative phase error is

$$-\frac{1}{6}(1-p)(1-2p)\omega^2 + \ldots,$$

and hence of order $O(\omega^2)$. When $p = 1$ or $p = \frac{1}{2}$ the error disappears at this order, otherwise the sign will depend on the size of p. When $p = 1$ we would expect the error to vanish since the scheme is exact at this value of p.

3.2.1 Test problems for the advection equation

We now examine the result of applying the FTBS scheme (3.3) to two test problems, the advection equation (3.1) with two different sets of initial conditions. Matlab programs for these examples are available from the course web page.

(1) The first test problem (the upper plot in the figures) uses a Gaussian pulse as initial condition:

$$u(x,0) = \exp\left[-400(x-0.15)^2\right].$$

The figures show the initial "spike" $u(x,0)$ on the left (solid line), and the exact solution $u(x,t)$ of the PDE (3.1) when $t = 0.5$ (r.h.spike, solid line). The solution of the approximation scheme at time $t = 0.5$ is shown as a dashed line.

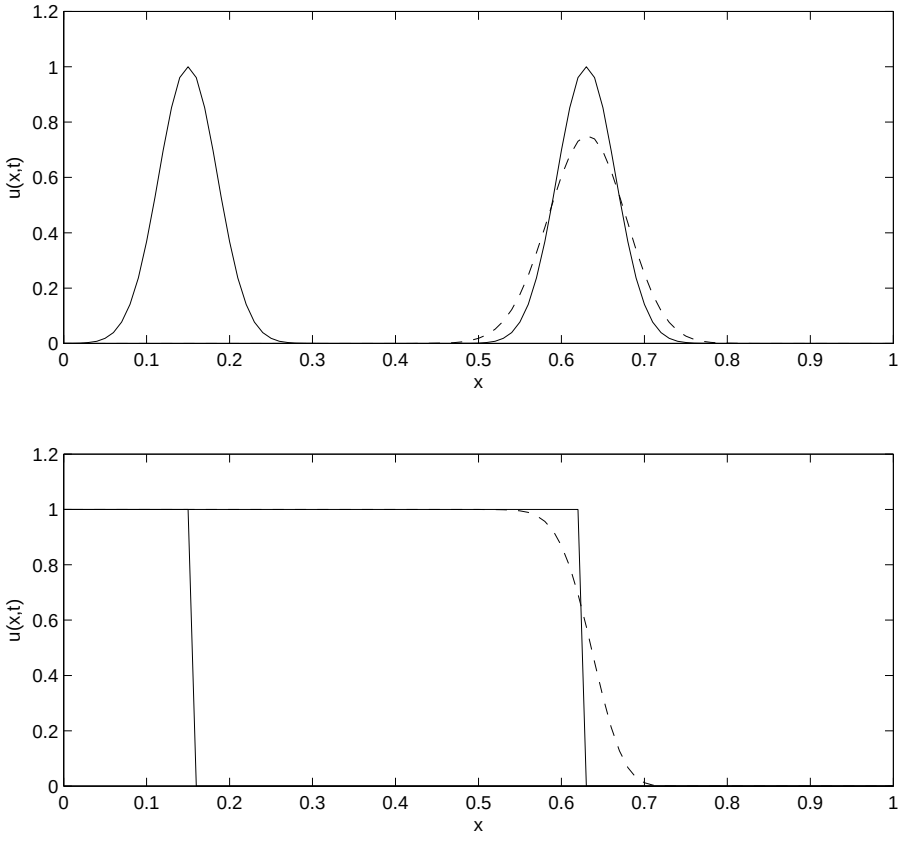
(2) The second test problem (the lower plot in the figure) uses a step function as initial condition:

$$u(x,0) = \begin{cases} 1 & \text{for } 0 \le x \le 0.15 \\ 0 & \text{for } 0.15 < x \le 1. \end{cases}$$

For this problem the figures show the exact solution $u(x,t)$ of (3.1) when $t = 0.5$ as a solid line, and the solution of the approximation scheme at time $t = 0.5$ as a dashed line.

Note: the closer the dashed and solid lines are together, the better the scheme is.

Numerical details: In all cases $J = 100$ and $p = 0.8$, so $\Delta x = 0.01$ and $\Delta t = 0.008$ (recall that we are using $a = 1$).



We see that the spike solution loses amplitude and spreads out, but its position looks accurate. The step also smears out but appears to be in the correct position. An analysis of the LTE explains the spreading. Using $u_{tt} = a^2 u_{xx}$ we have

$$L_\Delta = \underbrace{u_t + a u_x}_{\text{PDE}} + \underbrace{\frac{a}{2}(p-1)\Delta x\, u_{xx}}_{\text{LTE leading term}} + O(\Delta t^2, \Delta x^2),$$

Another way of looking at this is that we are using the scheme to model the equation

$$u_t + a u_x - \alpha u_{xx} = 0,$$

where $\alpha = \frac{a}{2}(1-p)dx$. In this case the $O(\Delta t^2, \Delta x^2)$ would be its local truncation error. The extra term is a *diffusion term* and historically this is called *artificial viscosity*. A small amount of artificial viscosity is usually necessary for a scheme, but too much smears the solution out. Alternatively we see from the stability analysis that

$$|\xi|^2 = 1 - 4p(1-p)\sin^2(\omega/2) \approx 1 - p(1-p)\omega^2,$$

so the ω^2 term is damping too much.

3.3 Leapfrog scheme for the advection equation

In this scheme, also called the CTCS scheme, we use central differences for both time and space

$$\begin{aligned} &\frac{w_j^{n+1} - w_j^{n-1}}{2\Delta t} + a\frac{w_{j+1}^n - w_{j-1}^n}{2\Delta x} = 0 \\ &\Rightarrow w_j^{n+1} = w_j^{n-1} - p(w_{j+1}^n - w_{j-1}^n), \end{aligned} \tag{3.5}$$

where $p = a\Delta t/\Delta x$ as usual. This scheme is also explicit but involves three time levels and so it needs *two* time levels to start. The IC gives $\mathbf{w}^0$ and we need to obtain $\mathbf{w}^1$ using another scheme, such as the (unstable) FTCS scheme.

Exercise: Show that the LTE of the leapfrog scheme (3.5) is

$$\begin{aligned} \text{LTE} &= \frac{1}{6}\left[\Delta t^2 u_{ttt} + a\Delta x^2 u_{xxx}\right] + O(\Delta t^4, \Delta x^4) \\ &= \frac{a}{6}(1-p^2)u_{xxx}\Delta x^2 + O(\Delta t^4, \Delta x^4), \quad \text{using } u_{ttt} = -a^3 u_{xxx}, \end{aligned}$$

i.e. (3.5) is *2nd order accurate*.

Now consider stability. Set $w_j^n = \xi^n e^{i\omega j}$ as usual to get after some simplification

$$\begin{aligned} \xi^2 &= 1 - p\xi\left(e^{i\omega} - e^{-i\omega}\right) \\ \text{or} \quad \xi^2 &+ 2ip\xi\sin\omega - 1 = 0. \end{aligned}$$

This has roots

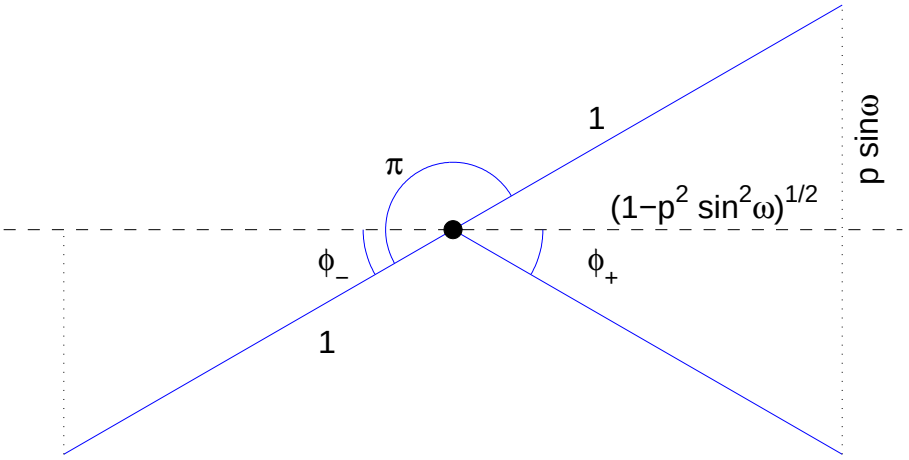
$$\xi_\pm = -ip\sin\omega \pm \sqrt{1 - p^2\sin^2\omega}.$$

We need $|\xi_\pm| \leq 1$ for stability. If $|p| > 1$ then when $\omega = \frac{\pi}{2}$, we have $\xi_\pm = -ip \pm i\sqrt{p^2-1} = -i\left[p \mp \sqrt{p^2-1}\right]$ and so one of the roots is > 1 in modulus, i.e. unstable if $|p| > 1$.

If $|p| \leq 1$ then $\sqrt{1-p^2\sin^2\omega}$ is real for all ω and so

$$|\xi_\pm|^2 = (-p\sin\omega)^2 + (1-p^2\sin^2\omega) = 1 \quad \forall\omega,$$

i.e. (3.5) is stable for all $|p| \leq 1$ ($-1 \leq p \leq 1$) and it works equally well for $a > 0$ and $a < 0$. There is no damping in this case since $|\xi_\pm|^2 = 1$. What about the phase error? There are two roots in this case and we need to consider each in turn.



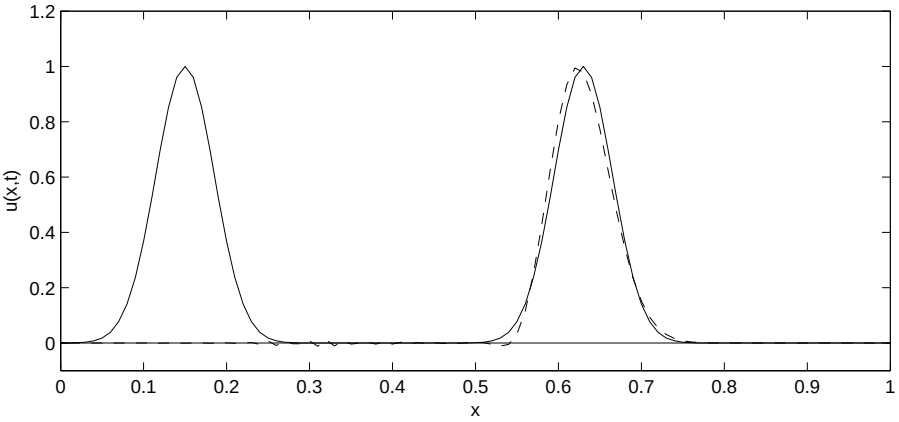
$$\begin{aligned}
\xi_+ : \quad \phi_+ &= -\sin^{-1}(p\sin\omega) \\
&= -\sin^{-1}(p\omega - p\omega^3/3! + \dots) \\
&= -p\omega(1 - \frac{1}{6}(1-p^2)\omega^2 + \dots) \\
&\quad (\text{using } \sin^{-1}(x) = x + x^3/6 + \dots)
\end{aligned}$$

This part is similar to upwind schemes, although note now the phase error is always of one sign. This seems fine, now consider the other root

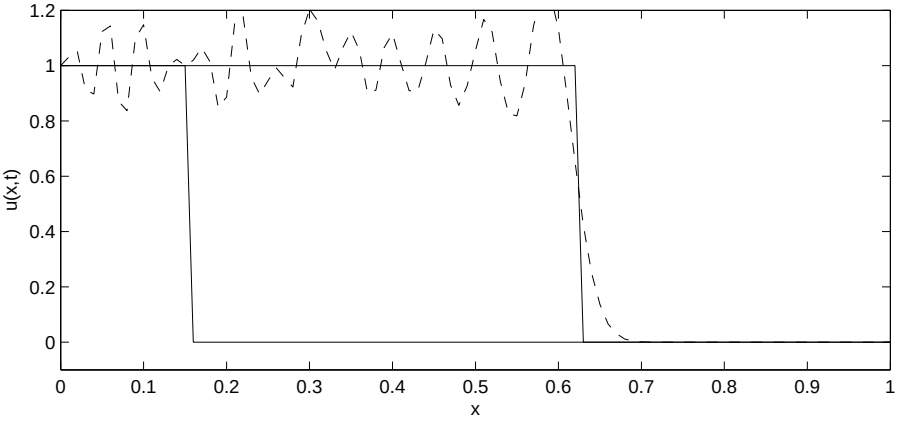
$$\begin{aligned}
\xi_- : \quad \phi_- &= \pi + \sin^{-1}(p\sin\omega) \\
&= p\omega + \pi - \frac{1}{6}p\omega^3(1-p^2) + \dots
\end{aligned}$$

This is a mode oscillating from time step to time step and travelling in the wrong direction, a mode absent in the exact solution of the equation. Suppressing this mode is necessary, some form of "filtering" will be required if this scheme is to be useful in practice.

The Leapfrog scheme applied to the test problems



The spike is handles pretty well by the scheme, but note some small amplitude oscillations to the left of the pulse (the amplitude of these increases if $\Delta x, \Delta t$ are made bigger).



The step front is reproduced quite well (steeper than in the upwind case, which is good), but there are large oscillations (bad). The step function excites all Fourier modes and stimulates the backwards travelling wave. Since $|\xi| = 1$ in the scheme, there is no damping so spurious oscillations persist for long periods. The leading term of the LTE is *dispersive* (u_{xxx}) and not *diffusive* (u_{xx}). What we need is a scheme that has a little artificial viscosity (but not too much) in order to damp out the oscillations. We would also like to have a 2nd order accurate method. The **Lax-Wendroff** scheme has these properties.

3.3.1 The Lax-Wendroff scheme

This is a modification of the (unstable) FTCS scheme

$$w_j^{n+1} = w_j^n - \frac{1}{2}p(w_{j+1}^n - w_{j-1}^n).$$

Suppose u is a smooth solution of $u_t + au_x = 0$. Then as we have seen,

$$\left(\frac{\partial}{\partial t}\right)^m u = \left(-a\frac{\partial}{\partial x}\right)^m u \quad \forall m$$

Now

$$\begin{aligned} u(x, t+\Delta t) &= u + \Delta t u_t + \frac{1}{2}\Delta t^2 u_{tt} + O(\Delta t^3)\Big|_{(x,t)}, \\ &= u - a\Delta t u_x + \frac{1}{2}a^2\Delta t^2 u_{xx} + O(\Delta t^3)\Big|_{(x,t)}, \\ &\approx u - a\Delta t\frac{D_x}{2\Delta x}u + \frac{1}{2}a^2\Delta t^2\frac{\delta_x^2}{\Delta x^2}u, \end{aligned}$$

where the final line results from truncating the expansion and replacing derivatives by their central difference approximations. This suggests the following explicit scheme

$$w_j^{n+1} = \underbrace{w_j^n - \frac{p}{2}(w_{j+1}^n - w_{j-1}^n)}_{\text{FTCS scheme}} + \underbrace{\frac{p^2}{2}(w_{j+1}^n - 2w_j^n + w_{j-1}^n)}_{\text{extra term}}$$

rearranging gives

$$w_j^{n+1} = (1-p^2)w_j^n - \frac{1}{2}p(1-p)w_{j+1}^n + \frac{1}{2}p(1+p)w_{j-1}^n, \tag{3.6}$$

the *Lax-Wendroff Scheme* (L-W).
Consider now the LTE of the L-W scheme. We can write the scheme in the form

$$L_\Delta w_j^n = \frac{w_j^{n+1} - w_j^n}{\Delta t} + a\frac{D_x}{2\Delta x}w_j^n - \frac{1}{2}a^2\Delta t\frac{\delta_x^2}{\Delta x^2}w_j^n,$$

which most closely mirrors the PDE $u_t + au_x = 0$. By definition

$$\begin{aligned} \text{LTE } &= L_\Delta u(x_j, t_n) = \frac{u(x_j, t_{n+1}) - u(x_j, t_n)}{\Delta t} + a\frac{D_x}{2\Delta x}u(x_j, t_n) - \frac{1}{2}a^2\Delta t\frac{\delta_x^2}{\Delta x^2}u(x_j, t_n) \\ &= \underbrace{\frac{1}{2}\Delta t[u_{tt} - a^2 u_{xx}]}_{=0} + \frac{1}{6}\Delta t^2 u_{ttt} + \frac{a}{6}\Delta x^2 u_{xxx} + O(\Delta t^3, \Delta x^4, \Delta t\Delta x^2) \\ &= (1-p^2)\frac{a}{6}\Delta x^2 u_{xxx} + O(\Delta x^3). \end{aligned}$$

Hence the method is 2nd order accurate.

We now look at stability. Inserting $w_j^n = \xi^n e^{i\omega j}$ into (3.6) and simplifying, we get

$$\begin{aligned}\xi &= (1-p^2) - \frac{1}{2}p(1-p)e^{i\omega} + \frac{1}{2}p(1+p)e^{-i\omega} \\ &= 1 + p^2(\cos\omega - 1) - ip\sin\omega \\ &= 1 - 2p^2\sin^2(\omega/2) - ip\sin\omega \\ &= 1 - 2p^2\sin^2(\omega/2) - 2ip\sin(\omega/2)\cos(\omega/2)\end{aligned}$$

So

$$\begin{aligned}|\xi|^2 &= \left[1 - 2p^2s^2\right]^2 + 4p^2s^2c^2, \text{ where } s = \sin(\omega/2), c = \cos(\omega/2) \\ &= 1 + 4p^2s^2(c^2-1) + 4p^4s^4 \\ &= 1 - 4p^2(1-p^2)s^4\end{aligned}$$

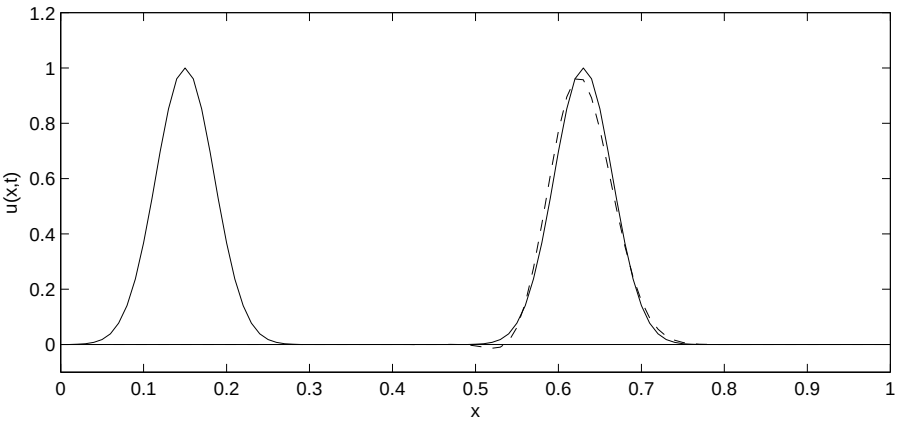
Clearly this is ≤ 1 for all $|p| \leq 1$ and > 1 for all $|p| > 1$, so the scheme is stable if and only if $|p| \leq 1$. We also see that $|\xi|^2 \approx 1 - p^2(1-p^2)\,\omega^4/4 + \ldots$ for small ω, so there is some damping but substantially less than the upwind schemes.

What about the phase error? We have

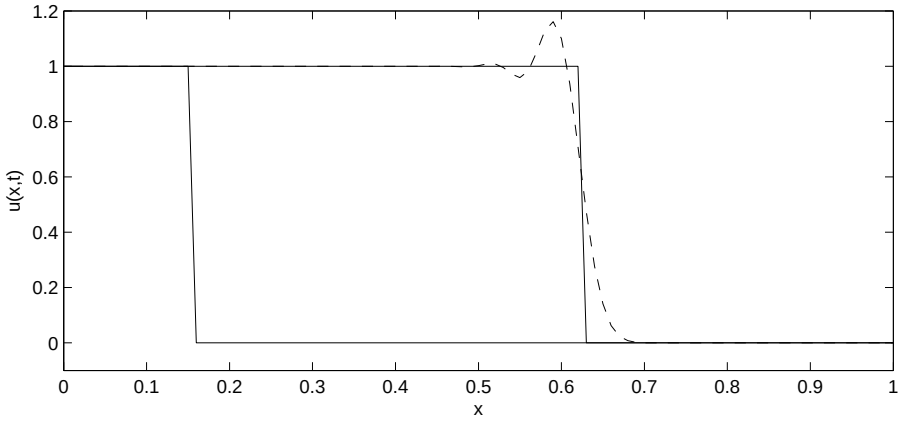
$$\begin{aligned}\phi &= -\tan^{-1}\left[\frac{p\sin\omega}{1-2p^2\sin^2(\omega/2)}\right] \\ &= -\tan^{-1}\left[p(\omega - \frac{1}{6}\omega^3 + \ldots)(1 + p^2\omega^2/2 + \ldots)\right] \\ &= -\tan^{-1}\left[p\omega(1 + \omega^2(\frac{1}{2}p^2 - \frac{1}{6}) + \ldots)\right] \\ &= -p\omega\left(1 - \frac{1}{6}\omega^2(1-p^2) + \ldots\right).\end{aligned}$$

This is the same as the first root of the Leapfrog scheme, a phase error always of one sign (lag), but the second troublesome root of the Leapfrog scheme is now absent.

The Lax-Wendroff scheme applied to the test problems



The spike is handled well by the scheme.



Less smeared out than the upwind scheme, fewer oscillations than the leapfrog scheme - most are damped.

The Lax-Wendroff is a well-used method.

All the methods we have seen so far are explicit. It is worth looking at the Crank-Nicholson scheme to see if an implicit scheme would be any better.

Backward time schemes

Backward time schemes can be constructed in a similar way to Forward time schemes. For example the BTCS scheme is

$$\frac{w_j^{n+1} - w_j^n}{\Delta t} + a\frac{w_{j-1}^{n+1} - w_{j+1}^{n+1}}{2\Delta x} = 0$$

or

$$w_j^{n+1} + \frac{1}{2}p(w_{j+1}^{n+1} - w_{j-1}^{n+1}) = w_j^n$$

Exercise Show the BTCS scheme is first order in the LTE. Show that the scheme is stable for all p.

By taking the average of the FTCS scheme and the BTCS scheme we get the following:

Crank-Nicholson scheme for the advection equation

The PDE is $u_t = -au_x$. Approximate the spatial part of this by an average across time levels n and $n+1$.

$$\begin{aligned}\frac{w_j^{n+1} - w_j^n}{\Delta t} &\approx -\frac{a}{2}\left(u_x\,|_{t=t_n} + u_x|_{t=t_{n+1}}\right) \\ &= -\frac{a}{2}\left(\frac{D_x}{2\Delta x}w_j^n + \frac{D_x}{2\Delta x}w_j^{n+1}\right),\end{aligned}$$

using central differences in space to approximate u_x. If we rearrange all this we get the CN scheme for the advection equation.

$$w_j^{n+1} + \frac{p}{4}\left(w_{j+1}^{n+1} - w_{j-1}^{n+1}\right) = w_j^n - \frac{p}{4}\left(w_{j+1}^n - w_{j-1}^n\right) \tag{3.7}$$

As in the parabolic case, we need to solve a tridiagonal system of equations to get the solution at each timestep.

As usual, with any new scheme, we need to consider the LTE, the stability, and the phase error. First the LTE. Standard calculations show that

$$\begin{aligned}\text{LTE} &= \underbrace{u_t + au_x}_{=0} + \underbrace{\frac{\Delta t}{2}(u_{tt} + au_{xt})}_{=0} + \frac{\Delta t^2}{6}u_{ttt} + \frac{a\Delta x^2}{6}u_{xxx} + \frac{a\Delta t^2}{4}u_{xtt} + \text{ h.o.t.} \\ &= \frac{a\Delta x^2}{6}u_{xxx}(1 + \frac{1}{2}p^2) + O(\Delta x^3)\end{aligned}$$

Now stability - inserting $w_j^n = \xi^n e^{i\omega j}$ into (3.7) and simplifying, we get

$$\begin{aligned}\xi\left[1 + \underbrace{\frac{p}{4}(e^{i\omega} - e^{-i\omega})}_{\frac{ip}{2}\sin\omega}\right] &= 1 - \frac{p}{4}(e^{i\omega} - e^{-i\omega}) \\ \Rightarrow \xi &= \frac{1 - \frac{1}{2}ip\sin\omega}{1 + \frac{1}{2}ip\sin\omega} = \frac{1 - \frac{p^2}{4}\sin^2\omega - ip\sin\omega}{1 + \frac{p^2}{4}\sin^2\omega} \\ \Rightarrow |\xi|^2 &= \frac{|1 - \frac{1}{2}ip\sin\omega|^2}{|1 + \frac{1}{2}ip\sin\omega|^2} = \frac{1 + \frac{1}{4}p^2\sin^2\omega}{1 + \frac{1}{4}p^2\sin^2\omega} = 1 \quad \forall p, \omega.\end{aligned}$$

So the scheme is stable for all p (all the other schemes we have looked at are unstable for $p > 1$). However it contains no artificial viscosity since $|\xi|^2 = 1$.

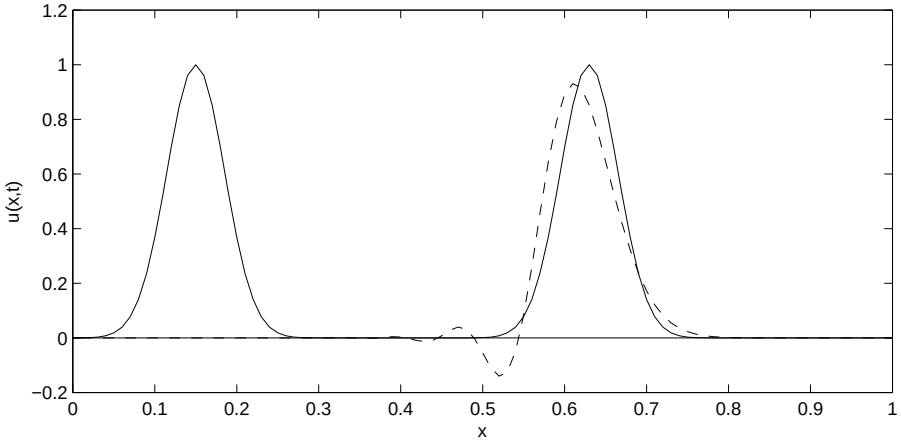
Now phase error. We have

$$\begin{aligned}\phi &= -\tan^{-1}\left[\frac{p\sin\omega}{1 - \frac{p^2}{4}\sin^2(\omega)}\right] \\ &= -\tan^{-1}\left[p\omega(1 + \omega^2\left(\frac{p^2}{4} - \frac{1}{6}\right) + \dots)\right] \\ &= -p\omega\left(1 - \frac{1}{6}\omega^2(1 + \frac{1}{2}p^2) + \dots\right).\end{aligned}$$

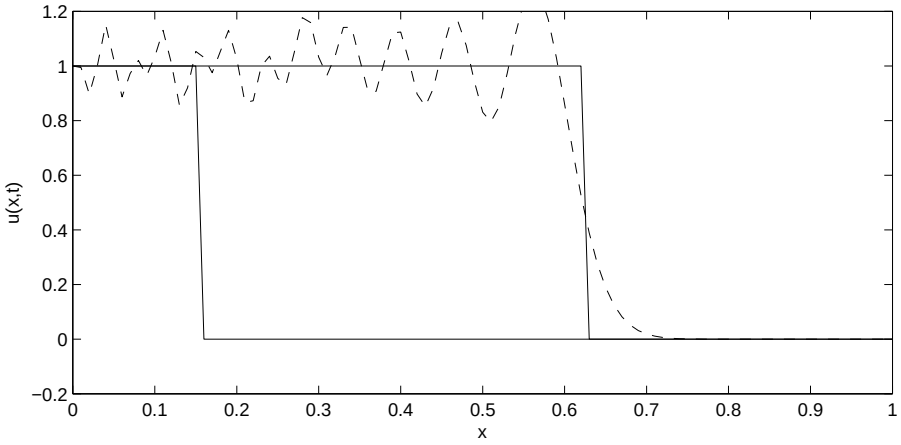
So although we can use a large $|p|$ from the stability result, the phase errors will grow when $|p| > 1$, so the extra stability range is not so useful.

Let's see how it works on the test problems.

The Crank-Nicholson scheme applied to the test problems



The spike is not resolved very well by the scheme and there are some oscillations.



The step conditions induce bad oscillations. The numerical evidence suggests that the lack of artificial viscosity in this scheme means that undamped oscillations are a serious problem.

We can summarise our results in a table.

Scheme	accuracy	stable	$p = \pm 1$	Comments
Upwind	1st	$\|p\| \leq 1$	exact	Need to use (3.3) for $a > 0$, (3.4) for $a < 0$. Solutions smear out too much.
Leapfrog	2nd	$\|p\| \leq 1$	exact	Multi-level. Bad oscillations.
Lax-Wendroff	2nd	$\|p\| \leq 1$	exact	Best solution
C-N	2nd	$\forall p$	Not exact	Implicit. Bad oscillations

3.4 2nd order equations

The prototype for 2nd order equations is the wave equation. In one spatial dimension this is

$$u_{tt} = a^2 u_{xx}, \tag{3.8}$$

and in 3D, for example, we have

$$u_{tt} = a^2(u_{xx} + u_{yy} + u_{zz}).$$

This models, for example, the transmission of sound waves in air or water. The parameter a is the speed of the wave, as in the advection equation. The appropriate initial conditions are to give *both* $u(x,0)$ and $u_t(x,0)$. In finite regions we would specify u or u_x at both l..h. and r.h. boundaries.

We consider now the 1D case. There is an exact solution due to D'Alembert

$$u(x,t) = F(x-at) + G(x+at)$$

for arbitrary functions F and G. The simplest numerical approximation is to use central differences in both space and time

$$\begin{aligned} \frac{w_j^{n+1} - 2w_j^n + w_j^{n-1}}{\Delta t^2} &= \frac{w_{j-1}^n - 2w_j^n + w_{j+1}^n}{\Delta x^2} \\ \Rightarrow w_j^{n+1} &= 2w_j^n - w_j^{n-1} + p^2(w_{j-1}^n - 2w_j^n + w_{j+1}^n), \quad p = a\Delta t/\Delta x. \end{aligned} \tag{3.9}$$

Since this is a 3-level scheme, we need some way of getting the $n = 1$ time level when we start. We can do this by a form of the fictitious point scheme, this time an extra set of grid points at $t = -\Delta t$. Suppose we have at $t = 0$ ($n = 0$) that

$$\begin{aligned} u(x,0) = f(x) \quad &\Rightarrow \qquad w_j^0 = f_i \\ u_t(x,0) = g(x) \quad &\Rightarrow \qquad \frac{w_j^1 - w_j^{-1}}{2\Delta t} = g_i, \end{aligned}$$

using central differences for the time derivative. Now write down the scheme (3.9) at $n = 0$ and use the above equations to give the w_j^0 and to eliminate the ficitious point at $n = -1$.

$$\begin{aligned} w_j^1 &= 2w_j^0 - w_j^{-1} + p^2(w_{j-1}^0 - 2w_j^0 + w_{j+1}^0) \\ \Rightarrow w_j^1 &= 2f_j - w_j^{-1} + p^2(f_{j-1} - 2f_j + f_{j+1}) \\ \Rightarrow w_j^1 &= 2f_j - w_j^1 + 2\Delta t g_i + p^2(f_{j-1} - 2f_j + f_{j+1}) \\ \Rightarrow w_j^1 &= f_j + \Delta t g_i + \frac{1}{2}p^2(f_{j-1} - 2f_j + f_{j+1}), \quad j = 1, 2, \ldots, J. \end{aligned}$$

This last equation gives the solution at the $n = 1$ time level from the known initial functions.

Exercise: show that the LTE of scheme (3.9) is

$$\begin{aligned} \text{LTE} &= \frac{1}{12}(\Delta t^2 u_{tttt} - a^2\Delta x^2 u_{xxxx}) + O(\Delta t^4, \Delta x^4) \\ &= \frac{a^2}{12}(p^2 - 1)\Delta x^2 u_{xxxx} + O(\Delta x^4), \end{aligned}$$

i.e. it is 2nd order accurate.

3.4.1 Stability of scheme for wave Equation.

Now consider stability of (3.9). Set $w_j^n = \xi^n e^{i\omega j}$ as usual to get after some simplification

$$\begin{aligned}\xi^2 &= 2\xi - 1 + p^2\xi \underbrace{\left(e^{i\omega} - 2 + e^{-i\omega}\right)}_{=-4\sin^2(\omega/2)} \\ &= 2\xi[1 - 2p^2\sin^2(\omega/2)] - 1.\end{aligned}$$

i.e. ξ satisfies the quadratic

$$\xi^2 - 2[1 - 2p^2\sin^2(\omega/2)] + 1 = 0.$$

This has roots

$$\begin{aligned}\xi_\pm &= 1 - 2p^2\sin^2(\omega/2) \pm \sqrt{(1 - 2p^2\sin^2(\omega/2))^2 - 1} \\ &= 1 - 2p^2\sin^2(\omega/2) \pm \sqrt{4p^2\sin^2(\omega/2)[p^2\sin^2(\omega/2) - 1]}.\end{aligned}$$

Since $4p^2\sin^2(\omega/2) > 0$, the expression under the square root is ≤ 0 if $p^2 \leq 1$, and in this case the roots are complex and have the form

$$\begin{aligned}\xi_\pm &= 1 - 2p^2\sin^2(\omega/2) \pm i2p\sin(\omega/2)\sqrt{1 - p^2\sin^2(\omega/2)} \\ \Rightarrow |\xi_\pm|^2 &= [1 - 2p^2\sin^2(\omega/2)]^2 + 4p^2\sin^2(\omega/2)[1 - p^2\sin^2(\omega/2)] \\ &= 1 \quad \forall\omega.\end{aligned}$$

So if $p^2 \leq 1$ then (3.9) is *stable*. Now consider the $p^2 > 1$ case. Put $\omega = \pi$, say, to get

$$\begin{aligned}\xi_\pm &= 1 - 2p^2 \pm 2p\sqrt{p^2 - 1} \\ \text{So } \xi_- &< 1 - 2p^2 < -1 \text{ for } p^2 > 1 \\ \Rightarrow |\xi_-| &> 1 \text{ at } \omega = \pi,\end{aligned}$$

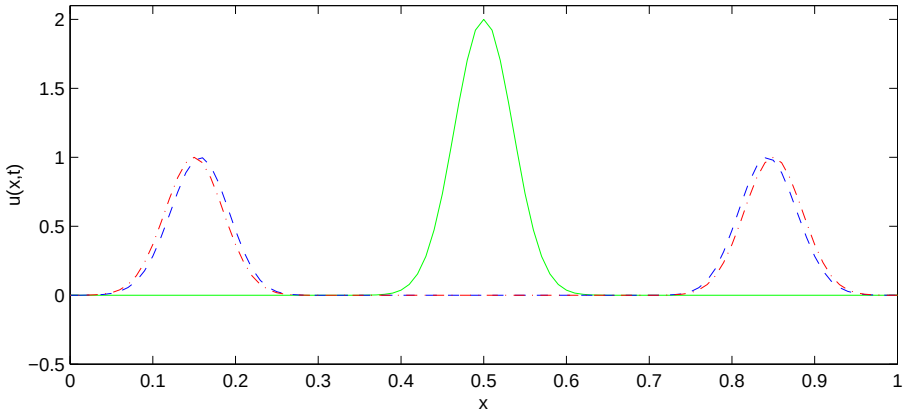
and so the scheme is *unstable* for $p^2 > 1$. To summarise, (3.9) is von Neumann stable $\Leftrightarrow p^2 \leq 1$. Now consider the phase error. We have (after some algebra)

$$\phi_\pm = \pm p\omega\left(1 - \frac{\omega^2}{24}(1 - p^2) + \dots\right).$$

This time we expect two solutions since the equation has waves travelling in both directions.

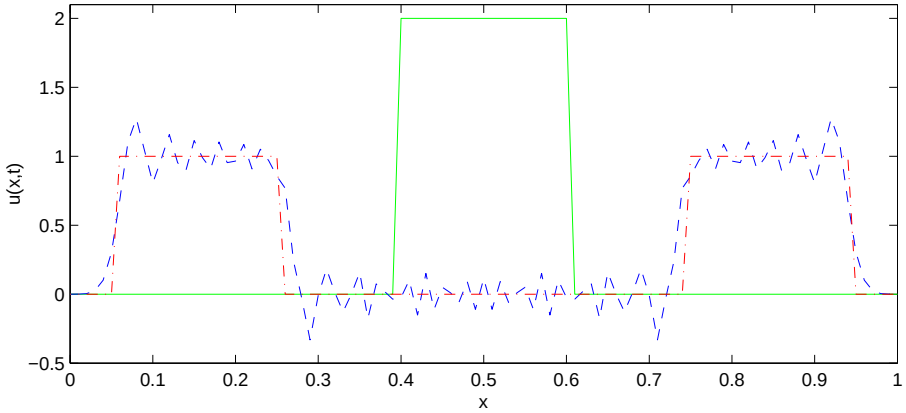
3.4.2 Test problems

This time we generalise our test problems to initial data which results in two pulses travelling in opposite directions. The first picture show the case where the two pulses are Gaussian shaped.



The initial pulse in the centre splits into two smaller pulses travelling to the left and to the right. The dashed-dotted line is the exact solution, the dashed line the numerical one. Agreement is quite good.

If we replace the gaussian pulses by square pulses we see the figure shown below.



This time we get quite strong oscillations, as in the corresponding leapfrog scheme for the advection equation. In this case the problem is that we have no damping since $|\xi| = 1$ and hence the oscillations induced by the sharp step do not die away.

3.4.3 Coupled equations

In practice it is common to write the 2nd order wave equation as a pair of coupled first order equations

$$u_t + a v_x = 0$$
$$v_t + a u_x = 0.$$

It is easy to check that if we differentiate the first equation with respect to t, the second with respect to x, then eliminate v_{xt}, we recover (3.8). Usually when the equation is given in this form we have initial conditions $u(x, 0)$ and $v(x, 0)$ given from physical grounds.

We can write the above in vector form

$$\mathbf{u}_t + A\mathbf{u}_x = 0, \quad \text{where} \quad \mathbf{u} = \begin{pmatrix} u \\ v \end{pmatrix}, \quad A = \begin{pmatrix} 0 & a \\ a & 0 \end{pmatrix}$$

In this form it is easy to apply the Lax-Wendroff scheme. As before we have

$$\left(\frac{\partial}{\partial t}\right)^m \mathbf{u} = \left(-A\frac{\partial}{\partial x}\right)^m \mathbf{u} \quad \forall m$$

So, again following the scalar case

$$\begin{aligned}\mathbf{u}(x,t+\Delta t) &= \mathbf{u} + \Delta t\mathbf{u}_t + \frac{1}{2}\Delta t^2\mathbf{u}_{tt} + O(\Delta t^3)\Big|_{(x,t)}, \\ &= \mathbf{u} - A\Delta t\mathbf{u}_x + \frac{1}{2}A^2\Delta t^2\mathbf{u}_{xx} + O(\Delta t^3)\Big|_{(x,t)}, \\ &\approx \mathbf{u} - A\Delta t\frac{D_x}{2\Delta x}\mathbf{u} + \frac{1}{2}A^2\Delta t^2\frac{\delta_x^2}{\Delta x^2}\mathbf{u},\end{aligned}$$

This suggests the following explicit scheme

$$\mathbf{w}_j^{n+1} = \mathbf{w}_j^n - \frac{1}{2}P(\mathbf{w}_{j+1}^n - \mathbf{w}_{j-1}^n) + \frac{1}{2}P^2(\mathbf{w}_{j+1}^n - 2\mathbf{w}_j^n + \mathbf{w}_{j-1}^n)$$

where

$$\mathbf{w} = \begin{pmatrix} w \\ z \end{pmatrix} \approx \begin{pmatrix} u \\ v \end{pmatrix}, \quad P = \frac{\Delta t}{\Delta x}A.$$

rearranging gives

$$\mathbf{w}_j^{n+1} = (I - P^2)\mathbf{w}_j^n - \frac{1}{2}P(I-P)\mathbf{w}_{j+1}^n + \frac{1}{2}P(I+P)\mathbf{w}_{j-1}^n, \tag{3.10}$$

where I is the 2×2 unit matrix.

3.5 Nonlinear Conservation Laws

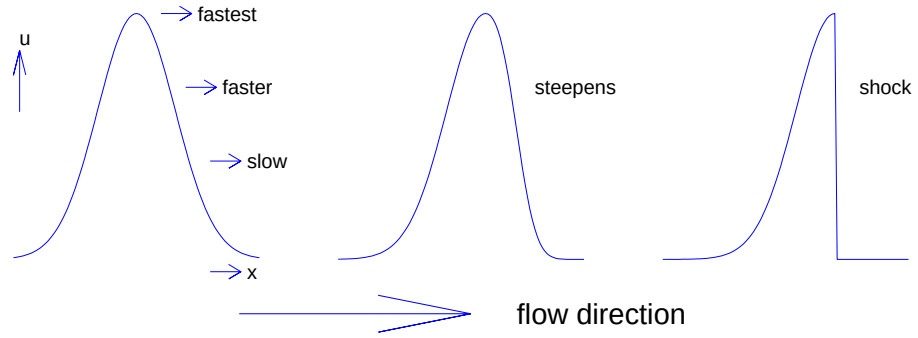
All the example hyperbolic problems we have seen so far are *linear*, i.e. we have no terms where u and its derivatives are multipled together or are the arguments of nonlinear functions. In real life, for example fluid and gas flow problems, the equations are *nonlinear*, for example

$$u_t + a(u)u_x = 0$$

now the wave speed a is a function of the solution $u(x,t)$. The simplest example is when $a(u) = u$, so the equation becomes *Burger's equation*

$$u_t + u\,u_x = 0.$$

Roughly speaking, the speed of the wave is now greater if its amplitude is bigger, so if we have a pulse travelling to the right, its "top" moves faster than the tail on either side, and eventually overtakes the tail, at which point a shock (discontinuity) forms



In this case $a(u) > 0$ so the flow is in the +ve x direction. If $a(u) < 0$ the steepening is in the $-$ve x direction.

Direct approximation of $u_t + a(u)u_x = 0$ often doesn't give sensible or stable results, but things work better if we write the equation as a *conservation law*

$$u_t + a(u)u_x = 0 \overset{\text{rearrange}}{\rightarrow} u_t + \frac{\partial}{\partial x}\left(F(u)\right) = 0 \tag{3.11}$$

where

$$\frac{d}{du}F(u) = a(u), \quad \text{since} \quad \frac{\partial F}{\partial x} = \frac{dF}{du}\frac{\partial u}{\partial x}$$

In the case where $a(u) = u$, we have $F'(u) = u$, so $F(u) = \frac{1}{2}u^2$ and the conservation form of Burgers equation is

$$u_t + \frac{1}{2}\frac{\partial}{\partial x}(u^2) = 0.$$

3.6 Nonlinear Lax Wendroff scheme

We now describe how to extend the Lax-Wendroff scheme to approximate the nonlinear conservation law equation

$$u_t + [F(u)]_x = 0 \tag{3.12}$$

We use a similar procedure to that for the advection equation $u_t + au_x = 0$. Suppose $u(x,t)$ is a smooth function (this will only be true up to the point where a shock forms). We write a truncated Taylor series for $u(x, t + \Delta t)$

$$u(x, t + \Delta t) \approx \left[u + \Delta t\, u_t + \frac{1}{2}\Delta t^2 u_{tt}\right]_{(x,t)}$$

and then replace the time derivatives by space derivatives, using (3.12)

$$\begin{aligned} u_t &= -[F(u)]_x \\ u_{tt} &= \frac{\partial}{\partial t}\left(-[F(u)]_x\right) = -\frac{\partial}{\partial x}\left([F(u)]_t\right) = -\frac{\partial}{\partial x}\left(F'(u)\cdot u_t\right) \\ &= -\frac{\partial}{\partial x}\left(F'(u)[-F(u)]_x\right) = \frac{\partial}{\partial x}\left(F'(u)\frac{\partial}{\partial x}F(u)\right) \end{aligned}$$

For simplicity define

$$Q(u) = F'(u)\frac{\partial}{\partial x}F(u), \text{ so that } u_{tt} = \frac{\partial}{\partial x}Q(u).$$

to get

$$u(x_j, t_{n+1}) \approx \left[u - \Delta t \frac{\partial}{\partial x} F(u) + \frac{1}{2} \Delta t^2 \frac{\partial}{\partial x} Q(u) \right]_{(x_j, t_n)}. \tag{3.13}$$

Now approximate the x-derivatives using central differences, firstly

$$\left. \frac{\partial}{\partial x} F(u) \right|_{(x_j, t_n)} \approx \frac{F(w^n_{j+1}) - F(w^n_{j-1})}{2\Delta x} = \frac{F^n_{j+1} - F^n_{j-1}}{2\Delta x},$$

where we have introduced the shorthand $F^n_j = F(w^n_j)$, etc. Similarly for the 2nd space derivative

$$\left. \frac{\partial}{\partial x} Q(u) \right|_{(x_j, t_n)} \approx \frac{Q^n_{j+\frac{1}{2}} - Q^n_{j-\frac{1}{2}}}{\Delta x},$$

but with the crucial difference is that the central difference here is applied with a spacing of $\Delta x/2$. We have also used the notation

$$Q^n_{j+\frac{1}{2}} = F'(w^n_{j+\frac{1}{2}}) \left. \frac{\partial}{\partial x} F(u) \right|_{(x_{j+\frac{1}{2}}, t_n)} \approx F'(w^n_{j+\frac{1}{2}}) \left(\frac{F^n_{j+1} - F^n_j}{\Delta x} \right)$$

so

$$\left. \frac{\partial}{\partial x} Q(u) \right|_{(x_j, t_n)} \approx \frac{1}{\Delta x^2} \left\{ F'(w^n_{j+\frac{1}{2}}) \left(F^n_{j+1} - F^n_j \right) - F'(w^n_{j-\frac{1}{2}}) \left(F^n_j - F^n_{j-1} \right) \right\}.$$

Since we do not have values for u at half-integer points $x_{j+\frac{1}{2}}$, we further approximate

$$F'(w^n_{j+\frac{1}{2}}) \approx \frac{1}{2} \left(F'(w^n_j) + F'(w^n_{j+1}) \right)$$

and similarly for $F'(w^n_{j-\frac{1}{2}})$. Putting all this into (3.13) we get

$$w^{n+1}_j = w^n_j - \frac{\Delta t}{2\Delta x} \left(F^n_{j+1} - F^n_{j-1} \right) + \frac{\Delta t^2}{2\Delta x^2} \left\{ a^n_{j+\frac{1}{2}} \left[F^n_{j+1} - F^n_j \right] - a^n_{j-\frac{1}{2}} \left[F^n_j - F^n_{j-1} \right] \right\} \tag{3.14}$$

where

$$a^n_{j+\frac{1}{2}} = \frac{1}{2} \left[F'(w^n_j) + F'(w^n_{j+1}) \right] \approx F'(w^n_{j+\frac{1}{2}}).$$

This nonlinear version of the L-W scheme should reduce to the linear version when $F(u) = au$, $a =$ const, i.e. when $F' = a$. Check:

$$\begin{aligned} w^{n+1}_j &= w^n_j - \frac{\Delta t}{2\Delta x} \left(a w^n_{j+1} - a w^n_{j-1} \right) + \frac{\Delta t^2}{2\Delta x^2} \left\{ a \left[a w^n_{j+1} - a w^n_j \right] - a \left[a w^n_j - a w^n_{j-1} \right] \right\} \\ &= w^n_j - \frac{p}{2} \left(w^n_{j+1} - w^n_{j-1} \right) + \frac{p^2}{2} \left(w^n_{j+1} - 2 w^n_j + w^n_{j-1} \right), \quad p = \frac{a \Delta t}{\Delta x} \end{aligned}$$

which is the formula derived earlier.

In the linear scheme we have stability only for $|p| \leq 1$. In the nonlinear scheme, $F'(u)$ plays the role of a, so we should require that

$$\left| F'(w^n_j) \right| \frac{\Delta t}{\Delta x} \leq 1$$

for all m. For safety we replace "≤ 1" by "$= 0.9$" and require

$$\Delta t = \frac{0.9 \Delta x}{\max_j \left| F'(w^n_j) \right|}$$

Note that Δt will vary from step to step.

4 Elliptic PDE's

4.1 Introduction

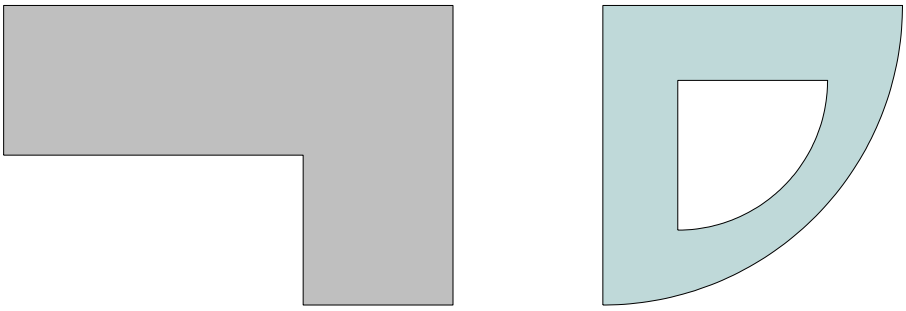
Typically Elliptic PDEs represent "equilibrium" problems. The most common (and most important) examples of an elliptic PDE is the *Poisson equation*

$$\nabla^2 u = f(x, y, \dots)$$
$$\text{In 2D: } u_{xx} + u_{yy} = f(x, y)$$

This models, for example, steady flow of incompressible fluids, potential fields obeying an inverse square law such as electrostatics or gravity, etc.

Note that f is a known function of $x, y, \dots$. When $f = 0$ we have the Laplace equation $\nabla^2 u = 0$, which for example models the steady state distribution of heat in a metal plate (2D) or other shapes in 3D, etc. This follows since the heat equation is $u_t = \nabla^2 u$.

An important feature of elliptic PDEs that we have not met so far is that quite often the equation is to be solved in an irregularly shaped domain Ω.

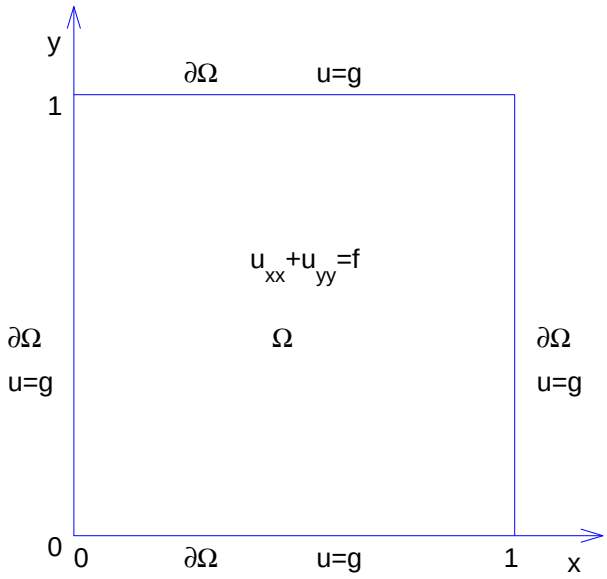


This is particularly important in engineering applications, where the equations model the distribution of temperature or of stress or strain throughout a body. The appropriate boundary conditions are that at each point on the boundary $\partial\Omega$, either u or its normal derivative $\partial u/\partial \mathbf{n}$ is specified.

We cover two different type of numerical methods for elliptic PDEs – *finite differences* and *finite elements*.

4.2 A finite difference example for $u_{xx} + u_{yy} = f(x, y)$.

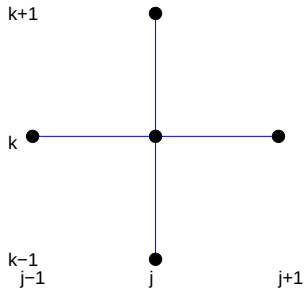
Consider the 2D Poisson equation applied on the square domain $(x, y) \in [0, 1]$ ($\Omega = [0, 1]^2$) and assume Dirichlet boundary conditions, $u(x, y) = g(x, y)$ on $\partial\Omega$.



Introduce a mesh and choose $\Delta x = \Delta y$ (assume there is no "preferred" space direction). Use $w_{j,k}$ to represent the finite difference approximation to the exact solution $u(x_j, y_k) \approx w_{j,k}$. Furthermore denote $f(x_j, y_k) = f_{j,k}$. Now use central difference approximations for both u_{xx} and u_{yy} to get after simplification

$$w_{j-1,k} + w_{j+1,k} + w_{j,k-1} + w_{j,k+1} - 4w_{j,k} = \Delta x^2 f_{j,k}. \tag{4.1}$$

We apply (4.1) at each internal point in the domain.



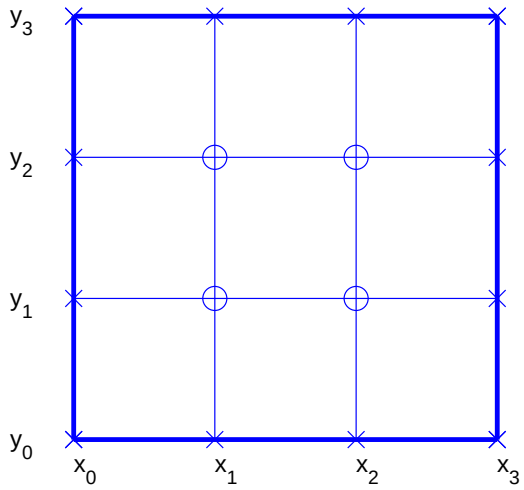
If $j = 0, \ldots, J$ and $k = 0, \ldots, K$ on the boundary of the square we will have in general $(J-1)(K-1)$ internal points and $(J-1)(K-1)$ unknowns, the $w_{j,k}$ at the internal points (of course in the square case we have $J = K$, but we keep the notation general to allow for rectangular regions). In matrix form we can write this as

$$S\mathbf{w} = \mathbf{b}$$

where

$$\mathbf{w} = \begin{pmatrix} w_{1,1} \\ w_{1,2} \\ \vdots \\ w_{1,K-1} \\ w_{2,1} \\ \vdots \\ w_{J-1,K-1} \end{pmatrix},$$

and $\mathbf{b}$ contains the boundary conditions. Each row of S will contain the coefficients in the lhs of (4.1), except those referring to boundary values. For example consider the case $\Delta x = 1/3$, $J = K = 3$. Then we have the figure shown



and the four equations from the PDE are

$$w_{0,1} + w_{2,1} + w_{1,0} + w_{1,2} - 4w_{1,1} = \frac{1}{9}f_{1,1}$$
$$w_{0,2} + w_{2,2} + w_{1,1} + w_{1,3} - 4w_{1,2} = \frac{1}{9}f_{1,2}$$
$$w_{1,1} + w_{3,1} + w_{2,0} + w_{2,2} - 4w_{2,1} = \frac{1}{9}f_{2,1}$$
$$w_{1,2} + w_{3,2} + w_{2,1} + w_{2,3} - 4w_{2,2} = \frac{1}{9}f_{2,2}.$$

The boundary conditions give us $w_{0,1} = g_{0,1}$, etc. so the four equations become

$$w_{2,1} + w_{1,2} - 4w_{1,1} = \frac{1}{9}f_{1,1} - g_{0,1} - g_{1,0}$$
$$w_{2,2} + w_{1,1} - 4w_{1,2} = \frac{1}{9}f_{1,2} - g_{0,2} - g_{1,3}$$
$$w_{1,1} + w_{2,2} - 4w_{2,1} = \frac{1}{9}f_{2,1} - g_{3,1} - g_{2,0}$$
$$w_{1,2} + w_{2,1} - 4w_{2,2} = \frac{1}{9}f_{2,2} - g_{3,2} - g_{2,3}.$$

or

$$S\mathbf{w} = \mathbf{b}$$

with

$$S = \begin{pmatrix} -4 & 1 & 1 & 0 \\ 1 & -4 & 0 & 1 \\ 1 & 0 & -4 & 1 \\ 0 & 1 & 1 & -4 \end{pmatrix}, \text{ and } \mathbf{b} = \begin{pmatrix} \frac{1}{9}f_{1,1} - g_{0,1} - g_{1,0} \\ \frac{1}{9}f_{1,2} - g_{0,2} - g_{1,3} \\ \frac{1}{9}f_{2,1} - g_{3,1} - g_{2,0} \\ \frac{1}{9}f_{2,2} - g_{3,2} - g_{2,3} \end{pmatrix}$$

Numerical Example: with the problem above, take $f(x,y) = 1$ and $g(x,0) = \sin \pi x$, $g(x,1) = g(0,y) = g(1,y) = 0$.
In this case the rhs vector is

$$\mathbf{b} = \begin{pmatrix} \frac{1}{9} - \frac{\sqrt{3}}{2} \\ \frac{1}{9} - \frac{\sqrt{3}}{2} \\ \frac{1}{9} \\ \frac{1}{9} \end{pmatrix},$$

since $\sin(\pi/3) = \sqrt{3}/2$. Solving this system by Gaussian elimination we get

$$\mathbf{w} = \begin{pmatrix} w_{1,1} \\ w_{1,2} \\ w_{2,1} \\ w_{2,2} \end{pmatrix} = \begin{pmatrix} -\frac{1}{18} + \frac{3\sqrt{3}}{16} \\ -\frac{1}{18} + \frac{3\sqrt{3}}{16} \\ -\frac{1}{18} + \frac{\sqrt{3}}{16} \\ -\frac{1}{18} + \frac{\sqrt{3}}{16} \end{pmatrix} = \begin{pmatrix} 0.26920 \\ 0.26920 \\ 0.05270 \\ 0.05270 \end{pmatrix}$$

Note the symmetry $w_{1,1} = w_{1,2}$, $w_{2,1} = w_{2,2}$, which we would expect since the BCs are symmetric about the line $x = 1/2$.
In general we would be using a smaller value of Δx and the matrix S will be large and sparse. We could solve the equation $S\mathbf{x} = \mathbf{b}$ directly, or using an iteration method such as the Gauss-Seidel or SOR method.
We next consider the LTE. Define

$$L_\Delta w_{j,k} = \frac{w_{j-1,k} - 2w_{j,k} + w_{j+1,k}}{\Delta x^2} + \frac{w_{j,k-1} - 2w_{j,k} + w_{j,k+1}}{\Delta x^2},$$

so the numerical solution satisfies $L_\Delta w_{j,k} - f_{j,k} = 0$. We define the LTE as

$$\begin{aligned} \text{LTE} &= L_\Delta u(x_j, y_k) - f_{j,k} \\ &= \frac{u(x_j - \Delta x, y_k) - 2u(x_j, y_k) + u(x_j + \Delta x, k)}{\Delta x^2} + \\ &\quad + \frac{u(x_j, y_k - \Delta x) - 2u(x_j, y_k) + u(x_j, y_k + \Delta x)}{\Delta x^2} - f_{j,k}, \\ &= \left[u_{xx} + \frac{1}{12}\Delta x^2 u_{xxxx} + O(\Delta x^4) \right]_{(x_j, y_k)} + \left[u_{yy} + \frac{1}{12}\Delta x^2 u_{yyyy} + O(\Delta x^4) \right]_{(x_j, y_k)} - f_{j,k}. \end{aligned}$$

Since u is a solution of $u_{xx} + u_{yy} = f$, the LTE after cancelling these terms is

$$\text{LTE} = \frac{1}{12}\Delta x^2 (u_{xxxx} + u_{yyyy}) + O(\Delta x^4).$$

Now consider convergence. The von Neumann stability analysis is not directly applicable here (the schemes do not develop over a series of steps, but are "all or nothing", i.e. we get the entire solution $w_{j,k}, \forall j, k$ in one step. How do we show convergence? We have

$$\begin{gathered} w_{j,k} \text{ is a solution of } L_\Delta w_{j,k} - f = 0 \\ \text{and } u(x_j, y_k) \text{ is a solution of } u_{xx} + u_{yy} - f = 0. \end{gathered}$$

Now define the error as $z_{j,k} = u(x_j, y_k) - w_{j,k}$. For convergence we require $|z_{j,k}| \to 0$ as $\Delta x \to 0$ for all j, k. Apply the finite difference operator L_Δ to $z_{j,k}$.

$$L_\Delta z_{j,k} = L_\Delta u(x_j, y_k) - L_\Delta w_{j,k} = \text{LTE}.$$

We now introduce a *comparison function* $C_{j,k} = x_j^2 + y_k^2$. We have

$$L_\Delta C_{j,k} = \frac{(x_j - \Delta x)^2 - 2x_j^2 + (x_j + \Delta x)^2}{\Delta x^2} + \frac{(y_k - \Delta x)^2 - 2y_k^2 + (y_k + \Delta x)^2}{\Delta x^2} = 2 + 2 = 4,$$

and $C_{j,k}$ is non-negative with a maximum value of 2 at $x = y = 1$. Now we define yet another function on the mesh, $c_{j,k}$

$$c_{j,k} = z_{j,k} + \frac{1}{4} C_{j,k} |\text{LTE}|_{\max}$$

so

$$\begin{aligned} L_\Delta c_{j,k} &= \text{LTE} + \frac{1}{4} \times 4 \, |\text{LTE}|_{\max} \\ &\geq 0, \quad \forall (x_j, y_k) \in \Omega \qquad (*) \end{aligned}$$

Now we claim that if $L_\Delta m_{j,k} \geq 0, \quad \forall (j,k)$ for any function $m_{j,k}$ (i.e. $c_{j,k}$), then $m_{j,k}$ attains its maximum value on the boundary. i.e. $(x_j, y_k) \in \partial\Omega$. (This is an example of what is known as a *maximum principle*).

Proof: Let $M = \max_{j,k=0,\ldots,J}[m_{j,k}]$, and assume this is attained at an interior point, say (j^*, k^*). This implies that

$$\begin{aligned} m_{j^*-1,k^*} + m_{j^*+1,k^*} + m_{j^*,k^*-1} + m_{j^*,k^*+1} - 4M &\geq 0 \\ \text{i.e. } M &\leq \frac{1}{4}(m_{j^*-1,k^*} + m_{j^*+1,k^*} + m_{j^*,k^*-1} + m_{j^*,k^*+1}) \\ &\leq \frac{1}{4}(4M) = M. \end{aligned}$$

Equality is only possible if all interior points and their neighbours take the value M, so the maximum value must be attained somewhere on the boundary. (We have assumed here that $M \geq 0$ which is sufficient for our needs).

Hence from (*), $c_{j,k}$ must attain its maximum value on the boundary, however $z_{j,k}$ vanishes on the boundary if the B.C. data is exact. Furthermore the maximum value of $C_{j,k}$ on the boundary is 2, so $\max(c_{j,k}) = \frac{1}{2}|\text{LTE}|_{\max}$.

Now we have

$$\begin{aligned} z_{j,k} &= c_{j,k} - \frac{1}{4} C_{j,k} |\text{LTE}|_{\max} \quad \text{by definition} \\ &\leq c_{j,k}, \quad \forall j,k, \quad \text{since } C_{j,k} \text{ is non-negative} \\ \text{so } z_{j,k} &\leq \frac{1}{2} |\text{LTE}|_{\max}. \end{aligned}$$

We can similarly repeat the analysis to place a lower bound on $z_{j,k}$. Define

$$\begin{aligned} \bar{c}_{j,k} &= -z_{j,k} + \frac{1}{4} C_{j,k} |\text{LTE}|_{\max} \\ \text{so } L_\Delta \bar{c}_{j,k} &= -\text{LTE} + |\text{LTE}|_{\max} \\ &\geq 0, \quad \forall j,k. \end{aligned}$$

so the max of $\bar{c}_{j,k}$ is attained on the boundary and is equal to $\frac{1}{2}|\text{LTE}|_{\max}$. Hence

$$\begin{aligned} z_{j,k} &= -\bar{c}_{j,k} + \frac{1}{4}C_{j,k}|\text{LTE}|_{\max} \\ &\geq -\bar{c}_{j,k} \qquad \text{since } C_{j,k} \text{ is nonnegative} \\ &\geq -\frac{1}{2}|\text{LTE}|_{\max}. \end{aligned}$$

We have finally that

$$\boxed{-\frac{1}{2}|\text{LTE}|_{\max} \leq z_{j,k} \leq \frac{1}{2}|\text{LTE}|_{\max}}$$

so $z_{j,k} \to 0$ as $\Delta x \to 0$ since the scheme is consistent. This is sufficient to prove convergence.

Derivative Boundary conditions

We can use the fictitious point method to approximate derivative boundary conditions in the same way as before - see Tutorial sheet 5.

The standard Finite Difference scheme does not work well on curved boundaries - better to use finite elements. Before we look at finite elements we need to review how we can write the PDE in variational form.

4.3 Elliptic PDE's as variational problems

4.3.1 1D case

It helps to consider the 1D case first, when the Poisson equation becomes an ODE

$$\frac{d^2u(x)}{dx^2} = f(x),$$

on some interval, say $\Omega = [0, 1]$. To write this as a variational problem, consider the functional

$$J[v] = \int_\Omega \left[\frac{1}{2}\left(\frac{dv(x)}{dx}\right)^2 + f(x)v(x)\right] dx \tag{4.2}$$

Now we claim that the function that minimises $J[v]$ is exactly $u(x)$, the solution of the ODE. That is $(\delta J[v]/\delta v)|_{v=u} = 0$, i.e.,

$$(\delta J[v]/\delta v)|_{v=u} = \lim_{\epsilon \to 0} \frac{J[u + \epsilon\phi] - J[u]}{\epsilon} = 0,$$

for a sufficiently smooth function ϕ which is zero on the boundary.

To prove this let $v = u + \epsilon\phi$ be a function for which $J[v]$ is defined and which satisfies the boundary conditions of the ODE (i.e. $\phi = 0$ at the boundaries). Then

$$\begin{aligned} \delta J &\equiv J[u + \epsilon\phi] - J[u] \\ &= \int_\Omega \left[\frac{1}{2}\left(\left(\frac{du}{dx} + \frac{d\epsilon\phi}{dx}\right)^2 - \left(\frac{du}{dx}\right)^2\right) + f(x)\left((u + \epsilon\phi) - u\right)\right] \mathrm{d}x \\ &= \int_\Omega \left[\left(\frac{du}{dx}\right).\left(\frac{d\epsilon\phi}{dx}\right) + f\epsilon\phi + \frac{1}{2}\left(\frac{d\epsilon\phi}{dx}\right)^2\right] dx \end{aligned}$$

We can simplify the first term in this expression by using integration by parts:

$$\int_\Omega \epsilon\phi \left(\frac{d^2u}{dx^2}\right) dx = \left[\epsilon\phi\frac{du}{dx}\right]_0^1 - \int_\Omega \left(\frac{d\epsilon\phi}{dx}\right).\left(\frac{du}{dx}\right) dx$$

Hence we have

$$\delta J = \epsilon \int_\Omega \left[-\frac{d^2u}{dx^2} + f\right] \phi\, dx \; + \epsilon \left[\phi\frac{du}{dx}\right]_0^1 + \frac{\epsilon^2}{2}\frac{d\phi}{dx}$$

Furthermore the second term in this expression is identically zero because $\phi = 0$ on the boundary.

At a minimum $\delta J/\epsilon$ will vanish for $\epsilon \to 0$. We see this can only happen if $d^2u/dx^2 = f$ since ϕ is arbitrary inside Ω. In other words, if $u(x)$ is the function which minimises $J[u]$ then the function must satisfy $d^2u/dx^2 = f$.

The result above is exact. How do we use it to find an approximate solution to (4.3.1)? We do this by writing

$$v(x) \approx \sum_{k=1}^{N} c_k\phi_k(x), \tag{4.3}$$

where the $\phi_k(x)$ are a known set of *basis* functions and the c_k are unknown coefficients. We then insert this into (4.2) to get

$$J[\mathbf{c}] = \int_\Omega \left[\frac{1}{2}\left(\sum_{k=1}^{N} c_k\frac{d\phi_k(x)}{dx}\right)^2 + \sum_{k=1}^{N} c_k f(x)\phi_k(x)\right] dx.$$

Now minimise over the c_k, for this we require that $\partial J/\partial c_j = 0,\ j = 1, \ldots, N$. This implies that

$$\begin{aligned}
&\int_\Omega \left[\frac{d\phi_j}{dx}\left(\sum_{k=1}^{N} c_k\frac{d\phi_k}{dx}\right) + f(x)\phi_j(x)\right] dx = 0 \\
&\Rightarrow \sum_{k=1}^{N} c_k \int_\Omega \left(\frac{d\phi_j}{dx}\frac{d\phi_k}{dx}\right) dx + \int_\Omega f(x)\phi_j(x)dx = 0 \\
&\Rightarrow \sum_{k=1}^{N} a_{jk}c_k + b_j = 0,
\end{aligned}$$

where

$$a_{j,k} = \int_\Omega \left(\frac{d\phi_j}{dx}\frac{d\phi_k}{dx}\right) dx, \quad b_j = \int_\Omega f(x)\phi_j(x).$$

In matrix form this is

$$A\mathbf{c} = -\mathbf{b}$$

where

$$A = \{a_{jk}\}, \quad \mathbf{b} = \{b_j\}.$$

If we solve these equations for $\mathbf{c}$, we then recover our approximation (4.3).

We now need to consider how to choose the basis functions $\phi_j(x)$. It is this specific choice that determines the finite element method. We wish these functions to be as simple as possible, and to vanish outside a small number of *elements* - in 1D the elements are small line segments $[x_{j-1}, x_j], j = 1 \ldots J$. The points x_j are called the *nodes*. We need the $\phi_j(x)$ to possess first derivatives, in order to calculate the a_{jk}.

To keep things simple, we consider BCs such that $u(0) = u(1) = 0$, and also assume $f(x)$ is the constant function f. In this case the simplest basis functions $\phi_n(x)$ are defined as *piecewise linear* functions as shown in the figure below.

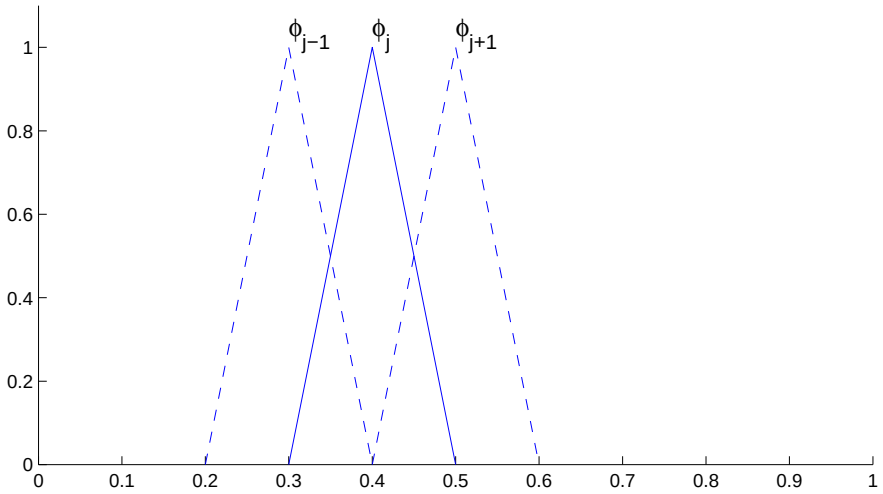


Figure 3: Piecewise linear *tent* functions, in the case $x_j = j\Delta x, \Delta x = 0.1$

So

$$\phi_j(x) = \begin{cases} (x - x_{j-1})/(x_j - x_{j-1}), & x_{j-1} \le x \le x_j \\ (x_{j+1} - x)/(x_{j+1} - x_j), & x_j \le x \le x_{j+1} \\ 0, \quad \text{otherwise.} \end{cases}$$

We see that each $\phi_j(x)$ is nonzero over only two elements, $[x_{j-1}, x_j]$ and $[x_j, x_{j+1}]$, and takes the value 1 at $x = x_j$. Since $u(x) = 0$ on the boundaries, in total there are $J-1$ basis functions $\phi_j, j = 1, \ldots, J-1$.

We can now calculate b_j easily

$$b_j = \int_0^1 f(x)\phi_j(x)\,dx = \int_0^1 f\phi_j(x)\,dx = f\int_{x_j-1}^{x_j} \frac{(x - x_{j-1})}{(x_j - x_{j-1})}\,dx + f\int_{x_j}^{x_{j+1}} \frac{(x - x_{j+1})}{(x_j - x_{j+1})}\,dx,$$
$$= \frac{f}{2}\frac{(x - x_{j-1})^2}{(x_j - x_{j-1})}\Big|_{x=x_{j-1}}^{x=x_j} + \frac{f}{2}\frac{(x - x_{j+1})^2}{(x_j - x_{j+1})}\Big|_{x=x_j}^{x=x_{j+1}} = \frac{f}{2}(x_j - x_{j-1}) + \frac{f}{2}(x_{j+1} - x_j).$$

In the case where the nodes are equally spaced, $x_j - x_{j-1} = \Delta x$ for all j, and hence $b_j = f\Delta x$. We could have by-passed the integration process by noting that the integral is just $f\times$ the area of a triangle with height 1 and base $2\Delta x$.

Now consider the matrix elements $a_{j,k}$. A little thought shows these will be zero if $j < k-1$ or $j > k+1$, since in this case either ϕ_j' or ϕ_k' will be zero for all values of x. So we have three

nonzero elements to calculate for each choice of j: $a_{j-1,j}, a_{j,j}$, and $a_{j,j+1}$. We note that ϕ_j' is a piecewise constant function

$$\phi_j'(x) = \begin{cases} 1/(x_j - x_{j-1}), & x_{j-1} \le x \le x_j \\ -1/(x_{j+1} - x_j), & x_j \le x \le x_{j+1} \\ 0, & \text{otherwise.} \end{cases}$$

First calculate $a_{jj} = \int \phi_j'(x)^2\, dx$. The integrand is nonzero over both $[x_{j-1}, x_j]$ and $[x_j, x_{j+1}]$.

$$\begin{aligned} a_{jj} &= \int \phi_j'(x)^2\, dx = \int_{x_{j-1}}^{x_j} \frac{1}{(x_j - x_{j-1})^2}\, dx + \int_{x_j}^{x_{j+1}} \frac{1}{(x_j - x_{j+1})^2}\, dx, \\ &= \frac{1}{(x_j - x_{j-1})} + \frac{1}{(x_{j+1} - x_j)} \end{aligned}$$

In the equally spaced case, we can simplify the final result to $a_{jj} = 2/\Delta x$

Now consider $a_{j-1,j} = \int \phi_{j-1}'(x)\, \phi_j'(x)\, dx$. The integrand is nonzero only over $[x_{j-1}, x_j]$. .

$$a_{j-1\,j} = \int_{x_{j-1}}^{x_j} \phi_{j-1}'(x)\, \phi_j'(x)\, dx = \int_{x_{j-1}}^{x_j} \frac{-1}{(x_j - x_{j-1})} \cdot \frac{1}{(x_j - x_{j-1})}\, dx = -\frac{1}{(x_j - x_{j-1})}$$

In the equally spaced case, this gives $a_{j-1\,j} = -1/\Delta x$.

A similar calculation shows that $a_{j,j+1} = -1/(x_{j+1} - x_j)$, which reduces to $-1/\Delta x$ in the equally spaced case.

So finally, in the equally spaced case, we have

$$\begin{pmatrix} 2/\Delta x & -1/\Delta x & 0 & & \\ -1/\Delta x & 2/\Delta x & -1/\Delta x & 0 & \\ \ddots & \ddots & \ddots & \ddots & \\ & 0 & -1/\Delta x & 2/\Delta x \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{J-1} \end{pmatrix} = - \begin{pmatrix} f\Delta x \\ f\Delta x \\ \vdots \\ f\Delta x \end{pmatrix}. \tag{4.4}$$

If we multiply through by $-\Delta x$, we see that this is exactly what we would get from a central difference approximation to the original o.d.e., so in this case the finite element method is not giving anything new. However if f was not a constant the results would be different. For example if f is piecewise linear, and $f(0) = f(1) = 0$, we can write

$$f(x) = \sum_{k=1}^{J-1} f_k \phi_k(x),$$

(why?) and then we find after some calculation that

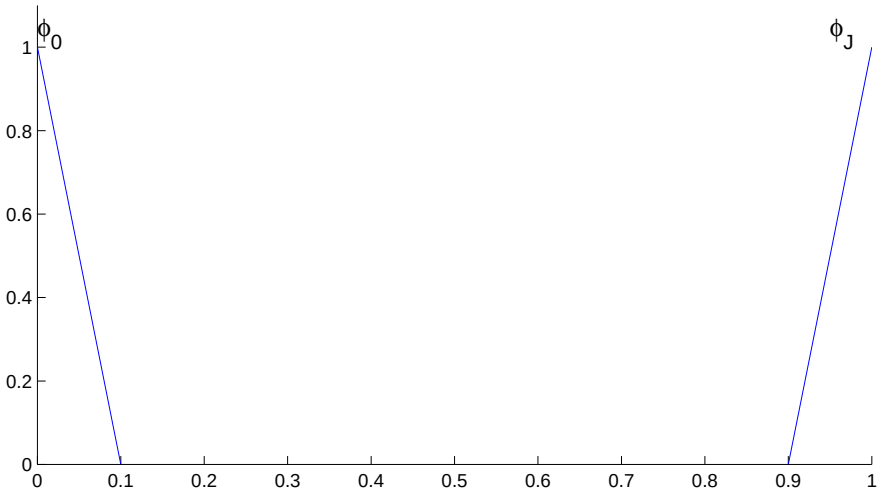
$$b_j = \int_0^1 \phi_j(x) \left(\sum_{k=1}^{J-1} f_k \phi_k(x) \right) dx = \frac{\Delta x}{6} \left(f_{j-1} + 4 f_j + f_{j+1} \right),$$

(check!). In the finite difference approach this would be just $\Delta x f_j$.

If we have nonzero boundary conditions for $u(0)$ and/or $u(1)$, say $u(0) = \alpha, u(1) = \beta$ we can easily deal with this by adding extra "end" basis functions to the approximation to $u(x)$

$$v(x) \approx \tilde{v}(x) = \alpha\phi_0 + \sum_{k=1}^{J-1} c_k \phi_k(x) + \beta\phi_J$$

where ϕ_0 and ϕ_J are shown in the figure.



Work out for yourselves how this will modify equations (4.4), assuming f constant.

Neumann (derivative) boundary condition (advanced)

Consider the 1D Poisson equation on $\Omega = [0, 1]$

$$\frac{d^2u(x)}{dx^2} = f(x),$$

with Dirichlet boundary condition at $x = 0$, i.e. $u(0) = 0$ and a derivative boundary condition at $x = 1$, i.e.,

$$\frac{du(1)}{dx} = g.$$

The integration by part formula implies (note, we need $\phi(0) = 0$ because of the first boundary codnition)

$$\int_\Omega \phi \left(\frac{d^2u}{dx^2}\right) dx = \left[\phi \frac{du}{dx}\right]_0^1 - \int_\Omega \left(\frac{d\phi}{dx}\right) . \left(\frac{du}{dx}\right) dx\,.$$

The boundary term above can be evaluated using the boundary conditions

$$\left[\phi \frac{du}{dx}\right]_0^1 = \underbrace{\phi(1)\frac{du(1)}{dx}}_{=\phi(1)g} - \underbrace{\phi(0)\frac{du(0)}{dx}}_{\phi(0)=0} = \phi(1)g\,.$$

The "exact" *variation formulation* of the Poisson problem now becomes

$$\int_0^1 \frac{du(x)}{dx}\frac{d\phi(x)}{dx}dx = \phi(1)g - \int_0^1 f(x)\phi(x)dx.$$

Finite Element Method - 2D We can follow the same variational approach as in 1D, with some extra complication due to the fact that all integrals are now over 2D regions.

Initially assume Dirichlet boundary conditions are applied on all boundaries. Take for simplicity the Poisson equation (more general PDE's can also be treated by this method)

$$\nabla^2 u = f(x, y) \tag{4.5}$$

in some domain Ω. Furthermore we define

$$J[v] = \int\int_\Omega \left[\frac{1}{2}\left(\nabla v(x,y)\right)^2 + f(x,y)v(x,y)\right] dx\,dy \tag{4.6}$$

Now we claim that the function that minimises $J[v]$ is exactly $u(x,y)$, the solution of the PDE. That is $(\delta J[v]/\delta v)|_{v=u} = 0$, i.e.,

$$(\delta J[v]/\delta v)|_{v=u} = \lim_{\epsilon\to 0}\frac{J[u+\epsilon\phi] - J[u]}{\epsilon} = 0,$$

for a sufficiently smooth function ϕ which is zero on the boundary.
To prove this let $v = u + \epsilon\phi$ be a function for which $J[v]$ is defined and which satisfies the boundary conditions of the PDE (i.e. $\phi = 0$ at the boundaries). Then

$$\begin{aligned}
\delta J &\equiv J[u+\epsilon\phi] - J[u] \\
&= \int\int_\Omega \left[\frac{1}{2}\left((\nabla u + \epsilon\nabla\phi)^2 - (\nabla u)^2\right) + f\left((u+\epsilon\phi) - u\right)\right] dx\,dy \\
&= \int\int_\Omega \left[\epsilon(\nabla u).(\nabla\phi) + \epsilon f\phi + \frac{\epsilon^2}{2}(\nabla\phi)^2\right] dx\,dy
\end{aligned}$$

We can simplify the first term in this expression by using the vector calculus equivalent of integration by parts (the divergence theorem):

$$\int\int_\Omega \phi(\nabla^2 u)\,dx\,dy = \int_{\partial\Omega}(\mathbf{n}.\nabla u)\phi dS - \int\int_\Omega (\nabla\phi).(\nabla u)\,dx\,dy$$

where $\mathbf{n}$ is the outward normal at the boundary (note $\mathbf{n}.\nabla u = \partial u/\partial n$). Hence we have

$$\delta J = \epsilon\int\int_\Omega \left[-\nabla^2 u + f\right]\phi\,dx\,dy + \epsilon\int_{\partial\Omega}\phi\frac{\partial u}{\partial n}dS + \mathcal{O}(\epsilon^2)$$

Furthermore the second term in this expression is identically zero because $\phi = 0$ on $\partial\Omega$.

At a minimum $\delta J/\epsilon$ will vanish for $\epsilon \to 0$. We see this can only happen if $\nabla^2 u = f$ since ϕ is arbitrary inside Ω. In other words, if $u(x,y)$ is the function which minimises $J[u]$ then the function must satisfy $\nabla^2 u = f$.

$$\Rightarrow \text{solving PDE (4.5)} \Leftrightarrow \text{minimising (4.6)}$$

As in the 1D case, this result is exact. How do we use it to find an approximate solution to (4.5)? We do this by generalising the basis function approximation idea to 2D

$$v(x,y) \approx \sum_{k=1}^{N} c_k\phi_k(x,y), \tag{4.7}$$

where the $\phi_k(x,y)$ are a known set of basis functions and the c_k are unknown coefficients. We then insert this into (4.6) to get

$$J[\mathbf{c}] = \int\int_\Omega \left[\frac{1}{2}\left(\sum_{k=1}^{N} c_k\frac{\partial\phi_k(x,y)}{\partial x}\right)^2 + \frac{1}{2}\left(\sum_{k=1}^{N} c_k\frac{\partial\phi_k(x,y)}{\partial y}\right)^2 + \sum_{k=1}^{N} c_k f(x,y)\phi_k(x,y)\right] dx\,dy.$$

Now minimise over the c_k, for this we require that $\partial J/\partial c_j = 0,\ j = 1, \ldots, N$

$$\begin{aligned}
&\frac{\partial J}{\partial c_j} = 0 \\
&\Rightarrow \int\int_\Omega \left[\frac{\partial \phi_j}{\partial x} \left(\sum_{k=1}^{N} c_k \frac{\partial \phi_k}{\partial x} \right) + \frac{\partial \phi_j}{\partial y} \left(\sum_{k=1}^{N} c_k \frac{\partial \phi_k}{\partial y} \right) + f(x,y)\phi_j(x,y) \right] dx\, dy = 0 \\
&\Rightarrow \sum_{k=1}^{N} c_k \int\int_\Omega \left(\frac{\partial \phi_j}{\partial x}\frac{\partial \phi_k}{\partial x} + \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_k}{\partial y} \right) dxdy + \int\int_\Omega f(x,y)\phi_j(x,y)\, dx\, dy = 0 \\
&\Rightarrow \sum_{k=1}^{N} a_{j,k}c_k + b_j = 0,
\end{aligned}$$

where

$$a_{j,k} = \int\int_\Omega \left(\frac{\partial \phi_j}{\partial x}\frac{\partial \phi_k}{\partial x} + \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_k}{\partial y} \right) dx\, dy, \quad b_j = \int\int_\Omega f(x,y)\phi_j(x,y)\, dx\, dy.$$

In matrix form this is given by

$$A\mathbf{c} = -\mathbf{b}$$

as in the 1D case, where $A = \{a_{j,k}\}$, $\mathbf{b} = \{b_j\}$. If we solve these equations for $\mathbf{c}$, we then recover our approximation (4.7).

Neumann boundary condition in 2D (advanced)

Conside the 2D Poisson problem on a domain Ω

$$\nabla^2 u = f(x,y)\,.$$

The boundary is split into two disjoint parts $\partial\Omega = \Gamma_1 \cup \Gamma_2$. We take the following boundary conditions

$$\nabla u(x) \cdot \mathbf{n} = g(x) \qquad \text{on} \qquad \Gamma_1,$$

and

$$u(x) = 0 \qquad \text{on} \qquad \Gamma_2.$$

We consider functions $\phi|_{\Gamma_2} = 0$. Integration by parts then gives

$$\begin{aligned}
\int\int_\Omega (\nabla^2 u)\phi\, dx\, dy &= \int_{\partial\Omega} (\mathbf{n}.\nabla u)\phi\, dS - \int\int_\Omega (\nabla\phi).(\nabla u)\, dx\, dy \\
&= \int_{\Gamma_1} (\mathbf{n}.\nabla u)\phi\, dS + \underbrace{\int_{\Gamma_2} (\mathbf{n}.\nabla u)\phi\, dS}_{\phi|_{\Gamma_2}=0} - \int\int_\Omega (\nabla u).(\nabla\phi)\, dx\, dy \\
&= \int_{\Gamma_1} g\phi\, dS - \int\int_\Omega (\nabla u).(\nabla\phi)\, dx\, dy.
\end{aligned}$$

Finally, the *variational formulation* of the 2D Poisson problem reads as

$$\int\int_\Omega (\nabla u)\cdot(\nabla\phi)\, dx\, dy = \int_{\Gamma_1} g\phi\, dS - \int\int_\Omega f\phi\, dx\, dy.$$

4.4 Finite Element Method - 2D

This section describes how to apply the Finite Element Method in 2D, to solve

$$u_{xx} + u_{yy} = f \qquad \text{in} \qquad \Omega. \tag{4.8}$$

In what follows we assume for simplicity that f is a constant, that our domain Ω is the unit square, and that the boundary conditions are $u = 0$ everywhere.
From the lecture notes we have

$$u(x, y) \approx w(x, y) = \sum_{i=1}^{N} c_k \phi_k(x, y)$$

where the c_k satisfy the equation.

$$A\mathbf{c} = -\mathbf{b},$$

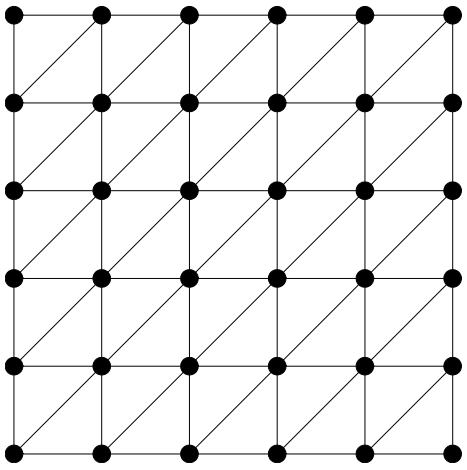
where $A = \{a_{j,k}\}$, and

$$\mathbf{c} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_N \end{pmatrix}, \qquad \mathbf{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{pmatrix}.$$

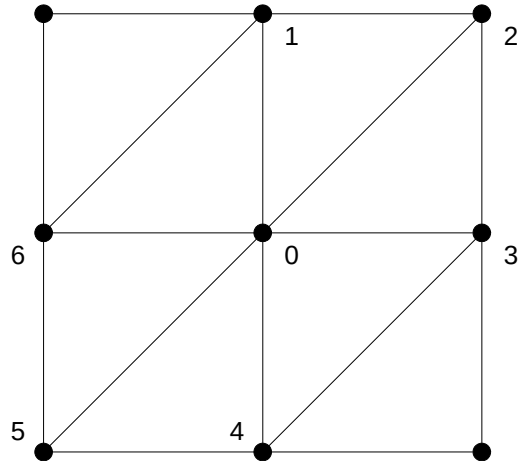
$a_{j,k}$ and b_j are defined through

$$a_{j,k} = \int_0^1 \int_0^1 \frac{\partial \phi_j}{\partial x}\frac{\partial \phi_k}{\partial x} + \frac{\partial \phi_j}{\partial y}\frac{\partial \phi_k}{\partial y}\, dx\, dy, \qquad b_j = \int_0^1 \int_0^1 f\phi_j\, dx\, dy$$

For the Finite element method in 2D, we need to divide our domain into elements, but now these elements are two-dimensional. The simplest case is to take the elements to be triangles, and we shall follow this here. Further, for simplicity we adopt the regular triangulation shown in the figure below. The vertexes of the triangular elements are called the *nodes*.



We concentrate on one of the interior nodes, which for convenience we will label '0', and surrounding nodes, labelled '1' to '6'.



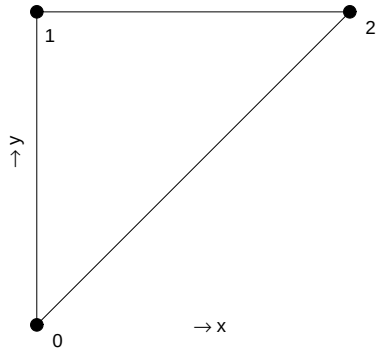
In each triangle $\Delta 012$, $\Delta 023$, etc., $\phi_0(x, y)$ is chosen to be a *linear* function of x and y.

$$\phi_0(x, y) = a_0 + b_0 x + c_0 y$$

but with a *different* a_0, b_0, c_0 in each triangle. Consider $\Delta 012$. The constants a_0, b_0, c_0 are chosen such that $\phi_0(x_0, y_0) = 1, \phi_0(x_1, y_1) = 0, \phi_0(x_2, y_2) = 0$, where (x_k, y_k) are the coordinates of node k. Similarly in $\Delta 012$,

$$\phi_1(x, y) = a_1 + b_1 x + c_1 y,$$

with the constants a_1, b_1, c_1 chosen such that $\phi_1(x_0, y_0) = 0, \phi_1(x_1, y_1) = 1, \phi_1(x_2, y_2) = 0$.
The only contribution to the matrix element a_{01} will occur in triangles $\Delta 012$ and $\Delta 016$. Consider $\Delta 012$, with the squares with side h. Without loss of generality we can move the origin to node 0.



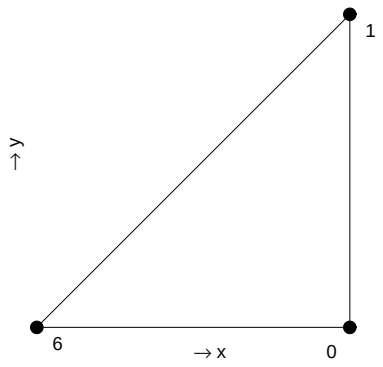
We have

$$\phi_0 = \frac{1}{h}(h - y), \quad \phi_1 = \frac{1}{h}(y - x), \quad \phi_2 = \frac{1}{h}x,$$
$$\frac{\partial \phi_0}{\partial x} = 0, \quad \frac{\partial \phi_1}{\partial x} = -\frac{1}{h}, \quad \frac{\partial \phi_0}{\partial y} = -\frac{1}{h}, \quad \frac{\partial \phi_1}{\partial y} = \frac{1}{h},$$

The contribution to a_{01} from $\Delta 012$ is

$$\iint_{\Delta 012} \left(0 \cdot \left(\frac{-1}{h}\right) + \left(\frac{-1}{h}\right) \cdot \left(\frac{1}{h}\right)\right) dx\, dy = -\frac{1}{h^2} \cdot \frac{h^2}{2} = -\frac{1}{2}$$

Consider now $\Delta 016$.



A short calculation shows that

$$\phi_0 = \frac{1}{h}(h + x - y), \quad \phi_1 = \frac{1}{h}y, \quad \phi_6 = -\frac{1}{h}x,$$
$$\frac{\partial \phi_0}{\partial x} = \frac{1}{h}, \quad \frac{\partial \phi_1}{\partial x} = 0, \quad \frac{\partial \phi_0}{\partial y} = -\frac{1}{h}, \quad \frac{\partial \phi_1}{\partial y} = \frac{1}{h}.$$

The contribution to a_{01} from $\Delta 016$ is

$$\iint_{\Delta 016} \left(\frac{-1}{h}\right) \cdot \left(\frac{1}{h}\right) dx\, dy = -\frac{1}{2}.$$

Hence $a_{01} = -\frac{1}{2} - \frac{1}{2} = -1$. By symmetry, $a_{03} = a_{04} = a_{06} = a_{01} = -1$.
Now consider a_{02}. This has contributions from $\Delta 012$ and $\Delta 023$. In $\Delta 012$

$$\frac{\partial \phi_0}{\partial x} = 0, \quad \frac{\partial \phi_2}{\partial x} = \frac{1}{h}, \quad \frac{\partial \phi_0}{\partial y} = -\frac{1}{h}, \quad \frac{\partial \phi_2}{\partial y} = 0,$$

so the contribution to a_{02} is zero. By symmetry, the contribution from $\Delta 023$ is zero also. So $a_{02} = 0$, and by symmetry, $a_{05} = 0$ also.
Now we calculate a_{00}. In $\Delta 012$ we get the contribution

$$\iint_{\Delta 012} \left(0^2 + \frac{1}{h^2}\right) dx\, dy = \frac{1}{2}.$$

By symmetry we get the same contributions from $\Delta 023$, $\Delta 045$, and $\Delta 056$. The contribution from $\Delta 016$ is

$$\iint_{\Delta 016} \left(\frac{1}{h^2} + \frac{1}{h^2}\right) dx\, dy = 1.$$

Total is

$$a_{00} = \frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{2} + 1 + 1 = 4.$$

Finally we calculate b_0. The contribution from $\Delta 012$ is

$$\begin{aligned} f \int_0^h \int_{x=0}^{x=y} \frac{1}{h}(h - y)\, dx\, dy &= \frac{f}{h} \int_0^h y(h - y) dy \\ &= \frac{f}{h} \int_0^h hy dy - \frac{f}{h} \int_0^h y^2 dy \\ &= \left[\frac{fy^2}{2} - \frac{fy^3}{3h}\right]_0^h = \frac{fh^2}{6}. \end{aligned}$$

By symmetry, there is the same contribution from $\Delta 023$, $\Delta 045$, $\Delta 056$.
The contribution from $\Delta 016$ is

$$\begin{aligned}\frac{f}{h}\int_0^h \int_{x=y-h}^{x=0} (h+x-y)\,dx\,dy &= \frac{f}{2h}\int_0^h (h+x-y)^2\Big|_{x=y-h}^{x=0} dy \\ &= \frac{f}{2h}\int_0^h (h-y)^2 dy = -\frac{f}{6h}(h-y)^3\Big|_0^h \\ &= \frac{fh^2}{6}.\end{aligned}$$

By symmetry, same contribution from $\Delta 034$. So the total contribution to b_0 is

$$b_0 = 6 \times \frac{fh^2}{6} = h^2 f.$$

So finally the equation for c_0 from node 0 is

$$-(c_1 + c_3 + c_4 + c_6) + 4c_0 = -h^2 f$$

or

$$(c_1 + c_3 + c_4 + c_6) - 4c_0 = h^2 f$$

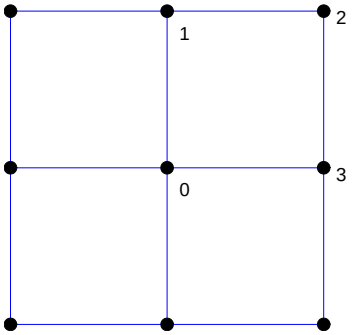
This is the same, in this simple case, as the Finite Difference approximation to

$$u_{xx} + u_{yy} = f.$$

Note however that in the more general case we can cover a more complicated area with triangles to deal with odd shapes.

4.4.1 Rectangular elements and bilinear basis functions

If we can cover our region Ω with rectangular elements, we can use *bilinear* basis functions instead of linear basis functions. Consider for example the square 0123 in the figure below.



We can define 4 bilinear basis functions ϕ_k in this square, each of which takes the value 1 at node k and zero at the other 3 nodes

$$\phi_0(x,y) = \frac{1}{h^2}(h-x)(h-y), \qquad \phi_1(x,y) = \frac{1}{h^2}(h-x)y$$
$$\phi_2(x,y) = \frac{1}{h^2}xy, \qquad \phi_3(x,y) = \frac{1}{h^2}x(h-y)$$

We then rerun the previous calculations with these new basis functions and their equivalents in the other squares. The functions to be integrated are slightly more complicated, but the regions of integration are simpler.

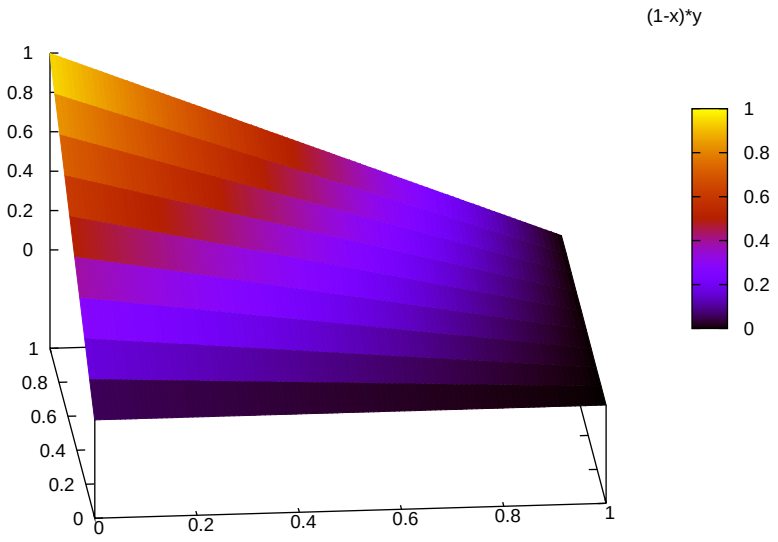


Figure 4: Plot of the ϕ_1 bilinear basis function for $h = 1$.

4.5 Stability and error estimates for the finite element method for elliptic problems (advanced)

In this section we derive estimates of the error between the exact solution u for elliptic problems

$$(\nabla u, \nabla \phi) + (u, \phi) = (f, \phi) \quad \forall \phi \in V, \tag{4.9}$$

and the finite element solution w

$$(\nabla w, \nabla \phi) + (w, \phi) = (f, \phi) \qquad \forall \phi \in V_h. \tag{4.10}$$

In what follows we assume for simplicity that f is a constant, that our domain Ω is the unit square, and that the boundary conditions are $u = 0$ everywhere.
In the above equations the notation (ϕ, ψ) means

$$(\phi, \psi) = \int_\Omega \phi\psi \, \mathrm{d}\Omega$$

V_h is the space of piecewise linear finite element basis functions defined earlier.

4.5.1 Sobolev spaces

The $L_2(\Omega)$ Hilbert space is a space of integrable real-valued functions $\phi : \Omega \to \mathbb{R}$ for which the following norm (or integral) is bounded:

$$\|\phi\|^2_{L_2(\Omega)} = \int_\Omega \phi^2 \mathrm{d}\Omega < \infty.$$

In the following text we will call the above norm the L_2-norm and for simplicity use the notation without the subscript, i.e., $\|\cdot\| \equiv \|\cdot\|_{L_2(\Omega)}$.
The functions ϕ belonging to the space $L_2(\Omega)$ are also called $L_2(\Omega)$-integrable.
The space of L_2-integrable functions $\phi \in L_2(\Omega)$ is equipped with the following scalar product

$$(\psi, \phi) = \int_\Omega \psi\phi \, \mathrm{d}\Omega < \infty.$$

Note that the scalar product satisfies

$$(\phi, \phi) = \|\phi\|^2.$$

Further the L_2-scalar product satisfies the so-called Cauchy-Schwartz inequality

$$|(\phi, \psi)| \leq \|\phi\| \, \|\psi\| \, . \tag{4.11}$$

The space $H^1(\Omega)$ (also called first Sobolev space) is a space of functions which are bounded in the following norm

$$\|\phi\|^2_{H^1(\Omega)} = \|\phi\|^2 + \|\nabla\phi\|^2 < \infty.$$

Thus the $H^1(\Omega)$ space contains L_2-integrable functions with L_2-integrable first order derivatives.

The space $H^2(\Omega)$ (also called second Sobolev space) is the space of functions which are bounded in the following norm

$$\|\phi\|^2_{H^2(\Omega)} = \|\phi\|^2 + \|\nabla\phi\|^2 + \|\nabla^2\phi\|^2 < \infty. \tag{4.12}$$

Thus the $H^2(\Omega)$ space contains L_2-integrable functions with L_2-integrable first and second order derivatives. Note, that the norm (4.12) is a slightly simplified version of the true H^2-norm, but the two norms are equivalent for simple domains Ω, such as considered here.

Further we define the H^1 and H^2 semi-norms as

$$|\phi|_{H^1(\Omega)} = \|\nabla\phi\| \qquad \text{and} \qquad |\phi|_{H^2(\Omega)} = \|\nabla^2\phi\|,$$

respectively.

The $H^1(\Omega)$ and $H^2(\Omega)$ spaces are more general analogues of the spaces $C^1(\Omega)$ (smooth functions with continuous first order derivatives) and $C^2(\Omega)$ (smooth functions with continuous second order derivatives). We also have that $C^1(\Omega) \subset H^1(\Omega)$, $C^2(\Omega) \subset H^2(\Omega)$ and $H^2(\Omega) \subset H^1(\Omega) \subset L_2(\Omega)$.

The following inequality (which is a combination of the Cauchy-Schwarz (4.11) and Young's ($|a|\,|b| \le C_\epsilon|a|^2 + \epsilon|b|^2$ for $a, b \in \mathbb{R}^d$, $d \ge 1$) inequalities) will be useful

$$|(\phi, \psi)| \le C_\epsilon\|\phi\|^2 + \epsilon\|\psi\|^2 \qquad \forall\phi, \psi \in L_2(\Omega), \tag{4.13}$$

where $\epsilon > 0$ is a arbitrary small positive constant, and C_ϵ is a positive constant depending on ϵ (C_ϵ grows for $\epsilon \to 0$). Note that for $\epsilon = 1/2$ the above inequality becomes

$$(\phi, \psi) \le \frac{1}{2}\|\phi\|^2 + \frac{1}{2}\|\psi\|^2 \qquad \forall\phi, \psi \in L_2(\Omega).$$

4.5.2 Interpolation

We define the interpolation operator $I^h : C(\Omega) \to V^h$ from the space of continuous function to the space V^h of piecewise linear functions such that

$$I^h\phi(\mathbf{x}_k) = \phi(\mathbf{x}_k),$$

for all points $\mathbf{x}_k$ that belong to the finite element mesh. Thus, for a given continuous function ϕ, the interpolation operator produces a piecewise linear function $I^h\phi$ that is equal to the original function at all mesh points. For a function ϕ, the function $I^h\phi$ is called the interpolant of ϕ.

The error between a function $\phi \in H^2(\Omega)$ and its piecewise linear interpolant $I^h\phi \in V^h$ can be estimated from the following "interpolation estimate"

$$\|\phi - I^h\phi\|_{H^1(\Omega)} \le Ch|\phi|_{H^2(\Omega)} \qquad \forall\phi \in H^2(\Omega),$$

where h is the mesh size and C is a fixed positive constant independent of h. We can see that $I^h\phi \to \phi$ as $h \to 0$, i.e. the error gets smaller for finer meshes.

Remark 1 *The proof of the interpolation estimate is based on Taylor's series. Note, that this is similar to finite difference operators, where the error depends on higher-order derivatives of the exact solution.*

4.5.3 Stability

The finite element solution w satisfies

$$(\nabla w, \nabla \phi) + (w, \phi) = (f, \phi) \qquad \forall \phi \in V_h. \tag{4.14}$$

The above equality is valid for any $\phi \in V^h$, thus we can take $\phi = w \in V^h$. Then (4.14) becomes

$$(\nabla w, \nabla w) + (w, w) = (f, w)\,,$$

which is equivalent to

$$\|w\|^2_{H^1(\Omega)} = (f, w).$$

The RHS can be estimated using (4.13) with $\epsilon = 1/2$ as

$$(f, w) \leq \frac{1}{2}\|f\|^2 + \frac{1}{2}\|w\|^2.$$

After combining the above calculations we arrive at

$$\|w\|^2_{H^1(\Omega)} = \|w\|^2 + \|\nabla w\|^2 = \frac{1}{2}\|f\|^2 + \frac{1}{2}\|w\|^2. \tag{4.15}$$

Next we subtract $\frac{1}{2}\|w\|^2$ from the above equation and get

$$\|w\|^2 + \frac{1}{2}\|\nabla w\|^2 \leq \|f\|^2,$$

which is equivalent to

$$\|w\|^2_{H^1(\Omega)} \leq C\|f\|^2,$$

for some (fixed) positive constant C independent of f, w, h.
Thus, if $f \in L_2(\Omega)$, we have just shown that finite element solution w is bounded in $H^1(\Omega)$-norm by a constant that depends on f and Ω (but not on h). Thus, the finite element solution is stable in the $V^h \approx H^1(\Omega)$ space for any h.

Remark 2 *The stability can be also shown for the Poisson equation without the zero order term, i.e., for*

$$(\nabla w, \nabla \phi) = (f, \phi).$$

In this case we get the following instead of (4.15)

$$\|\nabla w\|^2 = C_\epsilon \|f\|^2 + \epsilon \|w\|^2.$$

We can use the fact that for w with zero Dirichlet boundary condition

$$C_2\|w\|_{H^1(\Omega)} \geq \|\nabla w\|^2 \leq C_1\|w\|_{H^1(\Omega)},$$

i.e., the $L_2(\Omega)$ norm of gradient is equivalent to the $H^1(\Omega)$-norm (C_1,C_2 are fixed constants). We can then move the term $\frac{\epsilon}{2}\|w\|^2$ to the LHS for sufficiently small ϵ to obtain stability.

4.5.4 Error estimates

The exact solution u satisfies

$$(\nabla u, \nabla \phi) + (u, \phi) = (f, \phi) \qquad \forall \phi \in H^1(\Omega). \tag{4.16}$$

The finite element solution satisfies

$$(\nabla w, \nabla \phi) + (w, \phi) = (f, \phi) \qquad \forall \phi \in V^h \subset H^1(\Omega). \tag{4.17}$$

We subtract (4.17) from (4.16) and for all $\phi \in V^h$ we have

$$(\nabla e_h, \nabla \phi) + (e_h, \phi) = 0,$$

where $e_h = u - w$. Next, we set $\phi = I^h u - w$ and get

$$\begin{aligned}
&(\nabla e_h, \nabla (I^h u - w)) + (e_h, I^h u - w) \\
&= (\nabla e_h, \nabla (I^h u \mp u - w)) + (e_h, I^h u \mp u - w) \\
&= \|\nabla e_h\|^2 + \|e_h\|^2 + (\nabla e_h, \nabla (I^h u - u)) + (e_h, (I^h u - u)) \\
&= 0
\end{aligned}$$

After putting the last two term to the RHS we obtain

$$\|\nabla e_h\|^2 + \|e_h\|^2 = (\nabla e_h, \nabla (u - I^h u)) + (e_h, (u - I^h u)).$$

Next, we apply the inequality (4.13) with $\epsilon = 1/2$

$$\|\nabla e_h\|^2 + \|e_h\|^2 \leq \frac{1}{2} \left(\|\nabla e_h\|^2 + \|e_h\|^2 \right) + \|u - I^h u\|^2.$$

We move the first two terms on the RHS to the LHS and use the interpolation inequality (4.13) to get

$$\frac{1}{2}(\|\nabla e_h\|^2 + \|e_h\|^2) \leq \|u - I^h u\|^2 \leq C h^2 |u|^2_{H^2(\Omega)}.$$

Which, after taking a square root, proves

$$\|e_h\|_{H^1(\Omega)} \leq C h |u|_{H^2(\Omega)}.$$

We have just shown that if the exact solution $u \in H^2(\Omega)$ (i.e., $|u|_{H^2(\Omega)}$ is bounded) then

$$\|u - w\|_{H^1(\Omega)} \approx O(h).$$