Week 10: PDE revision and the wave equation

Well posedness, stability and the CFL condition, the wave equation as an example

Dr K Clough, Topics in Scientific computing, Autumn term 2023

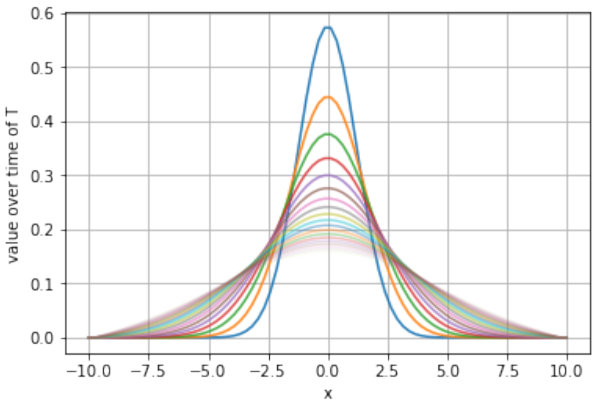
Plan for today

1. Revision of numerical differentiation
2. Revision of PDE types and their properties
3. Problems with PDEs - well posedness
4. Problems with PDEs - Von Neumann stability and the CFL condition
5. Solving second order in time PDEs - solution of the wave equation

Application: solving the heat equation

- In the tutorial you will solve the heat equation using solve_ivp()

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




```python
def calculate_dydt(self, t, current_state) :

    # Just for readability
    dTdt = np.zeros_like(current_state)

    # Now actually work out the time derivatives
    dTdt[:] =  self.alpha * np.dot(self.D2_matrix, current_state)

    # Zero the derivatives at the end for stability
    # (especially important in the pseudospectral method)
    dTdt[0] = 0.0
    dTdt[1] = 0.0
    dTdt[self.N_grid-1] = 0.0
    dTdt[self.N_grid-2] = 0.0

    return dTdt
```

Derivatives - stencil representation

We can see ***finite differencing*** as the convolution of a stencil with the current state vector.

$\Delta x = 0.5$

Position x	0	0.5	1	1.5	2	2.5
Temperature T	0	1	3	2	1	0

4	-8	4

Second derivative stencil

$$\approx \frac{g(x+\Delta x) - 2g(x) + g(x-\Delta x)}{\Delta x^2}$$

d2T/dx2			**-12**			

Derivatives - matrix representation

Here we are using the matrix representation to calculate the time derivative

Position x	0	0.5	1	1.5	2	2.5
Temperature T	0	1	3	2	1	0

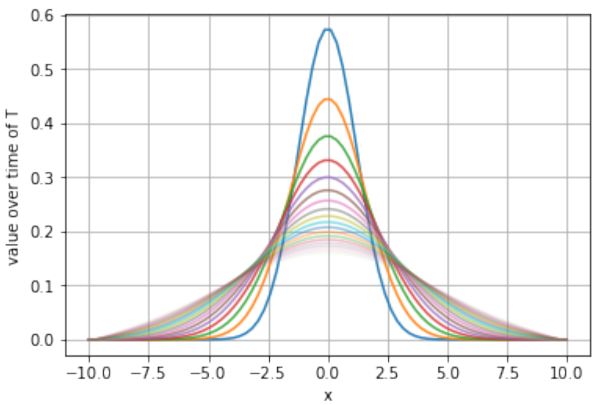
D2Tdx2 **=** ***Matrix D^2*** ***T***

$$\begin{bmatrix} 2 \\ 3 \\ 1 \\ -2 \\ -2 \\ -2 \end{bmatrix} = \begin{bmatrix} X & X & & & & \\ X & X & X & & & \\ & X & X & X & & \\ & & X & X & X & \\ & & & X & X & X \\ & & & & X & X \end{bmatrix} \bullet \begin{bmatrix} 0 \\ 1 \\ 3 \\ 2 \\ 1 \\ 0 \end{bmatrix}$$

Application: solving the heat equation

- In the tutorial you will solve the heat equation using solve_ivp()

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




```
def calculate_dydt(self, t, current_state) :

    # Just for readability
    dTdt = np.zeros_like(current_state)

    # Now actually work out the time derivatives
    dTdt[:] =  self.alpha * np.dot(self.D2_matrix, current_state)

    # Zero the derivatives at the end for stability
    # (especially important in the pseudospectral method)
    dTdt[0] = 0.0
    dTdt[1] = 0.0
    dTdt[self.N_grid-1] = 0.0
    dTdt[self.N_grid-2] = 0.0

    return dTdt
```

Plan for today

1. ~~Revision of numerical differentiation~~
2. Revision of PDE types and their properties
3. Problems with PDEs - well posedness
4. Problems with PDEs - Von Neumann stability and the CFL condition
5. Solving second order in time PDEs - solution of the wave equation

Classification of second order PDEs

Consider the most general second order PDE for 1 dependent variable with 2 independent variables:

$$A\frac{\partial^2 u}{\partial x^2} + 2B\frac{\partial^2 u}{\partial x \partial y} + C\frac{\partial^2 u}{\partial y^2} + D\frac{\partial u}{\partial x} + E\frac{\partial u}{\partial y} + F = 0$$

The equation is classified based on the discriminant $\Delta = B^2 - 4AC$:

$\Delta < 0$ Elliptic

$\Delta = 0$ Parabolic

$\Delta > 0$ Hyperbolic

Example 1: The heat equation

The heat equation, (α is a positive constant, S is any function of u, x and t but not their derivatives)

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} + S$$

What type is this equation?

Example 1: The heat equation

The heat equation is a parabolic equation $\Delta = 0$

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} + S$$

-> A = α, E = -1, B = C = D = 0, F = S

This equation is ***first order in time***, so solutions will evolve in time as exponentials in response to an instantaneous source. The dependence on the ***second derivative in space*** means that it has a tendency to smooth the solution - any bumps in the solution decrease in time.

The positive constant α controls the rate of diffusion of heat.

Example 1: The heat equation

The heat equation is a parabolic equation $\Delta = 0$

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} + S$$

-> A = α, E = -1, B = C = D = 0, F = S

A typical solution has the form:

$$T(x,t) = Ae^{-t/\tau}e^{ikx} \sim Ae^{-t/\tau}\sin(kx)$$

(In general it will be a superposition of many such terms with different k)

Example 2: The wave equation

The wave equation (c is a positive constant, S is any function of u, x and t but not their derivatives)

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

What type is this equation?

Example 2: The wave equation

The wave equation is a hyperbolic equation $\Delta > 0$

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

-> A = c, C = -1, B = D = E = 0, F = S

This equation is ***second order in time***, so solutions will evolve in time with oscillations in response to an instantaneous source. The dependence on the ***second derivative in space*** means that it has a tendency to pull any bumps back towards zero displacement.

Hyperbolic equations have a finite speed of propagation of information - c.

Example 2: The wave equation

The wave equation is a hyperbolic equation $\Delta > 0$

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

-> A = c, C = -1, B = D = E = 0, F = S

A typical solution has the form:

$$T(x,t) = Ae^{i(wt-kx)} \sim A\cos(\omega t)\ \ \sin(kx)$$

(In general it will be a superposition of many such terms with different k)

Example 3: Poisson's equation

The Poisson equation (f is any function of u, x and t but not their derivatives)

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f$$

What type is this equation?

Example 3: Poisson's equation

The Poisson equation is an elliptic equation $\Delta < 0$

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f$$

-> A = 1, C = 1, B = D = E = 0, F = -f

This equation is ***second order in both dimensions, which are usually thought of as two spatial directions*** (for reasons we will discuss next). If the source f is zero it is called Laplace's equation, and for zero boundary conditions the solution is a constant. A non zero source creates a displacement or bump in the solution.

Elliptic equations have an infinite speed of propagation of information.

Example 3: Poisson's equation in 1D

The Poisson equation (f is any function of u, x and t but not their derivatives)

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

What type is this equation?

Example 4: Poisson's equation in 1D

The Poisson equation (f is any function of u, x and t but not their derivatives)

$$\frac{d^2u}{dx^2} = f$$

Trick question! This is just an ODE like we studied before as there is only one independent variable!

Example 5: Katy's equation

Katy's equation (f is any function of u, x and t but not their derivatives)

$$\frac{\partial^2 u}{\partial t^2} + (t - 10)\frac{\partial^2 u}{\partial y^2} = f$$

What type is this equation?

Example 5: Katy’s equation

Katy’s equation (f is any function of u, x and t but not their derivatives)

$$\frac{\partial^2 u}{\partial t^2} + (t - 10)\frac{\partial^2 u}{\partial y^2} = f$$

This equation changes character at t=10 - before it is hyperbolic and after it is elliptic.

A system of PDEs can be of mixed type (e.g. Navier Stokes is mixed parabolic/hyperbolic) and they can change type at different points in space and time.

Plan for today

1. ~~Revision of numerical differentiation~~
2. ~~Revision of PDE types and their properties~~
3. Problems with PDEs - well posedness
4. Problems with PDEs - Von Neumann stability and the CFL condition
5. Solving second order in time PDEs - solution of the wave equation

Well posed problems - very active area of QMUL research!

- QMUL Maths is one of the leading places for solving issues of well-posedness.
- e.g. Prof Claudia Garetto of geometry, analysis and gravitation centre

On the well-posedness of weakly hyperbolic equations with time-dependent coefficients

Claudia Garetto [1], Michael Ruzhansky [2]

Show more

+ Add to Mendeley Share Cite

https://doi.org/10.1016/j.jde.2012.05.001 Get rights and content

Under an Elsevier user license open archive

Abstract

In this paper we analyse the Gevrey well-posedness of the Cauchy problem for weakly hyperbolic equations of general form with time-dependent coefficients. The results involve the order of lower order terms and the number of multiple roots. We also derive the corresponding well-posedness results in the space of Gevrey Beurling ultradistributions.

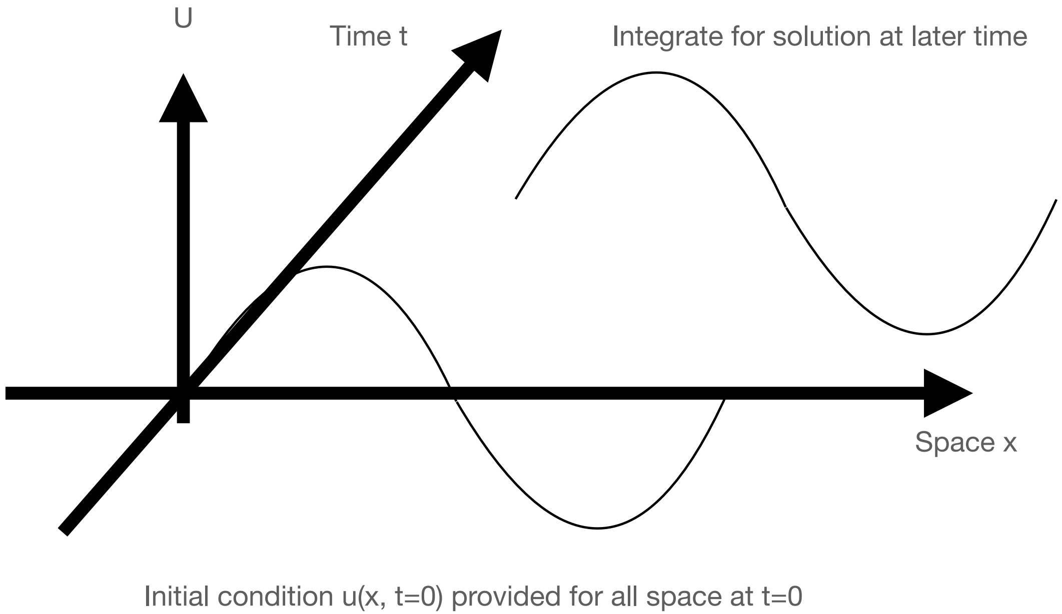
Well posed problems

- An initial value / Cauchy problem is well posed if:
  - A solution exists
  - The solution is unique
  - The solution depends continuously on the initial data

What is an initial value problem/Cauchy problem?

Initial value problem

- One of the independent variables is thought of as “time” (doesn’t have to actually ***be*** time)
- Boundary value is provided as a value of the function at some (arbitrary) time t=0
- Full solution is found by integrating in time
- We will see an alternative (boundary value solution via relaxation) next week



Well posed problems

- An initial value / Cauchy problem is well posed if:
  - A solution exists
  - The solution is unique
  - The solution depends continuously on the initial data

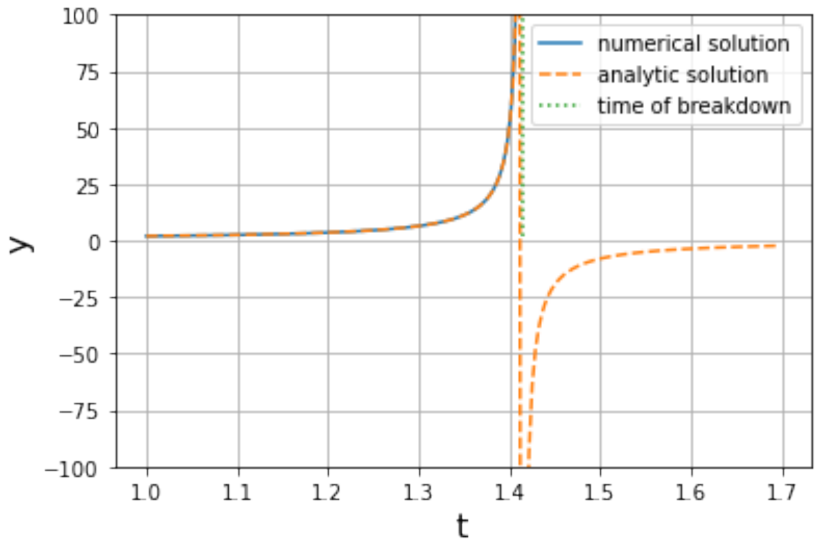
What did it mean for the solution to depend continuously on the initial data?

Well posed problems

Recall for ODEs:

- If $x_1(0) = a, x_2(0) = a + \delta$ it tells us that the solution changes by an amount that is bounded by $\delta\ e^{Lt}$ where L is some constant value - this is the meaning of "***depends continuously on the initial data***".

- We had the example that blows up at a value that depends on the initial conditions, so that a small change results in a change that is not bounded by an exponential



Well posed problems

- An initial value / Cauchy problem is well posed if:
  - A solution exists
  - The solution is unique
  - The solution depends continuously on the initial data

How can a solution not exist?

Well posed problems

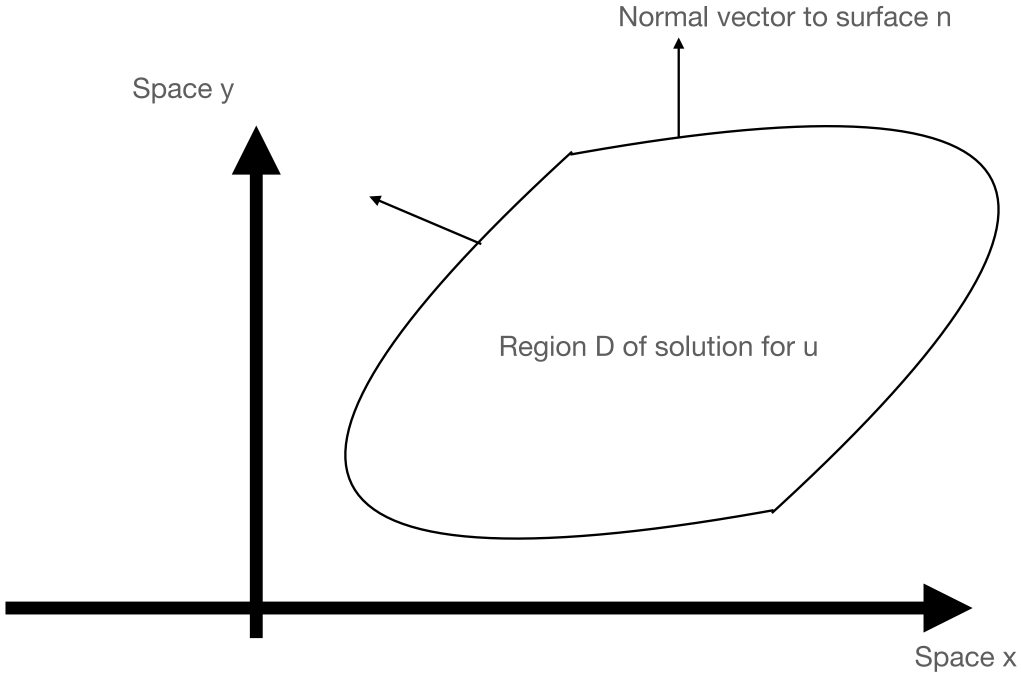
For Laplace's equation

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$$

with fixed boundary conditions

$$\nabla u \,.\, n = g(x, y) \quad (x, y) \in \partial D$$

No solution exists if $\displaystyle\int_{\partial D} g(x, y)\ ds \neq 0$



A nice detailed explanation is here: https://youtu.be/BmTFbUAOeec?si=22bdWktp55xLcT3s

Well posed problems

- An initial value / Cauchy problem is well posed if:
  - A solution exists
  - The solution is unique
  - The solution depends continuously on the initial data

How can a solution not be unique?

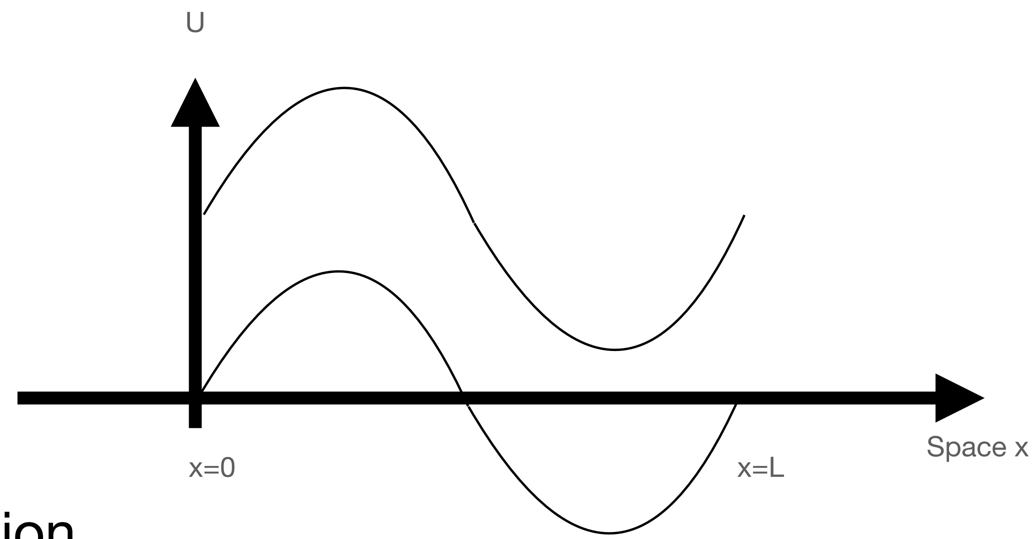
Well posed problems

Consider Poisson's equation

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f$$

with periodic boundary conditions
$u(x = L) = u(x = 0)$

Then for any solution $u(x, t)$ the solution $\bar{u}(x, t) = u(x, t) + C$ with C a constant is also a solution.



Well posed problems

- Theorems in mathematics guarantee the (local) well-posedness of linear and quasi-linear* strongly hyperbolic* and parabolic PDEs.
- Elliptic PDEs do not admit a well-posed IVP. This does not (necessarily) mean they cannot be solved, just that another method may be required.
- When in a correct numerical implementation one increases the resolution and the solution blows up faster, that usually implies an ill-posed initial value problem.

*We will discuss the exact meaning of these terms next week.
For now just think of hyperbolic and parabolic equations as generally ok.

Well posed problems - why elliptic equations fail as an initial value problem

Consider Laplace's equation but treat one of the directions as a "time":

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

And propose a wave like solution

$$u(x, t) = \exp(i[\omega t - kx])$$

Then

$$-\omega^2 u - k^2 u = 0 \quad \implies \omega = \pm\, i\,|\,k\,| \quad \implies u(x, t) = A \;\; \exp(\,|\,k\,|\,t + ikx) + \ldots$$

Which blows up exponentially at a faster rate for higher k (= shorter wavelengths)

Plan for today

1. ~~Revision of numerical differentiation~~
2. ~~Revision of PDE types and their properties~~
3. ~~Problems with PDEs - well posedness~~
4. Problems with PDEs - Von Neumann stability and the CFL condition
5. Solving second order in time PDEs - solution of the wave equation

Von Neumann stability analysis and the CFL condition (Courant Friedrich Lewy)

- Like for ODEs, numerical schemes for PDEs can be unstable, and they have to be analysed for each PDE and PDE scheme separately

- For an initial value problem, this usually results in a “**CFL condition**” on the time step of the form:

$$\Delta t = \lambda \Delta x \qquad \text{for hyperbolic equations}$$
$$\Delta t = \lambda \Delta x^2 \qquad \text{for parabolic equations}$$

- The main method to determine the CFL number λ is called the Von Neumann stability analysis. It is a is ***necessary but not sufficient*** condition for stability.

Von Neumann stability analysis and the CFL condition

Method:

1. Calculate for the given numerical scheme the amplification factor between timesteps - assume that this is the same for the solution and the error

$$\Lambda = \frac{u_i^{n+1}}{u_i^n}$$

2. Make an assumption about the form of the solution

3. Require $|\Lambda| \leq 1$ for the solution error to not be amplified, which gives rise to a condition on Δt in terms of Δx.

Von Neumann stability analysis

e.g. Euler update for the heat equation, using 3 point stencil
(will do in lecture, but not examinable):

1. Calculate for the given numerical scheme the amplification factor between timesteps

$$T_i^{n+1} = T_i^n + \Delta t \ \frac{\partial T_i^n}{\partial t}$$

$$\implies T_i^{n+1} = T_i^n + \alpha \Delta t \ \frac{\partial^2 T_i^n}{\partial x^2}$$

$$\implies T_i^{n+1} \approx T_i^n + \alpha \frac{\Delta t}{(\Delta x)^2} (T_{i-1}^n - 2T_i^n + T_{i+1}^n)$$

Von Neumann stability analysis

e.g. Euler update for the heat equation, using 3 point stencil (will do in lecture, but not examinable):

2. Assume the form of solution for the heat equation

$$T(t, x) = E_k(t)e^{ikx}$$

$$\implies E_i^{n+1}e^{ikx} \approx E_i^n e^{ikx}\left[1 + \alpha\frac{\Delta t}{(\Delta x)^2}\left(e^{-ik\Delta x} - 2 + e^{ik\Delta x}\right)\right]$$

$$\implies E_i^{n+1}e^{ikx} \approx E_i^n e^{ikx}\left[1 - 4\alpha\frac{\Delta t}{(\Delta x)^2}\sin^2\left(\frac{k\Delta x}{2}\right)\right]$$

Von Neumann stability analysis

e.g. Euler update for the heat equation, using 3 point stencil (will do in lecture, but not examinable):

3. The amplification factor should have a magnitude of less than 1

$$\implies E_i^{n+1} = E_i^n \left[1 - 4\alpha \frac{\Delta t}{(\Delta x)^2} \sin^2\left(\frac{k\Delta x}{2}\right)\right]$$

$$\implies \Lambda_{min} = \left|1 - 4\alpha \frac{\Delta t}{(\Delta x)^2}\right| \leq 1$$

$$\implies \Delta t \leq \frac{1}{2\alpha}(\Delta x)^2$$ CFL number is $\frac{1}{2\alpha}$

Von Neumann stability analysis and the CFL condition (Courant Friedrich Lewy)

- "**CFL condition**" on the time step of the form:

$$\Delta t = \lambda \Delta x \qquad \text{for hyperbolic equations}$$
$$\Delta t = \lambda \Delta x^2 \qquad \text{for parabolic equations}$$

- By physical arguments, we should expect $\lambda \leq 1/c$ (wave eqn) or $\lambda \leq 1/\alpha$ (heat eqn) since these constants determine the speed of propagation (if we take too big timesteps we don't respect causality).

- In practise - we can usually just use trial and error to find how high/low λ can be before the code becomes numerically unstable.

Plan for today

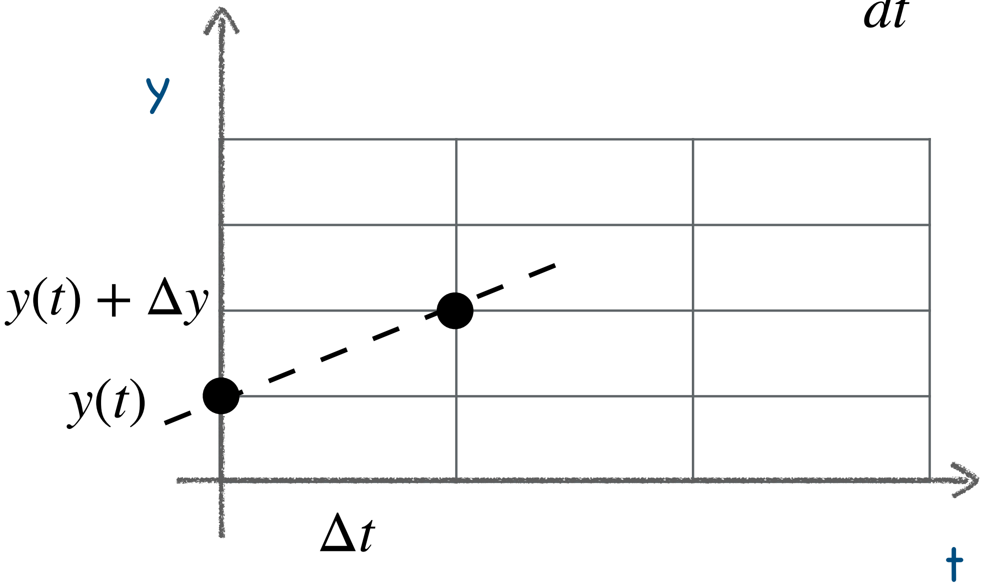
1. ~~Revision of numerical differentiation~~
2. ~~Revision of PDE types and their properties~~
3. ~~Problems with PDEs - well posedness~~
4. ~~Problems with PDEs - Von Neumann stability and the CFL condition~~
5. Solving second order in time PDEs - solution of the wave equation

Recall: How do I integrate second order ODEs numerically?

$$\frac{d^2y}{dt^2} - \frac{dy}{dt} + f(y,t) = 0$$

$$\begin{cases} \dfrac{dv}{dt} - v + f(y,t) = 0 \\ \dfrac{dy}{dt} = v \end{cases}$$

1. Decompose the second order equation into two first order ones



$$\Delta v = \Delta t \left(v - f(y,t) \right)$$

$$\Delta y = v \ \Delta t$$

2. Solve as a dimension 2 first order system

Solving second order PDEs - the wave equation

Consider the wave equation for u:

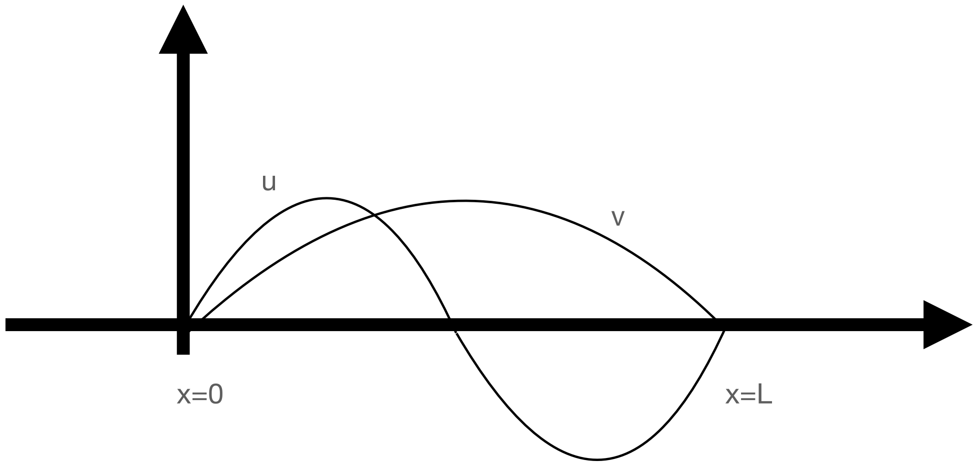
$$\frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} = 0$$

And define the time derivative to be

$$v(x,t) = \frac{\partial u}{\partial t}$$

Then we solve the coupled system:

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



Wave equation - matrix representation

Recall that we can also represent this in matrix form: $\frac{\partial v}{\partial t} = \frac{\partial^2 u}{\partial x^2}$ $\frac{\partial u}{\partial t} = v$

dv/dt		***Matrix D^2***							***u***
2	=	X	x					•	0
3		1	-2	1					1
1	=		1	-2	1				3
-2				1	-2	1			2
-2					1	-2	1		1
-2						x	x		0

All blank entries zero

Wave equation - matrix representation

Recall that we can also represent this in matrix form: $\frac{\partial v}{\partial t} = \frac{\partial^2 u}{\partial x^2}$ $\frac{\partial u}{\partial t} = v$

$$
\underset{\textbf{du/dt}}{\begin{bmatrix} 2 \\ 3 \\ 1 \\ -2 \\ -2 \\ -2 \end{bmatrix}}
=
\underset{\textbf{Matrix I}}{\begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \end{bmatrix}}
\bullet
\underset{\textbf{v}}{\begin{bmatrix} 0 \\ 1 \\ 3 \\ 2 \\ 1 \\ 0 \end{bmatrix}}
$$

All blank entries zero

Wave equation - state vector in python

Need to unpack and repack the state vector in python.

Some useful commands:

```
u0 = get_y_test_function(x_values)
v0 = np.zeros_like(u0)
y0 = np.concatenate([u0,v0])
```

```
# Just for readability
[u,v] = np.array_split(current_state, 2)
dydt = np.zeros_like(current_state)
dudt, dvdt = np.array_split(dydt, 2)
```

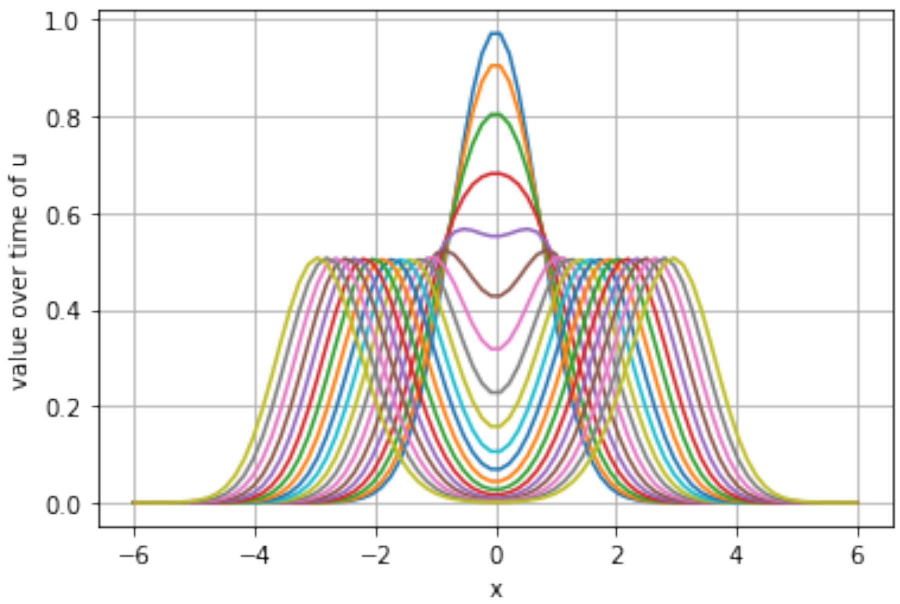
```
dudt[:] =  v
```

y

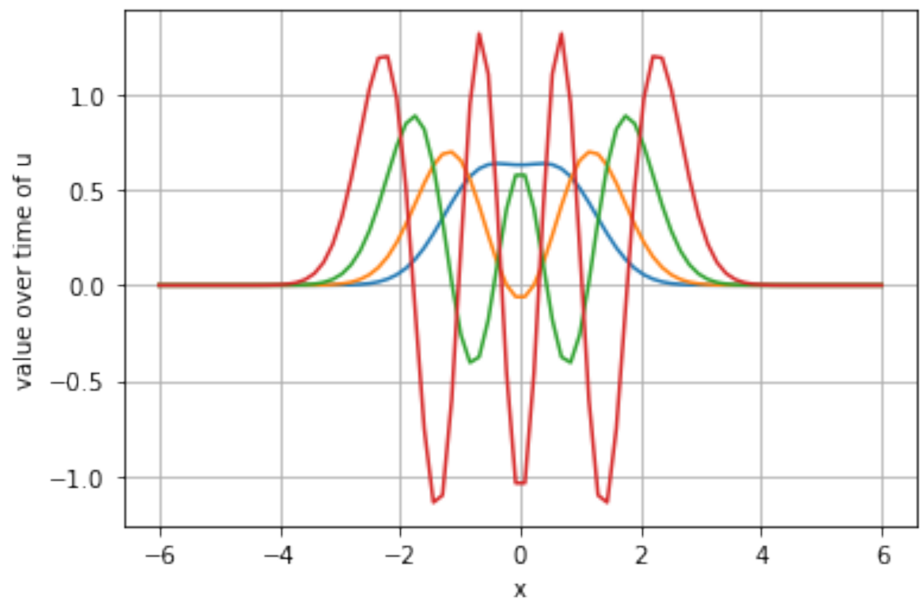
	2
	3
	1
u	-2
	-2
	-2
	0
	1
v	3
	2
	1
	0

Wave equation - tutorial

In the tutorial you will update the heat equation code from last week for the wave equation, and test the CFL condition.



CFL condition respected



CFL condition not respected

Plan for today

1. Revision of numerical differentiation
2. Revision of PDE types and their properties
3. Problems with PDEs - well posedness
4. Problems with PDEs - Von Neumann stability and the CFL condition
5. Solving second order in time PDEs - solution of the wave equation