Computational Physics

Part II: Numerical Solution of PDEs

Ville Havu, Senior University Lecturer

Topics is Part 2

- General tools for solving partial differential equations
  - Spatial discretization
  - Linear algebra
- Methods for different types of PDEs
  - Elliptic, parabolic and hyperbolic
  - Poisson, heat and wave equation

Today’s Topics

- Why PDEs and how they arise in describing physical phenomena
- From infinite to finite dimensions
- Finite difference formulae

Why Partial Differential Equations?

- Omnipresent in physics & engineering:
  - Poisson $-\Delta u = f$ Electrostatics: $\left.\begin{array}{l}\nabla \cdot \vec{E} = \dfrac{\rho}{\epsilon_0} \\ \vec{E} = -\nabla u\end{array}\right\} \Rightarrow \boxed{-\nabla \cdot \nabla u = -\Delta u = \dfrac{\rho}{\epsilon_0}}$
  - Heat $\dfrac{\partial u}{\partial t} = \Delta u$ Diffusion / Heat flow: $\left.\begin{array}{l}\dfrac{d}{dt}\displaystyle\int_V u\, d\mathbf{r} = -\int_{\partial V} \vec{j} \cdot d\vec{S} = -\int_V \nabla \cdot \vec{j}\, d\mathbf{r} \\ \vec{j} = -k\nabla u\end{array}\right\} \Rightarrow \displaystyle\int_V \frac{\partial}{\partial t} u\, d\mathbf{r} = \int_V \nabla \cdot k\nabla u\, d\mathbf{r} \Rightarrow \boxed{\frac{\partial}{\partial t} u = k\Delta u}$
  - Wave $\dfrac{\partial^2 u}{\partial t^2} = \Delta u$ Newton's II law for a perturbation: $\left.\begin{array}{l}\rho\dfrac{\partial^2 u}{\partial t^2} = \nabla \cdot \vec{F} \\ \vec{F} = K\nabla u\end{array}\right\} \Rightarrow \boxed{\dfrac{\partial^2 u}{\partial t^2} = \dfrac{K}{\rho}\Delta u}$
  - Schrödinger $-\dfrac{1}{2}\Delta\psi + V\psi = E\psi$ Quantum mechanical particle in the potential *V*

Discuss: What needs to be addressed when finding (numerical) solutions to PDEs?

Why Partial Differential Equations?

- Omnipresent in physics & engineering:
  - Poisson $-\Delta u = f$
  - Heat $\frac{\partial u}{\partial t} = \Delta u$
  - Wave $\frac{\partial^2 u}{\partial t^2} = \Delta u$
  - Schrödinger $-\frac{1}{2}\Delta\psi + V\psi = E\psi$

Tasks:

- PDEs have infinite dimension
  → must find a finite-dimensional representations
- Linear operators lead to linear equations
  → need matrix algebra
- Time dependent problems
  → need time stepping

Need both spatial and temporal discretization

Boundary and Initial Conditions

- Basic boundary conditions:
  - Dirichlet $u = 0$ ”zero potential”
  - Neumann $\frac{\partial u}{\partial n} = \vec{n} \cdot \nabla u = 0$

    ”zero normal component” $\vec{n} \cdot \vec{E} = 0$
  - Robin $\frac{\partial u}{\partial n} = \alpha u + \beta g$

    ”flux depends on the value” $k \frac{\partial T}{\partial n} = \alpha(T - T_0)$

Initial conditions:

- Heat equation $\frac{\partial u}{\partial t} = \Delta u$ set $u(t = 0) = u_0$
- Wave equation $\frac{\partial^2 u}{\partial t^2} = \Delta u$

  set $u(t = 0) = u_0$ and $\frac{\partial u}{\partial t}(t = 0) = v_0$

Step 1: Spatial discretization

- Take first 1-D: $-\frac{\partial^2}{\partial x^2} u \to$

- Simple setting: Interval 0…L, uniform grid $x_i = i \cdot h$, $h = L/N$
- Expand

$$u((i-1)h) = u(ih) - u'(ih)h + \frac{1}{2}u''(ih)h^2 + \ldots$$

$$u((i+1)h) = u(ih) + u'(ih)h + \frac{1}{2}u''(ih)h^2 + \ldots$$

Add and reorganize

$$-u''(ih) = \frac{1}{h^2}\left[2u(ih) - u((i+1)h) - u((i-1)h)\right] + \ldots$$

$$\boxed{-\frac{\partial^2}{\partial x^2} u \approx \frac{1}{h^2}\left(2u_i - u_{i+1} - u_{i-1}\right)}$$

Famous 3-point stencil

Step 1: Spatial discretization

- How do you know if this is any good: $-\frac{\partial^2}{\partial x^2} u \approx \frac{1}{h^2}\left(2u_i - u_{i+1} - u_{i-1}\right)$
- Spectral testing:

Continuous:

$$-u'' = \lambda u, \quad u(0) = u(L) = 0$$

$$u(x) = C_1 \sin\left(\sqrt{\lambda} x\right) + C_2 \cos\left(\sqrt{\lambda} x\right)$$

$$\begin{cases} u(0) = 0 \Rightarrow C_2 = 0 \\ u(L) = 0 \Rightarrow \lambda = \lambda_k = \left(\frac{k\pi}{L}\right)^2 \end{cases}$$

Discrete:

Try: $u_i^k = \sin\left(\frac{k\pi}{L} ih\right)$

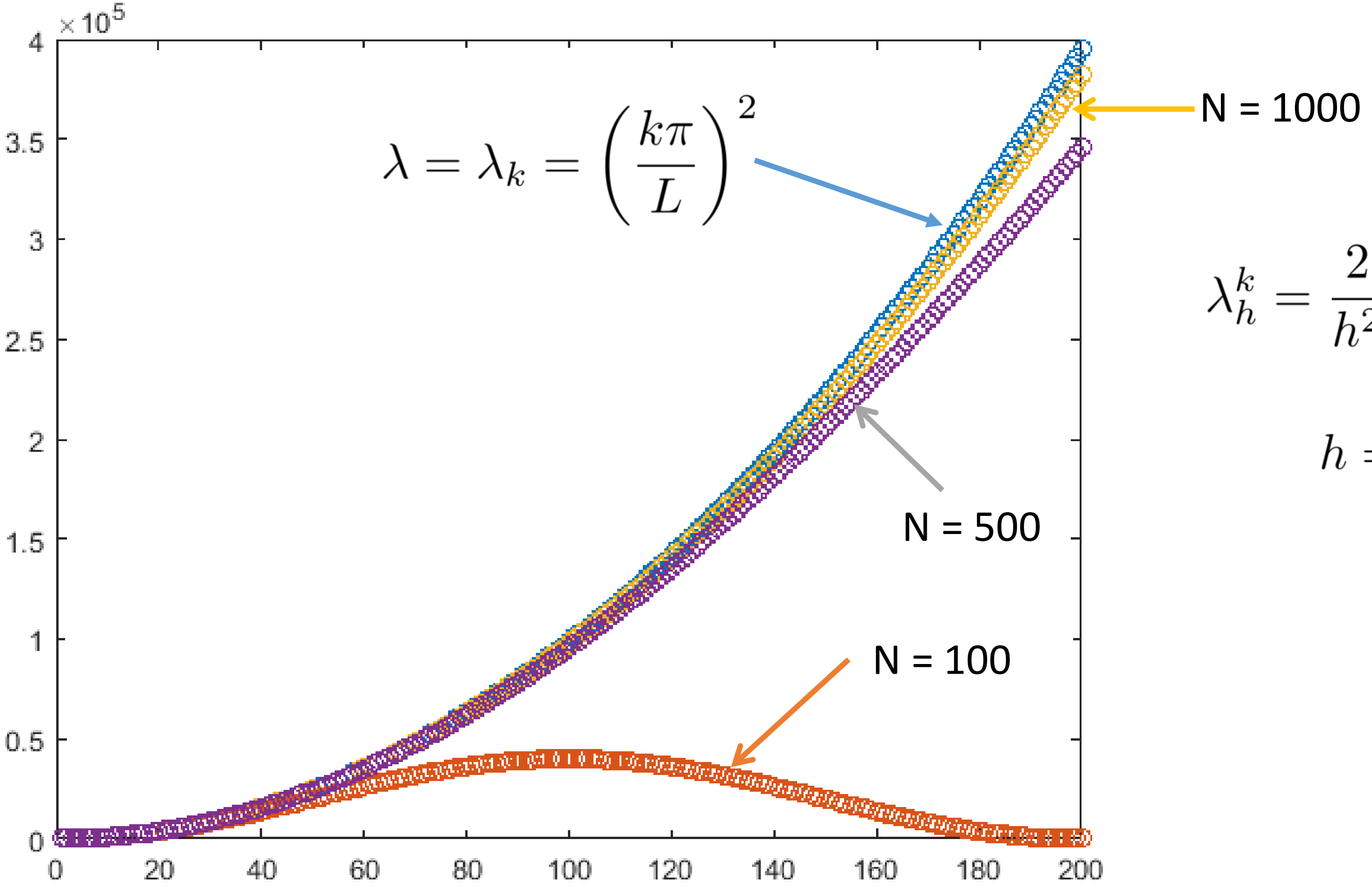
$$\Downarrow$$

$$\frac{1}{h^2}\left(2u_i - u_{i+1} - u_{i-1}\right) = \frac{2}{h^2}\left[1 - \cos\left(\frac{k\pi}{L} h\right)\right] u_i$$

$$\Rightarrow \quad \lambda_h^k = \frac{2}{h^2}\left[1 - \cos\left(\frac{k\pi}{L} h\right)\right] \approx \lambda_k$$

Ok, the approximation looks reasonable

Spatial discretization: Spectral testing



$$\lambda_h^k = \frac{2}{h^2}\left[1 - \cos\left(\frac{k\pi}{L}h\right)\right] \approx \lambda_k$$

$$h = L/N = 1/N$$

Ok, the approximation looks still reasonable

Another sanity check: Polynomials

$$-\frac{\partial^2}{\partial x^2}u \approx \frac{1}{h^2}\left(2u_i - u_{i+1} - u_{i-1}\right)$$

- What happens when *u(x) = ax²* + *bx* + *c* ?

$$\frac{1}{h^2}\left(2u_i - u_{i+1} - u_{i-1}\right) = -2a$$ and even when $u(x) = x^3$ ⇒ $$\frac{1}{h^2}\left(2u_i - u_{i+1} - u_{i-1}\right) = -6ih$$

But when $u(x) = x^4$ ⇒ $$\frac{1}{h^2}\left(2u_i - u_{i+1} - u_{i-1}\right) = \left(-12\, i^2 - 2\right) h^4$$

Order	OK?
1	Yes
2	Yes
3	Yes
≥4	No

More dimensions

- Since in 1D: $-\frac{\partial^2}{\partial x^2}u \approx \frac{1}{h^2}\left(2u_i - u_{i+1} - u_{i-1}\right)$
- 2D becomes easy:

$$-\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)u \approx \frac{1}{h^2}\left(2u_{i,j} - u_{i+1,j} - u_{i-1,j} + 2u_{i,j} - u_{i,j+1} - u_{i,j-1}\right)$$

$$= \frac{1}{h^2}\left(4u_{i,j} - u_{i+1,j} - u_{i-1,j} - u_{i,j+1} - u_{i,j-1}\right)$$

The famous 5-point stencil: $\begin{bmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix}$

3D is then crystal clear (7-point stencil):

$$-\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}\right)u \approx \frac{1}{h^2}\left(6u_{i,j,k} - u_{i+1,j,k} - u_{i-1,j,k} - u_{i,j+1,k} - u_{i,j-1,k} - u_{i,j,k+1} - u_{i,j,k-1}\right)$$

Finite difference formulae from polynomials

Fit the data points with a polynomial, take the derivative of the polynomial and evaluate that at desired point.

$$\tilde{u}(x) = \frac{(x-x_1)(x-x_2)}{2h^2}u(x_0) - \frac{(x-x_0)(x-x_2)}{h^2}u(x_1) + \frac{(x-x_0)(x-x_1)}{2h^2}u(x_2) \Rightarrow \tilde{u}''(x) = \frac{1}{h^2}u(x_0) - \frac{2}{h^2}u(x_1) + \frac{1}{h^2}u(x_2)$$

Forward difference formula

$$x_0 = ih,\ x_1 = (i+1)h,\ x_2 = (i+2)h \Rightarrow -u''(ih) = \frac{1}{h^2}\left[2u((i+1)h) - u(ih) - u((i+2)h)\right]$$

Backward difference formula

$$x_0 = (i-2)h,\ x_1 = (i-1)h,\ x_2 = ih \Rightarrow -u''(ih) = \frac{1}{h^2}\left[2u((i-1)h) - u(ih) - u((i-2)h)\right]$$

Central difference formula

$$x_0 = (i-1)h,\ x_1 = ih,\ x_2 = (i+1)h \Rightarrow -u''(ih) = \frac{1}{h^2}\left[2u(ih) - u((i+1)h) - u((i-1)h)\right]$$

Finite difference formulae from polynomials

Fit the data points with a polynomial, take the derivative of the polynomial and evaluate that at desired point.

Three-point formula for the 2nd derivative:

```
Nn = 1; (*Number of points at both directions*)
Print["Total number of points is ", 2 Nn + 1]

Total number of points is 3

Dd = 2; Print["Derivative of the order ", Dd]

Derivative of the order 2

g[x_] := (*Interpolate a polynomial*)
 Evaluate[InterpolatingPolynomial[
   Table[{x0 + i, y[i]}, {i, -Nn, Nn}], x]]

gp[x_] := Evaluate[D[g[x], {x, Dd}]]
 (*Derivative of the interpolating polynomial*)

Simplify[
 Expand[Collect[gp[x0 + 0],
   Table[y[i], {i, -Nn, Nn}]]]]

y[-1] - 2 y[0] + y[1]
```

Five-point formula for the 2nd derivative:

```
Nn = 2; (*Number of points at both directions*)
Print["Total number of points is ", 2 Nn + 1]

Total number of points is 5

Dd = 2; Print["Derivative of the order ", Dd]

Derivative of the order 2

g[x_] := (*Interpolate a polynomial*)
 Evaluate[InterpolatingPolynomial[
   Table[{x0 + i, y[i]}, {i, -Nn, Nn}], x]]

gp[x_] := Evaluate[D[g[x], {x, Dd}]]
 (*Derivative of the interpolating polynomial*)

Simplify[
 Expand[Collect[gp[x0 + 0],
   Table[y[i], {i, -Nn, Nn}]]]]
```

$$\frac{1}{12}\left(-y[-2] + 16\, y[-1] - 30\, y[0] + 16\, y[1] - y[2]\right)$$

Seven-point formula for the 3rd derivative: $\frac{1}{8}\left(y[-3] - 8\, y[-2] + 13\, y[-1] - 13\, y[1] + 8\, y[2] - y[3]\right)$

Convergence of the finite difference formulae

Test the central difference formula for exp(x) at x=0

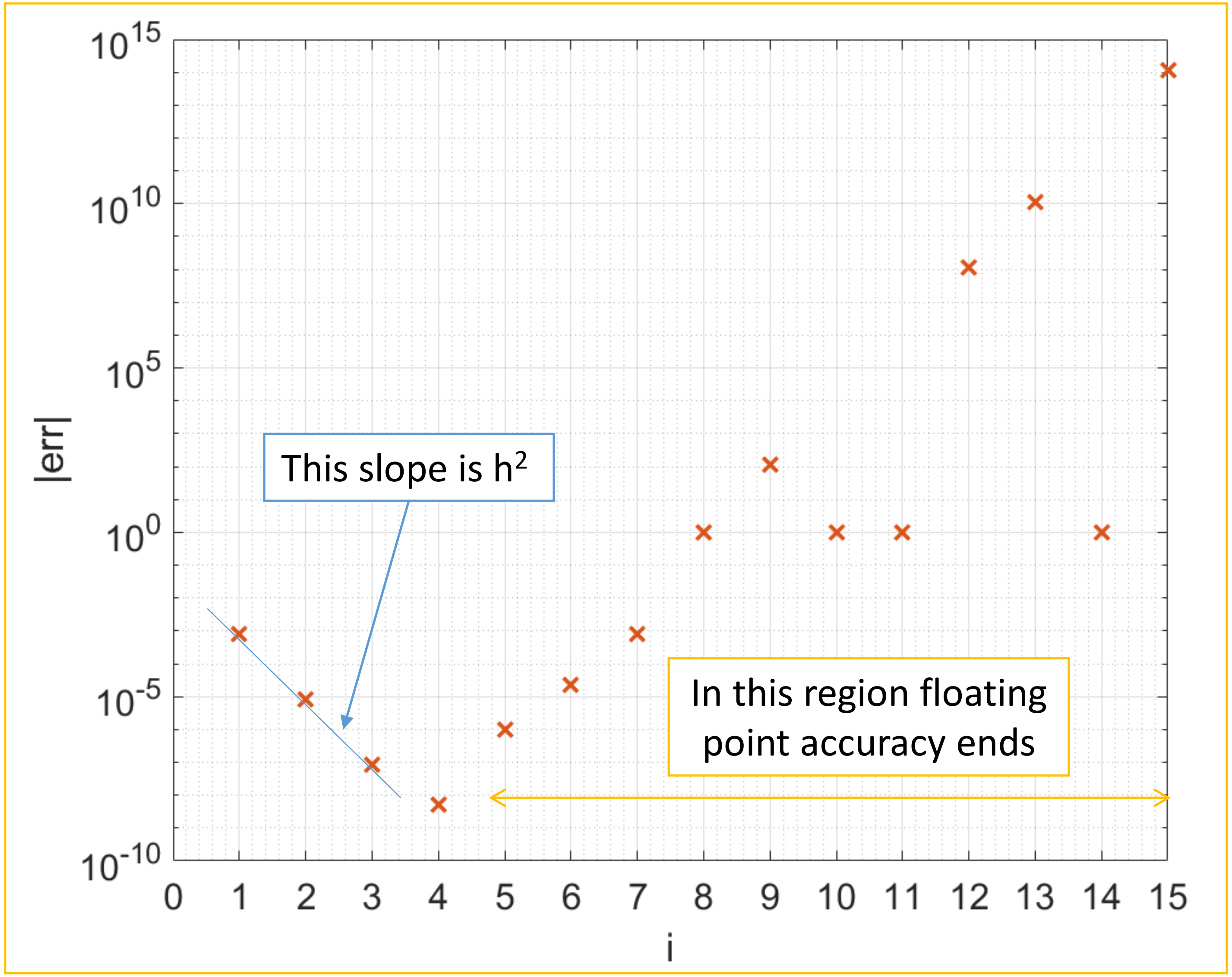
Exact: $$\frac{d^2}{dx^2}e^x = e^x \quad x = 0 \Rightarrow e^0 = 1$$

Approximation:

$$e^0 \approx \frac{1}{h^2}\left[e^h - 2 + e^{(-h)}\right], \quad h = 10^{-i},\ i = 1, \ldots, N$$

Measure of error:

$$\text{err} = \frac{1}{h^2}\left[e^h - 2 + e^{(-h)}\right] - 1$$



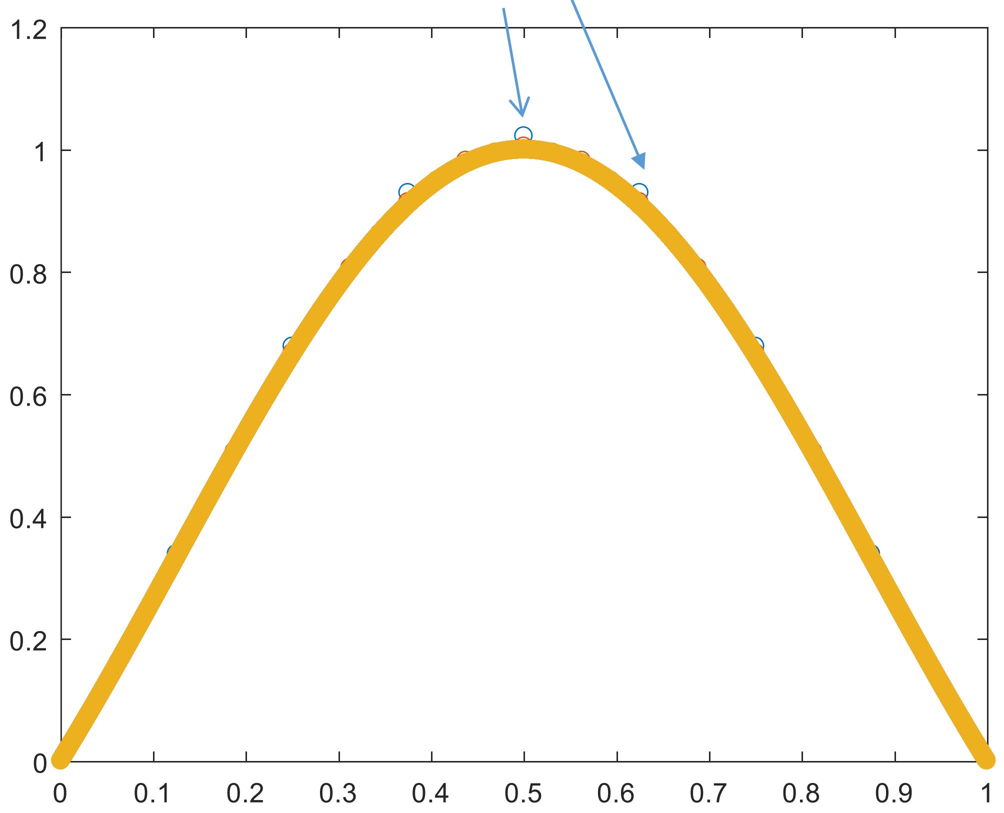
Solving the Poisson equation: 1D

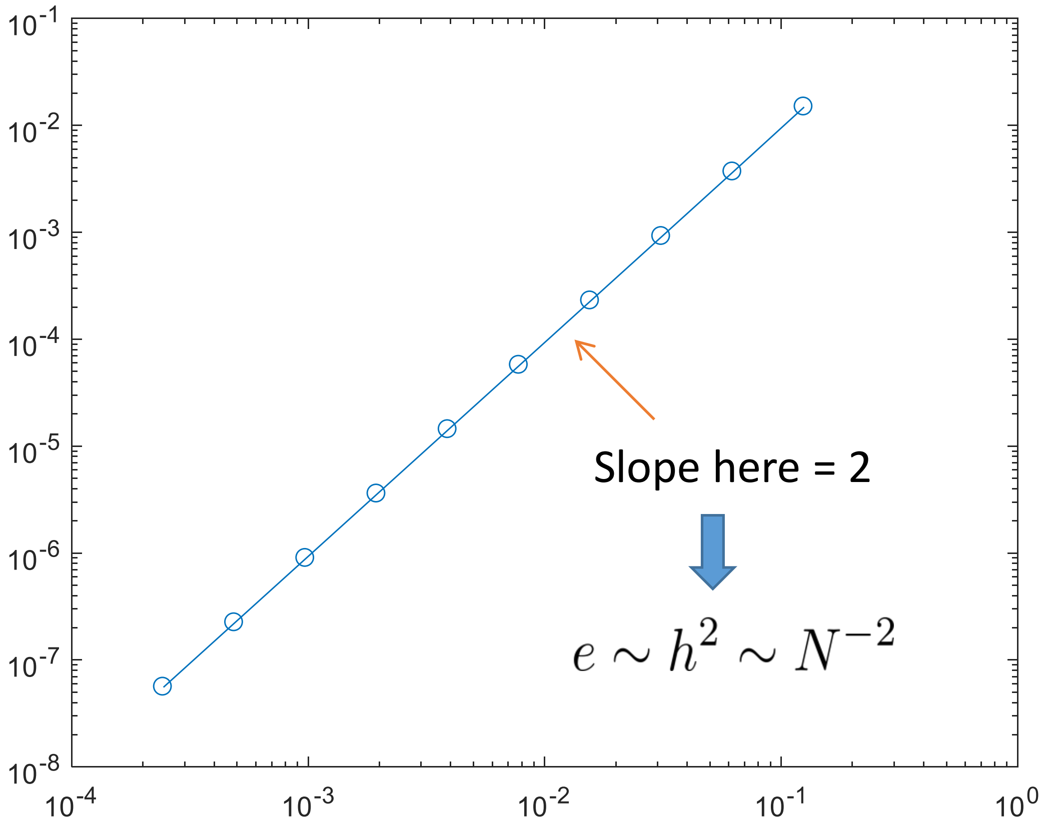
Easy case: $u(x) = \sin\left(\frac{\pi x}{L}\right) \exp\left(-(x - L/2)^2\right)$

$$N = 2^k, \quad k = 3 \dots 12$$

Error as: $e = \sqrt{h \sum_i (u(x_i) - u_i)^2}$

Only the sparsest grid differs noticeably



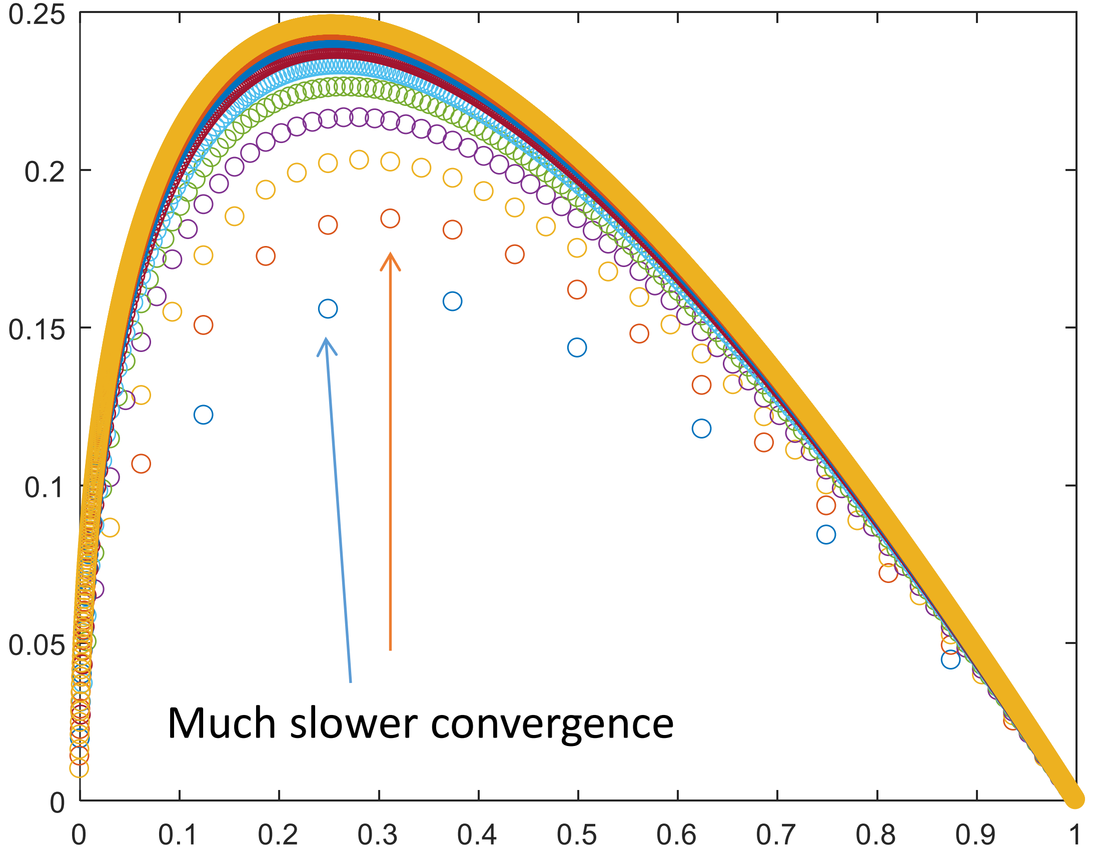


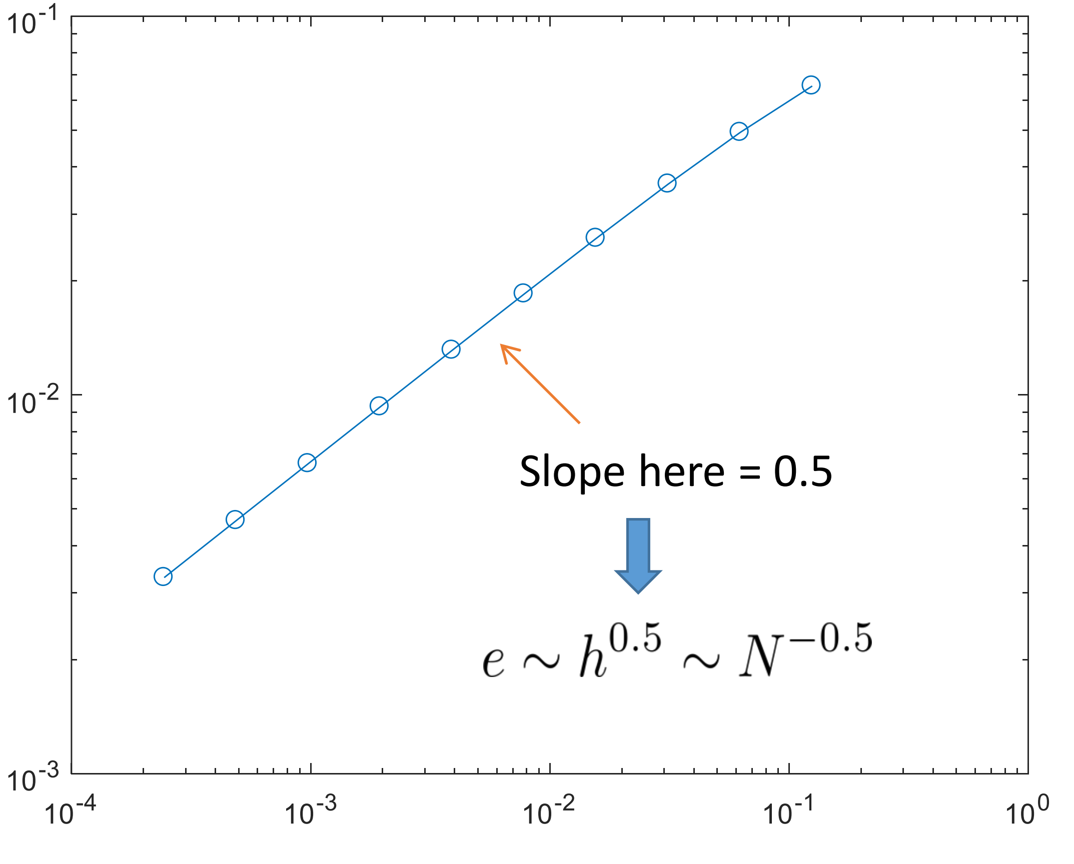
Solving the Poisson equation: 1D

Tougher case: $u(x) = \sqrt{x} - x$

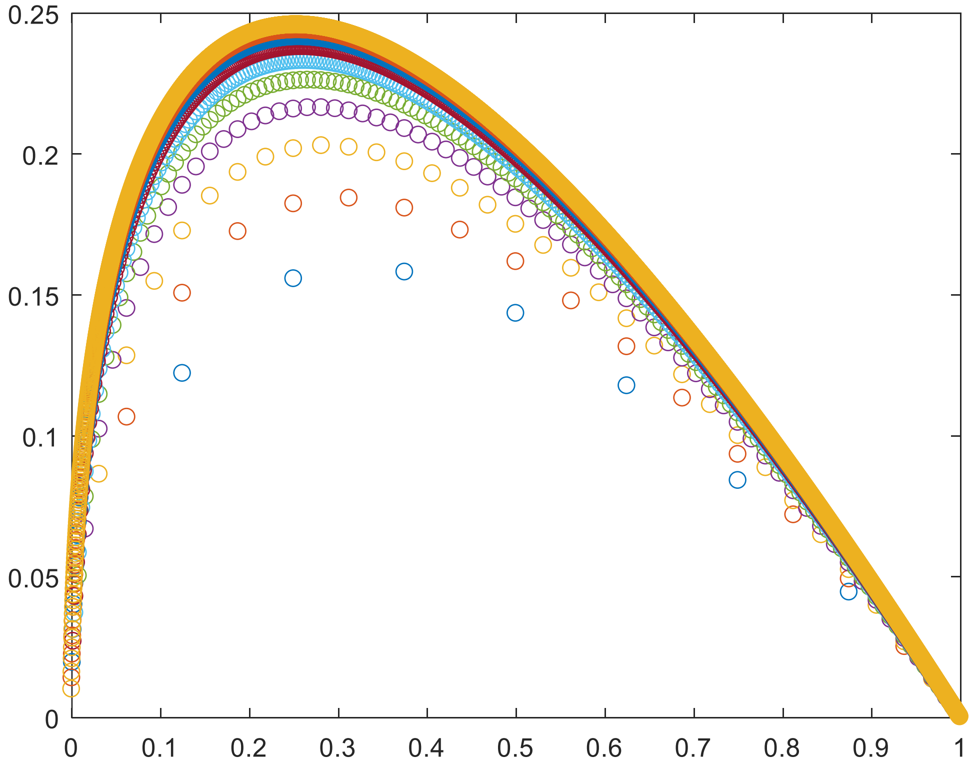
$$N = 2^k, \quad k = 3 \dots 12$$

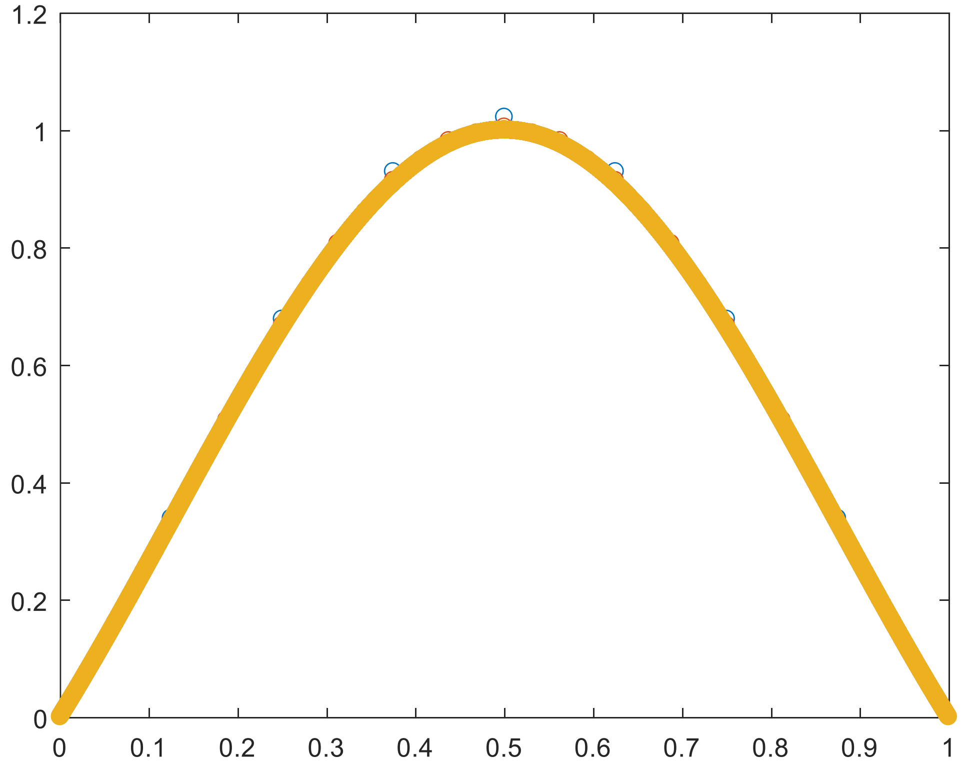
Error as: $e = \sqrt{h \sum_i (u(x_i) - u_i)^2}$





Finite Difference: Convergence





Discuss: What is the difference between left and right?

From classical to quantum physics

Classical Hamiltonian of a particle (mass m) in external potential V:

$$H = \frac{p^2}{2m} + V(x)$$

that is the energy of the system, kinetic and potential energy.

We can get to quantum mechanics simply by:

$$p \to -\mathrm{i}\hbar \frac{\partial}{\partial x}$$

so that the quantum Hamiltonian is an operator that looks like:

$$\mathcal{H} = -\frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2} + V(x)$$

This acts on the wave function, and gives the energy: $\mathcal{H}\psi(x) = E\psi(x)$.

$$\boxed{-\frac{1}{2}\frac{\partial^2 \psi(x)}{\partial x^2} + V(x)\psi(x) = E\psi(x)\ .}$$

This is known as the 1D Schrödinger equation. (All constants set to 1)

Two familiar cases of 1D Schrödinger

a) Particle in a box of size L: $V = 0, 0 < x < L, V = \infty$ outside.

$$\phi_i = \sin\left(i\frac{\pi x}{L}\right)\sqrt{\frac{2}{L}}\ ,$$

where $i = 1, 2, 3, \ldots,$ and energy

$$E_i = \frac{\pi^2 i^2}{2L^2}\ .$$

b) Particle in a harmonic potential: $V = \frac{1}{2}\omega^2 x^2$.

$$\phi_i = (\omega/\pi)^{1/4}\frac{1}{\sqrt{2^i i!}}H_i(\sqrt{\omega}x)\exp(-\omega x^2/2)\ ,$$

where $i = 0, 1, 2, 3, \ldots,$ and energy is

$$E_i = \omega\left(i + \frac{1}{2}\right)\ .$$

Finite differences for 1D Schrödinger

Take the equation to be solved:

$$-\frac{1}{2}\frac{\partial^2\psi(x)}{\partial x^2} + V(x)\psi(x) = E\psi(x)$$

and discretize the space using an uniform grid

$$x \to \{x_i\}_{i=1}^{N} = -x_m, -x_m + h, -x_m + 2h, \ldots, x_m - h, x_m$$

so that the wave function becomes a vector of length N:

$$\psi(x) \to \{\psi(x_i)\}_{i=1}^{N} = \{\psi_i\}_{i=1}^{N}$$

Potential also at the same grid: $V(x_i)$

The product $V(x_i)\psi(x_i)$ gives a vector, so we can think V as a diagonal matrix

with elements: $V_{i,j} = \delta_{ij} V(x_i)$

Finite differences for 1D Schrödinger

Kinetic energy by the finite difference:

$$T\psi(x) = -\frac{1}{2}\frac{d^2\psi(x)}{dx^2} \approx \left\{-\frac{1}{2}\left[\psi(x+h)+\psi(x-h)\right]+\psi(x)\right\}/h^2$$

or

$$T\psi_i = -\frac{1}{2h^2}\psi_{i-1} - \frac{1}{2h^2}\psi_{i+1} + \frac{1}{h^2}\psi_i$$

Kinetic energy operator couples the wave function values at neighbouring sites.

$$T_{i,i} = \tfrac{1}{h^2}, T_{i,i+1} = -\tfrac{1}{2h^2}, T_{i,i-1} = -\tfrac{1}{2h^2},$$

and the of the elements rest are zero. This is a tridiagonal matrix.

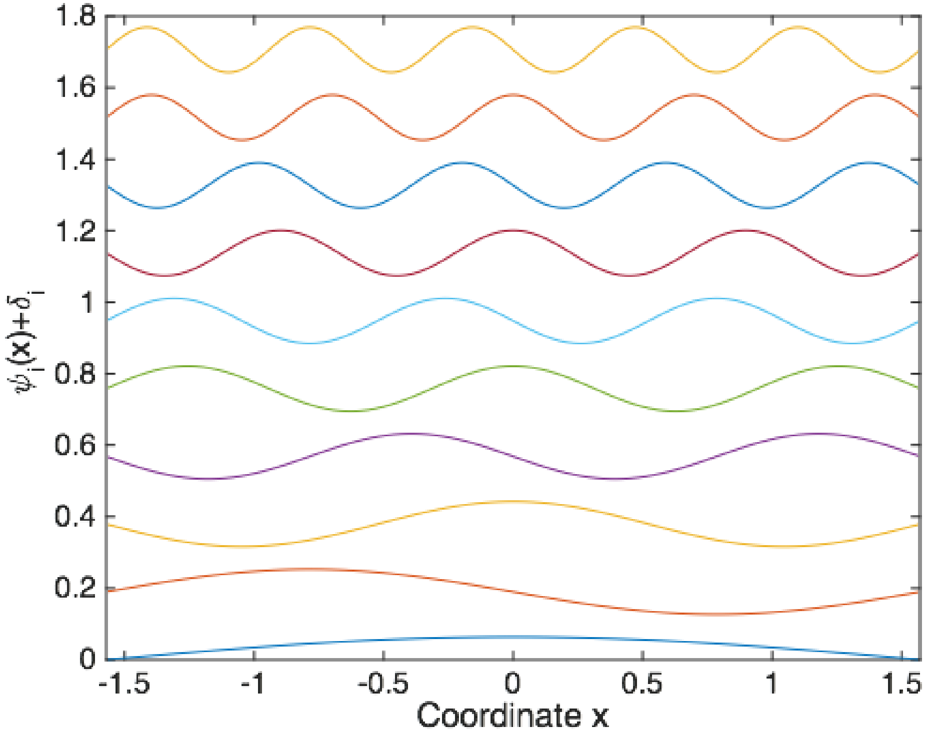
So the Schrödinger equation maps to a tridiagonal matrix eigenvalue equation:

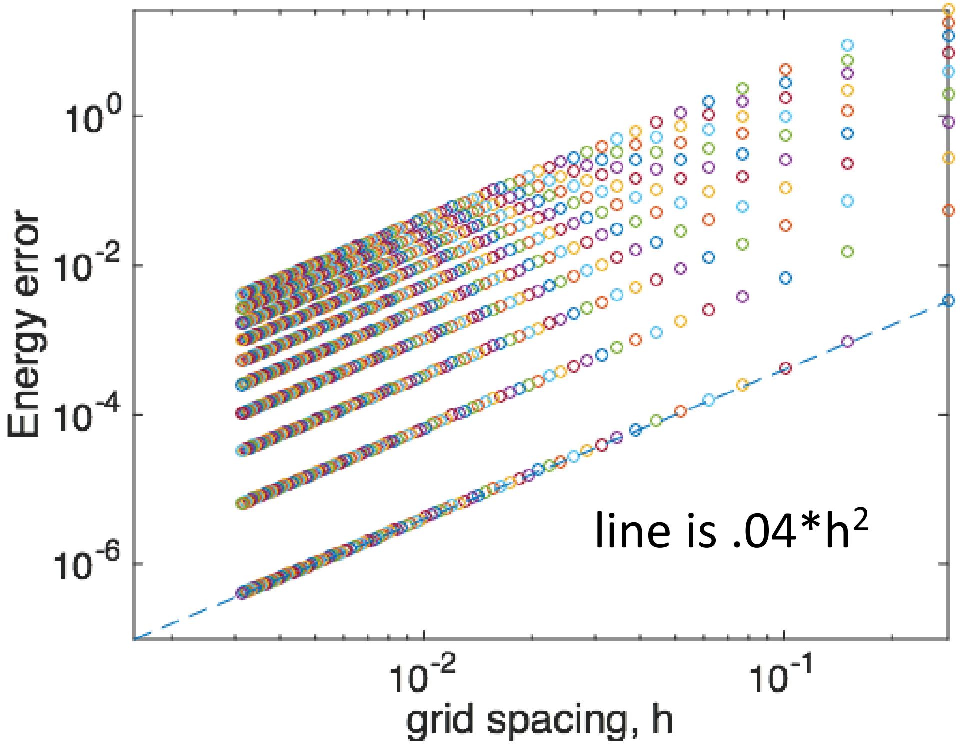
$$(T+V)\psi = E\psi$$

Test for the particle in a box

```
L=pi;
omega=0;
Nx=500;
h=L/(Nx+1);
x=-.5*L+h*(1:Nx);
V=.5*(omega*x).^2;

T=(diag(ones(Nx,1))-.5*diag(ones(Nx-1,1),1)-.5*diag(ones(Nx-1,1),-1))./h^2;
[wf, enes]=eig(T+diag(V));
```





Homework 7: FD stencils in 1D and 2D

a) Dimension 1. Implement a solver for the one-dimensional problem

$$\begin{cases} -u''(x) = (x-0.5)^3 - 2(x-0.5), & 0 < x < 1 \\ u(0) = u(1) = 0 \end{cases}$$

using the three-point stencil

$$-u''(x_i) \approx \frac{1}{h^2}(-u_{i-1} + 2u_i - u_{i+1})$$

Check that when varying the grid spacing h you converge to the exact solution at the rate h^2.

c) Dimension 2. Implement a solver for the two-dimensional problem

$$\begin{cases} -\Delta u(x,y) = \exp(-\frac{(x-0.5)^2+(y-0.5)^2}{18}), & (x,y) \in [0,1] \times [0,1] \\ u(0,y) = u(1,y) = u(x,0) = u(x,1) = 0 \end{cases}$$

using the five-point stencil (see, e.g.,
https://en.wikipedia.org/wiki/Discrete_Poisson_equation)

$$-\Delta u(x_i, y_j) \approx \frac{1}{h^2}(4u_{i,j} - u_{i-1,j} - u_{i+1,j} - u_{i,j-1} - u_{i,j+1})$$

What can you say about the convergence with respect to the grid speing h? (3 p)

Finite Differences: Pros & Cons

Pros:

Cons:

Discuss: What are the Pros and Cons of Finite Differences