Numerical Simulation Methods in Geophysics, Part 6: Finite Elements

1. MGPY+MGIN

thomas.guenther@geophysik.tu-freiberg.de

TUBAF

Die Ressourcenuniversität. Seit 1765.

Finite Differences

Recap Finite Differences

- elliptic (Poisson) or parabolic PDE problems
- replace partial differential operators ∂ by finite differences Δ
- transfer PDE into a matrix-vector equation $\mathbf{Au} = \mathbf{b}$
- finite-difference stencil spatial or temporal
- spatial derivative $\Rightarrow$ stiffness $\mathbf{A}$, temporal $\Rightarrow$ mass matrix $\mathbf{M}$
- time-stepping explicit, implicit or mixed (stable & accurate)
- accuracy depends on discretization, parameter contrast

Helmholtz equations

e.g. from Fourier assumption $u = u_0 e^{\imath\omega t}$

$$\boldsymbol{\nabla} \cdot (a \boldsymbol{\nabla} u) + k^2 u = f$$

- Poisson operator assembled in stiffness matrix $\mathbf{A}$
- additional terms with $u_i \Rightarrow$ mass matrix $\mathbf{M}$

$$\Rightarrow \mathbf{A} + \mathbf{M} = \mathbf{b}$$

Hyberbolic equations

Acoustic wave equation in 1D

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

u..pressure/velocity/displacement, c..velocity

Damped (mixed parabolic-hyperbolic) wave equation

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

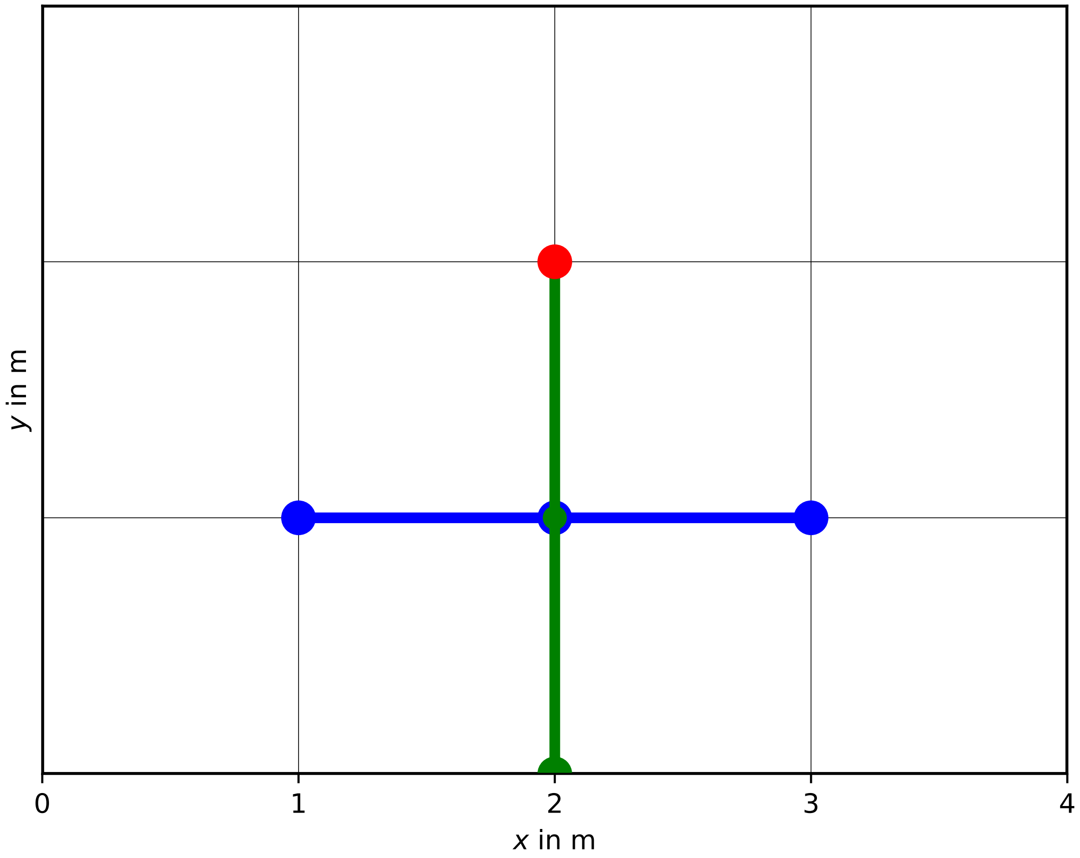
Discretization

$$\frac{\partial^2 u}{\partial t^2}^n \approx \frac{u^{n+1} - u^n}{\Delta t} - \frac{u^n - u^{n-1}}{\Delta t}$$

$$= \frac{u^{n+1} + u^{n-1} - 2u^n}{\Delta t^2} = c^2 \frac{\partial^2 u}{\partial x^2}^n$$

$$u^{n+1} = c^2 \Delta t^2 \frac{\partial^2 u}{\partial x^2}^n + 2u^n - u^{n-1}$$

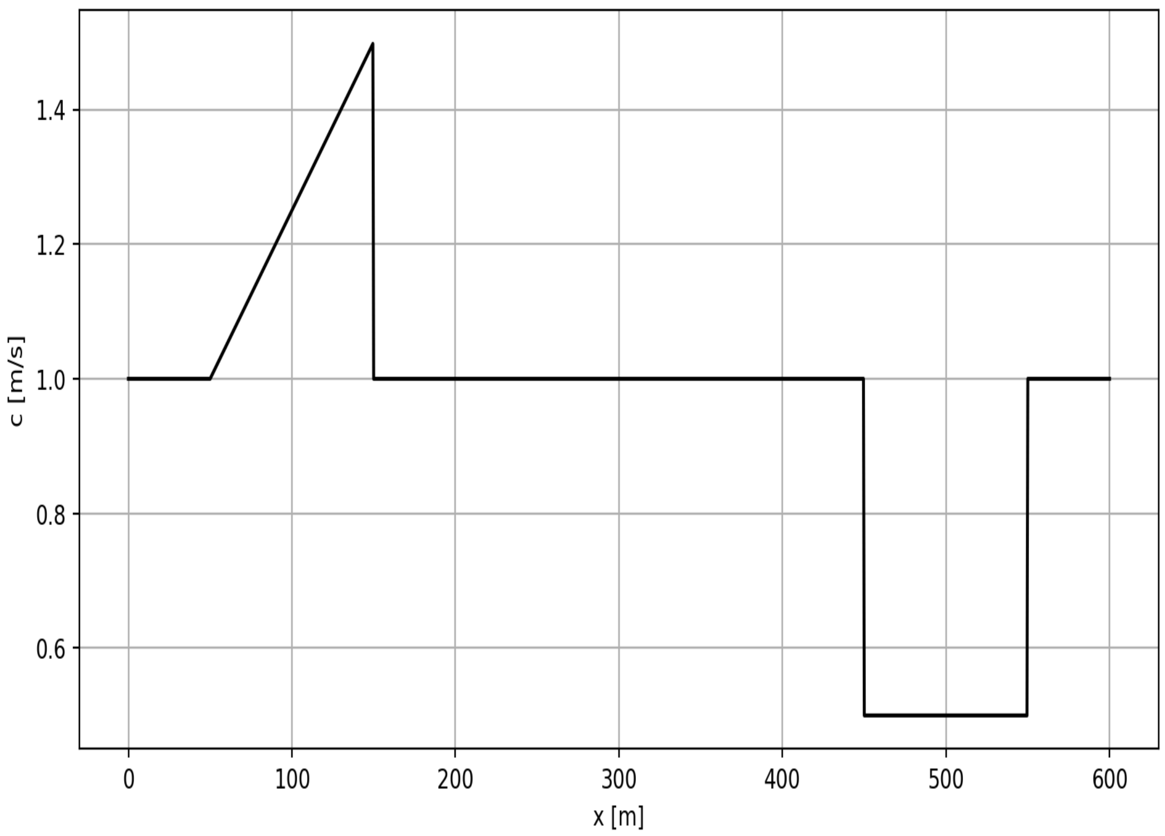
$$\mathbf{M}\mathbf{u}^{n+1} = (\mathbf{A} + 2\mathbf{M})\mathbf{u}^n - \mathbf{M}\mathbf{u}^{n-1}$$



Second derivative

Example: velocity distribution

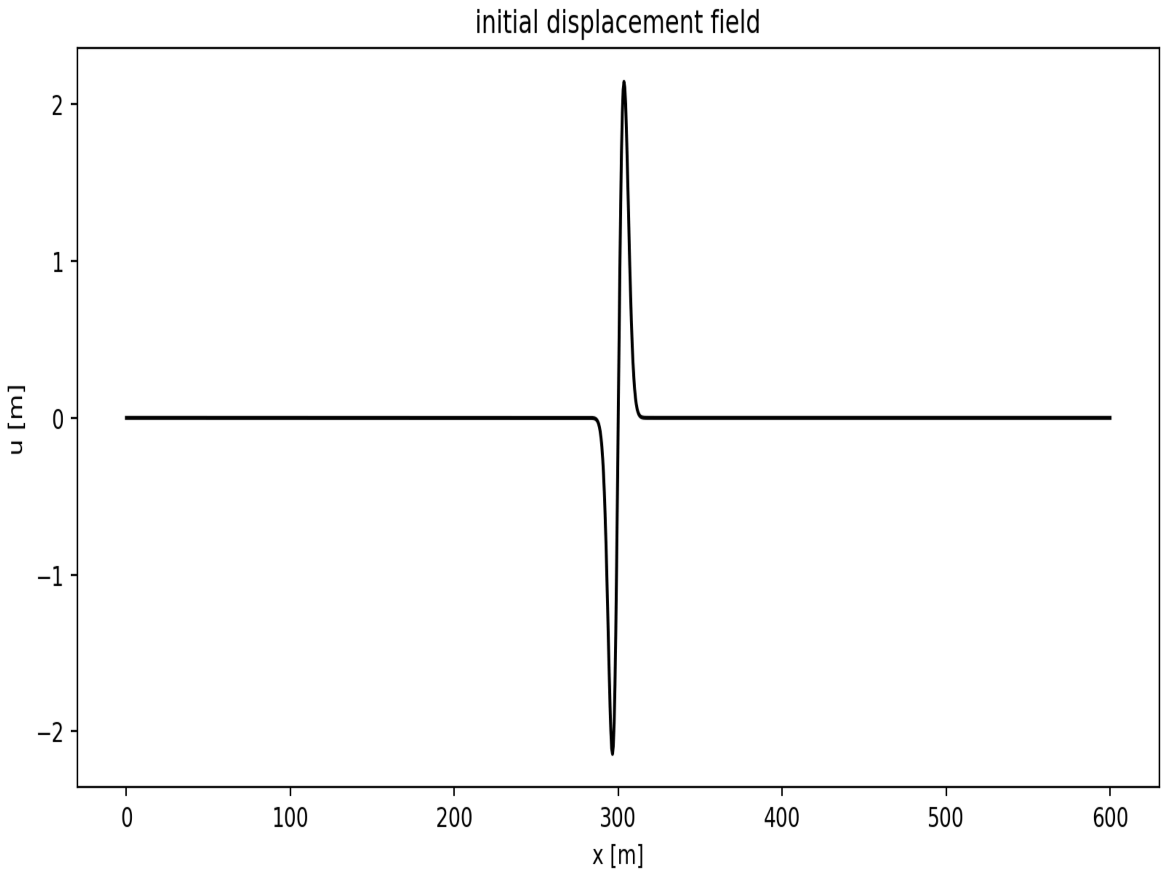
```python
import numpy as np
import matplotlib.pyplot as plt
x=np.arange(0, 600.01, 0.5)
c = 1.0*np.ones_like(x) # velocity in m/s
c[100:300] = 1 + np.arange(0,0.5,0.0025)
c[900:1100] = 0.5  # low velocity zone

# Plot velocity distribution.
plt.plot(x,c,'k')
plt.xlabel('x [m]')
plt.ylabel('c [m/s]')
plt.grid()
```



Initial displacement

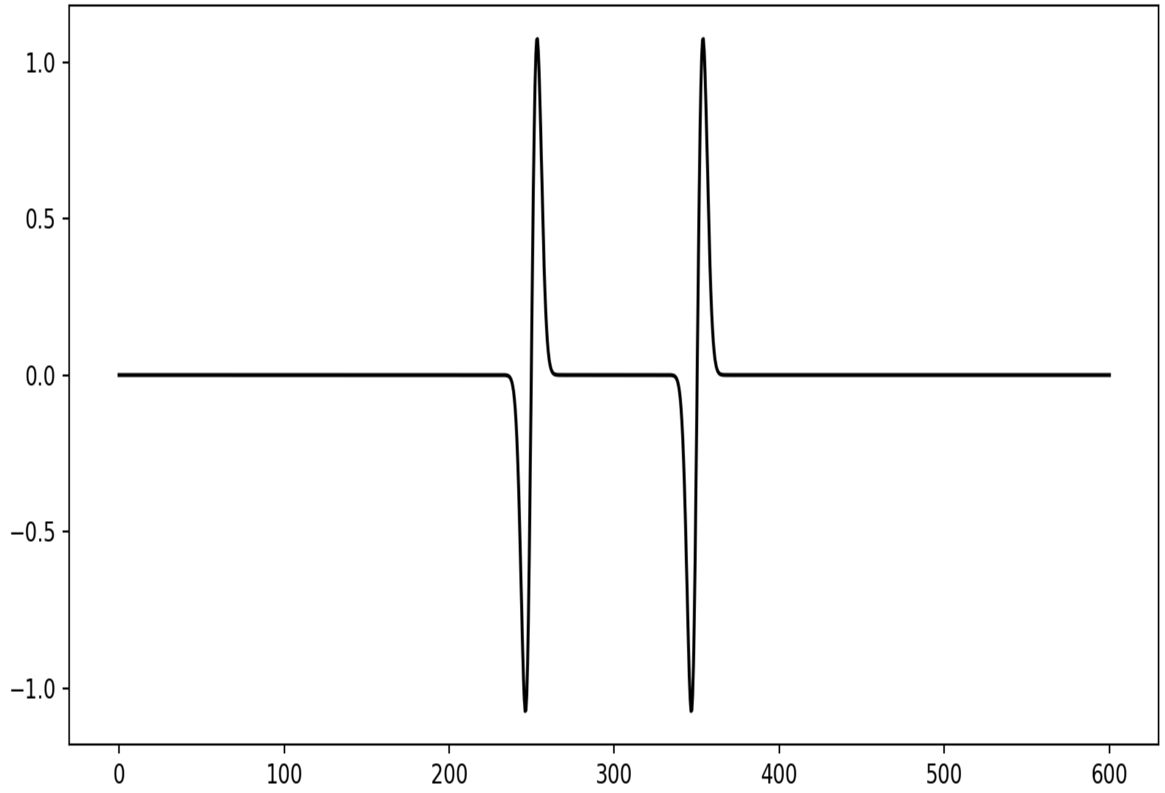
Derivative of Gaussian (Ricker wavelet)

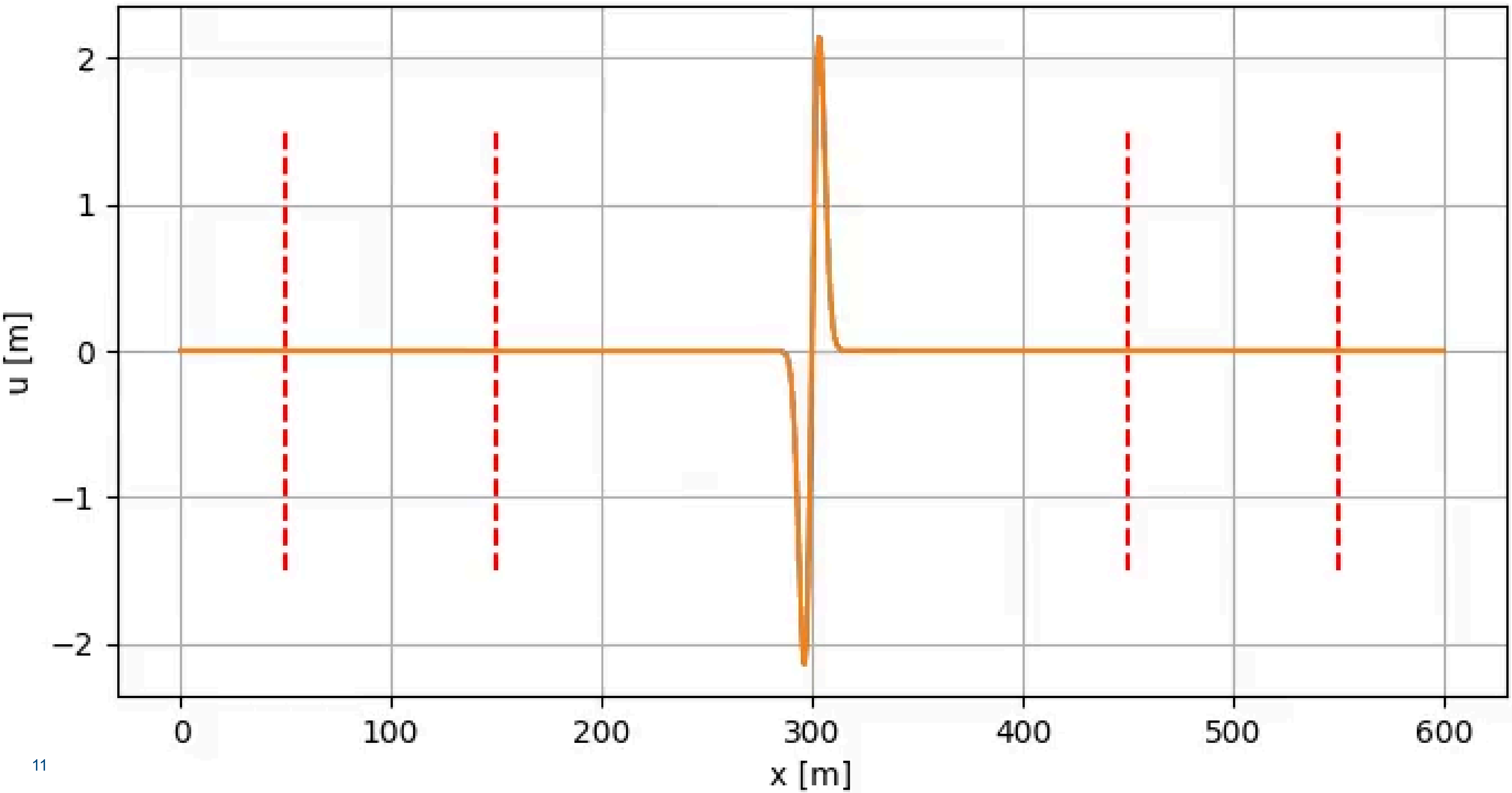
```python
l=5.0

# Initial displacement field [m].
u=(x-300.0)*np.exp(-(x-300.0)**2/l**2)
# Plot initial displacement field.
plt.plot(x,u,'k')
plt.xlabel('x [m]')
plt.ylabel('u [m]')
plt.title('initial displacement field')
plt.show()
```



Time propagation

```
u_last=u
dt = 0.5
ddu = np.zeros_like(u)
dx = np.diff(x)
for i in range(100):
    dudx = np.diff(u)/dx
    ddu[1:-1] = np.diff(dudx)/dx[:-1]
    u_next = 2*u-u_last+ddu*c**2 * dt**2
    u_last = u
    u = u_next

plt.plot(x,u,'k')
```




2
1
0
−1
−2
u [m]
0
100
200
300
400
500
600
x [m]

The Finite Element Method

History and background

- [1943] Courant: Variational Method
- [1956] Turner, Clough, Martin, Topp: Stiffness
- [1960] Clough: Finite Elements for static elasticity
- [1970-80] extension to structural, thermic and fluid dynamics
- [1990] computational improvements
- now main method for almost all PDE types

Geophysics: Poisson equation in 1970s, revival in 1990s and predominant in 2000s up to now

Variational formulation of Poisson equation

$$-\boldsymbol{\nabla} \cdot a \boldsymbol{\nabla} u = f$$

Multiplication with test function w and integration $\Rightarrow$ weak form

$$-\int_\Omega w \boldsymbol{\nabla} \cdot a \boldsymbol{\nabla} u \mathrm{d}\Omega = \int_\Omega w f \mathrm{d}\Omega$$

$$\boldsymbol{\nabla} \cdot (b\mathbf{c}) = b \boldsymbol{\nabla} \cdot \mathbf{c} + \boldsymbol{\nabla} b \cdot \mathbf{c}$$

$$\int_\Omega a \boldsymbol{\nabla} w \cdot \boldsymbol{\nabla} u \mathrm{d}\Omega - \int_\Omega \boldsymbol{\nabla} \cdot (w a \boldsymbol{\nabla} u) \mathrm{d}\Omega = \int_\Omega w f \mathrm{d}\Omega$$

Variational formulation of Poisson equation

$$\int_\Omega a\boldsymbol{\nabla} w \cdot \boldsymbol{\nabla} u \mathrm{d}\Omega - \int_\Omega \boldsymbol{\nabla} \cdot (wa\boldsymbol{\nabla} u)\mathrm{d}\Omega = \int_\Omega wf\mathrm{d}\Omega$$

use Gauss' law $\int_\Omega \boldsymbol{\nabla} \cdot \mathbf{A} = \int_\Gamma \mathbf{A} \cdot \mathbf{n}$

$$\int_\Omega a\boldsymbol{\nabla} w \cdot \boldsymbol{\nabla} u \mathrm{d}\Omega - \int_\Gamma aw\boldsymbol{\nabla} u \cdot \mathbf{n}\mathrm{d}\Gamma = \int_\Omega fw\mathrm{d}\Omega$$

Let u be constructed by shape functions v: $u = \sum_i u_i v_i$

$$\int_\Omega a\boldsymbol{\nabla} w \cdot \boldsymbol{\nabla} v_i \mathrm{d}\Omega - \int_\Gamma aw\boldsymbol{\nabla} v_i \cdot \mathbf{n}\mathrm{d}\Gamma = \int_\Omega fw\mathrm{d}\Omega$$

Galerkins method

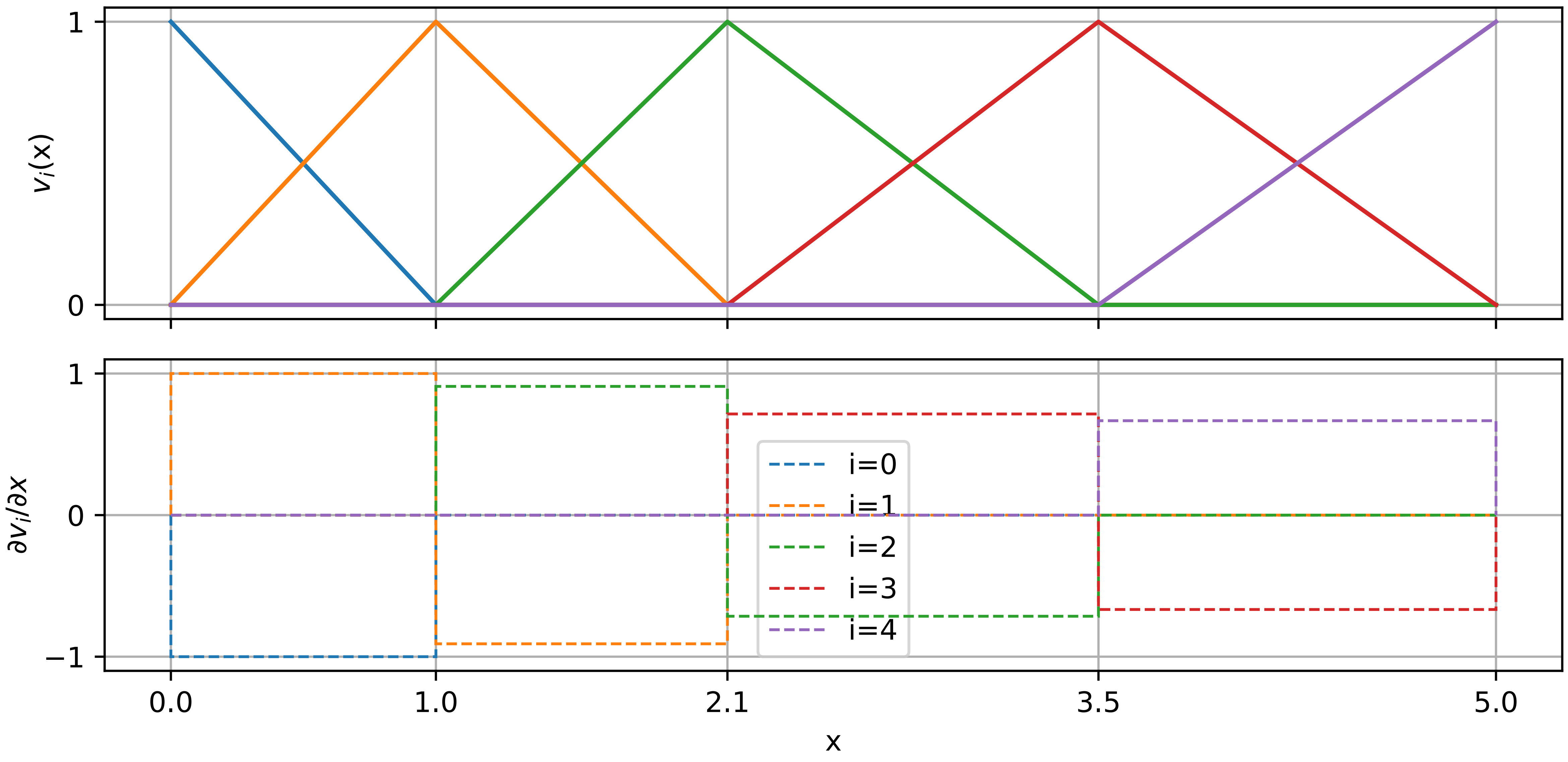
$$\int_\Omega a\boldsymbol{\nabla} w \boldsymbol{\nabla} v_i \mathrm{d}\Omega - \int_\Gamma a w \boldsymbol{\nabla} v_i \mathrm{d}\Gamma = \int_\Omega f w \mathrm{d}\Omega$$

Test functions equal shape (trial) functions $w_i = v_i$

$$\int_\Omega a\boldsymbol{\nabla} v_i \boldsymbol{\nabla} v_j \mathrm{d}\Omega - \int_\Gamma a w \boldsymbol{\nabla} v_j \mathrm{d}\Gamma = \int_\Omega f v_j \mathrm{d}\Omega$$

- choose v_i so that $\boldsymbol{\nabla} v_i$ is simple and $\boldsymbol{\nabla} v_i \cdot \boldsymbol{\nabla} v_j$ mostly 0
- divide subsurface in sub-volumes Ω_i with constant a_i and $\boldsymbol{\nabla} v_j$

Hat functions



Gradients for hat functions

Every element is surrounded by two nodes “carrying” a hat.

The gradients are piece-wise constant $\pm 1/\Delta x_i$

$$\Rightarrow \int_\Omega a\boldsymbol{\nabla} v_i \cdot \boldsymbol{\nabla} v_{i+1} \mathrm{d}\Omega = -\frac{a_i}{\Delta x_i^2} \cdot \Delta x_i = -\frac{a_i}{\Delta x_i}$$

$$-\int_\Omega a\boldsymbol{\nabla} v_i \cdot \boldsymbol{\nabla} v_i \mathrm{d}\Omega = \frac{a_{i-1}}{\Delta x_{i-1}^2}\Delta x_{i-1} + \frac{a_i}{\Delta x_i^2}\Delta x_i = \frac{a_{i-1}}{\Delta x_{i-1}} + \frac{a_i}{\Delta x_i}$$

System (stiffness) matrix

Matrix integrating gradient of base functions for neighbors A

$$\mathbf{A}_{i,i+1} = -\frac{a_i}{\Delta x_i^2} \cdot \Delta x_i$$

$$A_{i,i} = \int_{\Omega} a \boldsymbol{\nabla} v_i \cdot \boldsymbol{\nabla} v_i \mathrm{d}\Omega = -A_{i,i+1} - A_{i+1,i}$$

$\Rightarrow$ matrix-vector equation $\mathbf{A}\mathbf{u} = \mathbf{b} \Rightarrow$

Boundary conditions

second term

$$-\int_\Gamma a v_i \boldsymbol{\nabla} v_j \mathrm{d}\Gamma$$

Right-hand side vector

The right-hand-side vector $b=\int v_i f \mathrm{d}\Omega$ also scales with Δx

e.g. $f=\mathbf{\nabla}\cdot\mathbf{j}_s \Rightarrow b=\int v_i \mathbf{\nabla}\cdot\mathbf{j}_s\mathrm{d}\Omega=\int_\Gamma v_i\mathbf{j}_S\cdot\mathbf{n}$

system identical to FD for $\Delta x=$const

Difference of FE to FD

Any source function $f(x)$ can be integrated on the whole space!

Solution

$\mathbf{u}$ holds the coefficient u_i creating $u(x) = \sum u_i v_i(x)$

Difference of FE to FD

u is described on the whole space and approximates the solution, not the PDE!

Hat functions: u_i potentials on nodes, u piece-wise linear

Generality of FE

Arbitrary base functions v_i can be used to describe u