Scientific Computing I

Module 8: Discretisation of PDEs

Michael Bader

Lehrstuhl Informatik V

Winter 2005/2006

The Model Problem

- 2D Poisson Equation on unit square:

$$\frac{\partial^2}{\partial x^2}u(x,y) + \frac{\partial^2}{\partial y^2}u(x,y) = f(x,y) \qquad \text{in } \Omega = (0,1)^2$$

- Dirichlet boundary conditions:

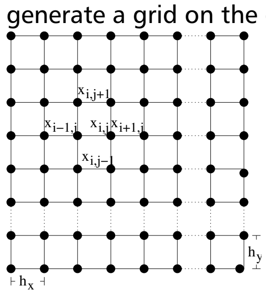
$$u(x,y) = g(x,y) \qquad \text{on } \partial\Omega$$

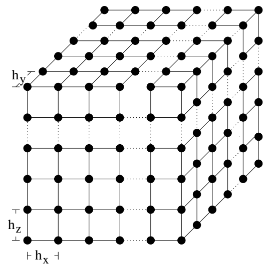
Part I

Finite Differences

Grid Generation

- generate a grid on the given domain





- Compute values of unknown function u at each grid point:

$$u_{ij} \approx u(x_{ij}) \qquad u_{ijk} \approx u(x_{ijk})$$

Finite Difference Discretisation

- Replace derivatives (at each grid point) by difference quotients:

$$\frac{\partial^2 u}{\partial x^2}(x_{i,j}) \approx \frac{u(x_{i+1,j}) - 2u(x_{i,j}) + u(x_{i-1,j})}{h_x^2}$$

$$\frac{\partial^2 u}{\partial x^2}(x_{i,j}) \approx \frac{u(x_{i+1,j}) - 2u(x_{i,j}) + u(x_{i-1,j})}{h_x^2}$$

- leads to linear system of equations:

$$\begin{aligned} \tfrac{1}{h^2}(u_{i+1,j} + u_{i,j+1} - 4u_{i,j} \\ + u_{i,j-1} + u_{i-1,j}) &= f(x_{i,j}) \quad x_{i,j} \in (0,1)^2 \\ u(x_{i,j}) &= g(x_{i,j}) \quad x_{i,j} \in \partial\Omega \end{aligned}$$

System of Linear Equations

- write linear system as $A_h x_h = f_h$
- A_h is a sparse matrix (band structure):

$$A_h = \frac{1}{h^2}\begin{pmatrix} B_h & -I & & \\ -I & \ddots & \ddots & \\ & \ddots & \ddots & -I \\ & & -I & B_h \end{pmatrix}$$

- $B_h = \text{tridiag}(-1, 4, -1)$
- A_h block-tridiagonal (5-diagonal) matrix
- boundary values to right-hand side

Stencil Notation

- illustrate matrix structure as a *discretisation stencil*
- represents one line of the matrix
- elements placed according to the geometrical position
- stencils for the Poisson equation:

$$1D: \quad \begin{bmatrix} 1 & -2 & 1 \end{bmatrix} \qquad 2D: \quad \begin{bmatrix} & 1 & \\ 1 & -4 & 1 \\ & 1 & \end{bmatrix}$$

Accuracy

- for 5-point Poisson stencil;
  order of accuracy: $\|u_h - u\| = \mathcal{O}(h^2) = \mathcal{O}(N^{-2})$
- *curse of dimension:*
  for that, we need $\mathcal{O}(N^d)$ points

Possibilities of an improvement:

- use higher-order discretisation
  - via higher order terms of Taylor series
  - leads to larger stencils
    (involving more neighbouring grid points)
- use locally refined (*adaptive*) grids

Part II

Finite Element Methods

Finite Elements – Main Ingredients

1. compute a *function* as numerical solution;
   search in a function space V_h:

$$u_h = \sum_j u_j \varphi_j(x), \qquad \text{span}\{\varphi_1, \ldots \varphi_J\} = V_h$$

2. solve *weak form* of PDE to reduce regularity properties

$$u'' = f \quad \longrightarrow \quad \int v'u' \, dx = \int vf \, dx$$

3. choose basis function with a *local support*, e.g.:

$$\varphi_j(x_i) = \delta_{ij}$$

Weak Forms and Weak Solutions

- consider a PDE $Lu = f$ (e.g. $Lu = \Delta u$)
- transformation to the *weak form*:

$$\langle Lu, v \rangle = \int vLu \, dx = \int vf \, dx = \langle f, v \rangle \quad \forall v \in V$$

  V a certain class of functions
- “real solution” u also solves the weak form
- $\langle Lu, v \rangle$ a *bilinear form*; often written as:

$$a(Lu, v) = \langle f, v \rangle \quad \forall v \in V$$

Weak Form of the Poisson Equation

- Poisson equation with Dirichlet conditions:

$$-\Delta u = f \quad \text{in } \Omega, u = 0 \quad \text{on } \delta\Omega$$

- weak form:

$$-\int_\Omega v \Delta u \, d\Omega = \int_\Omega vf \, d\Omega \quad \forall v$$

  apply Green’s formula (and boundary conditions):

$$\int_\Omega \nabla v \cdot \nabla u \, d\Omega = \int_\Omega vf \, d\Omega \quad \forall v$$

- weaker requirements for a solution u:
  twice differentiabale $\rightarrow$ *first derivative integrable*

Choose Test and Ansatz Space

- search for weak solutions u in a certain function space W

$$\int vL\left(\sum_j u_j\varphi_j(x)\right)\mathrm{d}x = \int vf\,\mathrm{d}x \quad \forall v \in V$$

where $\mathrm{span}\{\varphi_j\} = W$ ("ansatz space")
- choose a basis $\{\psi_i\}$ of the *test* space V; then:

$$\int \psi_i L\left(\sum_j u_j\varphi_j(x)\right)\mathrm{d}x = \int \psi_i f\,\mathrm{d}x \quad \forall \psi_i$$

- leads to system of equations for unknowns u_j
- usually V and W chosen identically (Ritz-Galerkin method)

Discretisation – Finite Elements

- L linear $\Rightarrow$ system of linear equations

$$\sum_j \underbrace{\left(\int \psi_i L\varphi_j(x)\,\mathrm{d}x\right)}_{=:A_{ij}} u_j = \int \psi_i f\,\mathrm{d}x \quad \forall \psi_i$$

- aim: make matrix A *sparse* $\rightarrow$ most $A_{ij} = 0$
- approach: local basis functions on a discretisation grid
- ψ_j, φ_j zero everywhere except in grid cells adjacent to grid point x_j
- $A_{ij} = 0$, if ψ_i and φ_j don't overlap

A Road to Theory

- weak formulation is equivalent to variational approach:
solution u minimises an energy functional
- *best approximation* property:

$$\left\|u - u_h^{FE}\right\|_a \leq \inf_{u_h \in V} \|u - u_h\|_a$$

in terms of the norm induced by the bilinear form a (energy norm)
- thus: error bounded by interpoplation error (in energy norm)

Example Problem: Poisson 1D

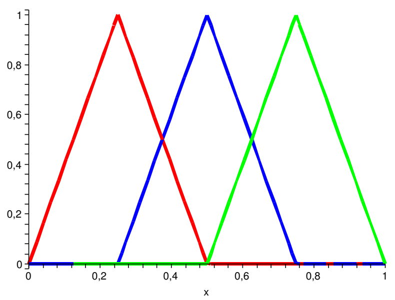
- 1D Poisson's equation on $\Omega = [0,1]$, homogeneous Dirichlet boundary conditions
- weak form:

$$\int_0^1 \nabla v \cdot \nabla u\,\mathrm{d}x = \int_0^1 vf\,\mathrm{d}x \quad \forall v$$

- compuational grid:
$x_i = ih$, (for $i = 1,\ldots,n-1$); mesh size $h = 1/n$
- $V = W$: piecewise linear functions (on intervals $[x_i, x_{i+1}]$)

Nodal Basis

$$\varphi_i(x) := \begin{cases} \frac{1}{h}(x - x_{i-1}) & x_{i-1} < x < x_i \\ \frac{1}{h}(x_{i+1} - x) & x_i < x < x_{i+1} \\ 0 & \text{otherwise} \end{cases}$$



Nodal Basis – System of Equations

- stiffness matrix:

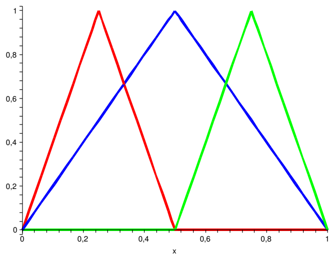
$$\frac{1}{h}\begin{pmatrix} 2 & -1 & & \\ -1 & 2 & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{pmatrix}$$

- right hand sides (assume $f(x) = \alpha \in \mathbb{R}$):

$$\int_0^1 \varphi_i(x)f(x)\,\mathrm{d}x = \int_0^1 \varphi_i(x)\alpha\,\mathrm{d}x = \alpha h$$

- system of equations very similar to finite differences

Hierarchical Basis



- leads to diagonal stiffness matrix! (for 1D Poisson)
- solution function identical to that with nodal basis (same function space)

Element Stiffness Matrices

- domain Ω splitted into finite elements $\Omega^{(i)}$
- element-wise evaluation of the integrals:

$$\begin{aligned}\int_\Omega \nabla v \cdot \nabla u \,\mathrm{d}x &= \sum_i \int_{\Omega^{(i)}} \nabla v \cdot \nabla u \,\mathrm{d}x \\ \int_\Omega v f \,\mathrm{d}x &= \sum_i \int_{\Omega^{(i)}} v f \,\mathrm{d}x\end{aligned}$$

- leads to system of equations for each element:

$$A^{(i)} x = b^{(i)}$$

- accumulate to obtain global system:

$$\underbrace{\sum_i A^{(i)}}_{=:A} x = \sum_i b^{(i)}$$

Example: 1D Poisson

Notation:

- notation: omit zero columns/rows (leaves only unknowns that are in $\Omega^{(i)}$)

1D Poisson:

- $\Omega = [0,1]$ splitted into $\Omega^{(i)} = [x_{i-1}, x_i]$
- nodal basis; leads to element stiffness matrix:

$$A^{(i)} = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$$

- stencil notation:

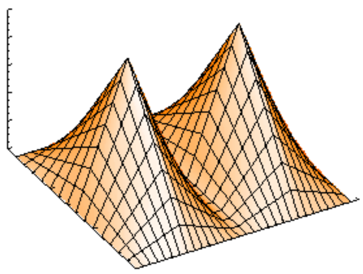
$$[1^* \;\; -1] + [-1 \quad 1^*] \rightarrow [-1 \quad 2 \; -1]$$

Example: 2D Poisson

- $-\Delta u = f$ on domain $\Omega = [0,1]^2$
- splitted into $\Omega^{(i,j)} = [x_{i-1}, x_i] \times [x_{j-1}, x_j]$
- bilinear basis functions

$$\varphi_{ij}(x,y) = \varphi_i(x)\varphi_j(y)$$

- "pagoda" functions



Example: 2D Poisson (2)

- leads to element stiffness matrix:

$$A^{(i)} = \begin{pmatrix} 2 & -\frac{1}{2} & -\frac{1}{2} & -1 \\ -\frac{1}{2} & 2 & -1 & -\frac{1}{2} \\ -\frac{1}{2} & -1 & 2 & -\frac{1}{2} \\ -1 & -\frac{1}{2} & -\frac{1}{2} & 2 \end{pmatrix}$$

- accumulation leads to 9-point stencil

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Typical workflow

1. choose elements:
   - quadratic or cubic cells
   - triangles (structured, unstructured)
   - tetrahedra, etc.
2. set up basis functions for each element Ω_h; at all nodes $x_i \in \Omega_h$

$$\begin{aligned}\varphi_i(x_i) &= 1 \\ \varphi_i(x_j) &= 0 \quad \text{for all } j \neq i\end{aligned}$$

3. for element stiffness matrix, compute all

$$A_{ij} = \int_{\Omega_h} \varphi_i L \varphi_j \,\mathrm{d}\Omega$$

4. accumulate global stiffness matrix