MAST90026 Computational Differential Equations: Week 1

Jesse Collis
Modified from Hailong Guo (2022)

Semester 1 2024

The University of Melbourne

Goal of the subject

Cover numerical techniques for solving

- Boundary value problems (BVPs) for ODEs
- Simple PDEs
  - 2D Elliptic PDES
  - 1+1 (1 space dimension, 1 time dimension) parabolic, hyperbolic
- Sparse solvers for linear systems

Model equation

Convection-diffusion equation:

$$u_t + \vec{v} \cdot \nabla u = \nabla \cdot (\mathrm{D}\, \nabla u) + f.$$

$\nabla = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y})$

$\vec{v}, D > 0, f$ given functions of (x, y)

Scalar, linear parabolic equation

Model equation

Limiting cases $\vec{v} = \vec{0}$: Heat equation

$$u_t = \nabla \cdot (\mathrm{D}\, \nabla u) + f.$$

1D heat equation:

$$u_t = D u_{xx} + f.$$

Model equation

Limiting cases $D = 0$: Advection/transport/1-way wave equation

$$u_t + \vec{v} \cdot \nabla u = f.$$

1D hyperbolic equation:

$$u_t + vu_x + f.$$

Model equation

Limiting cases $u_t = 0$ (steady state): elliptic equation

$$\vec{v} \cdot \nabla u = \nabla \cdot (\mathrm{D} \, \nabla u) + f.$$

Further $\vec{v} = \vec{0} \rightarrow$ Poisson equation

$$-\nabla \cdot (\mathrm{D} \, \nabla u) = f.$$

1D elliptic equation

$$v u_x = \mathrm{D} \, u_{xx} + f.$$

Issues

How do we iterate in time (for IVPs)?

How do we represent the solution in space (for BVPs/PDEs)?

How do we handle higher dimensionality?

How do we handle non-linearities?

2nd order BVPs

Linear BVP

$$-u'' + p(x)u' + q(x)u = r(x), x \in [a, b]$$

with linear separated boundary condition (Robin boundary condition)

$$\begin{aligned} A_{11}u(a) + A_{12}u'(a) &= \beta_1, \\ A_{21}u(b) + A_{22}u'(b) &= \beta_2. \end{aligned}$$

More general 2nd order BVPs

General form

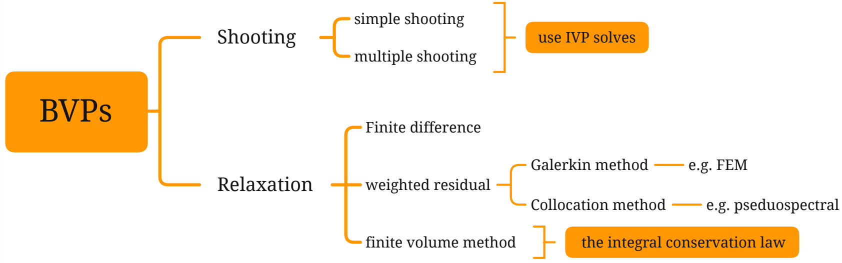
$$u'' = f(x, u, u'),$$

with separated boundary condition at $x = a$ and $x = b$

$$g_1(u(a), u'(a)) = B_1,$$
$$g_2(u(b), u'(b)) = \beta_2.$$

Note linear BVPs could have 0, 1, or ∞ many solutions.

Existing methods for boundary value problems



General principle

Consider

$$Lu = r(x)$$

where L is a linear operator involving derivatives.

The main difference between finite difference methods(FDMs) and weighted residual methods (WRMs):

1. For FDM, approximate the differential operator L.
2. For WRMs, approximate the solution u.

Finite difference formulae

Idea: Approximate u' and u''.

Only approximate derivatives at selected points: $h = \frac{b-a}{N+1}$ and $x_j = a + jh$

Can also use nonuniform meshes $\{x_j\}$ with $h_j = x_{j+1} - x_j$.

Some examples

For u':

- Forward difference: $u'(x) = \frac{u(x+h) - u(x)}{h} + \text{error}$.
- Backward difference: $u'(x) = \frac{u(x) - u(x-h)}{h} + \text{error}$.
- Central difference: $u'(x) = \frac{u(x+h) - u(x-h)}{2h} + \text{error}$.

For u'':

- Central difference: $u''(x) = \frac{u(x+h) - 2u(x) + u(x-h)}{h^2} + \text{error}$.

General finite difference scheme

In general, approximate $u^{(k)}$ as

$$u^{(k)}(x) = \sum_{i=1}^{n} c_i u(x + h_i) + \text{error}.$$

Derivation:

- The method of underdetermined coefficient (Taylor series expansion)
- Interpolation and then differentiation

Error for interpolating using n points is $\mathcal{O}(h^n)$. The worst-case error for approximating $u^{(k)}$ is $\mathcal{O}(h^{n-k})$.

Discretizaiton error

Central finite difference error:

$$\begin{aligned}
&\frac{u(x+h)-u(x-h)}{2h} \\
=&\frac{1}{2h}\left(u(x)+hu'(x)+\frac{1}{2}h^2u''(x)+O(h^3)\right)- \\
&\frac{1}{2h}\left(u(x)-hu'(x)+\frac{1}{2}h^2u''(x)+O(h^3)\right) \\
=&\frac{2hu'(x)+O(h^3)}{2h} \\
=&u'(x)+O(h^2),
\end{aligned}$$

Similarly, the error for FD, BD, CD for u''.

FDM for BVPs

Consider

$$u''(x) + p(x)u'(x) + q(x)u(x) = r(x),$$
$$u(a) = \alpha, u(b) = \beta.$$

Consider $h = \frac{b-a}{N+1}$ and $x_j = a + jh$:

Idea: Use central difference approximation for u' and u'' at x_j:

$$\frac{u_{j+1} - 2u_j + u_{j-1}}{h^2} + p(x_j)\frac{u_{j+1} - u_{j-1}}{2h} + q(x_j)u_j = r(x_j).$$

Linear system for FDM

Let $p(x_j) = p_j$, $q(x_j) = q_j$ and $r(x_j) = r_j$.

$$\begin{aligned}
u_0 &= \alpha, \\
\frac{u_0 - 2u_1 + u_2}{h^2} + p_1 \frac{u_2 - u_0}{2h} + q_1 u_1 &= r_1, \\
\frac{u_1 - 2u_2 + u_3}{h^2} + p_2 \frac{u_3 - u_1}{2h} + q_2 u_2 &= r_2, \\
&\vdots \\
\frac{u_{N-1} - 2u_N + u_{N+1}}{h^2} + p_N \frac{u_{N+1} - u_{N-1}}{2h} + q_N u_N &= r_N, \\
u_{N+1} &= \beta.
\end{aligned}$$

Simplification of Linear system for FDM

Change the second and second last equation to:

$$
\begin{aligned}
\frac{-2u_1 + u_2}{h^2} + p_1 \frac{u_2}{2h} + q_1 u_1 &= r_1 - \frac{\alpha}{h^2} + p_1 \frac{\alpha}{2h}, \\
\frac{u_{N-1} - 2u_N}{h^2} + p_N \frac{-u_{N-1}}{2h} + q_N u_N &= r_N - \frac{\beta}{h^2} - p_N \frac{\beta}{2h}.
\end{aligned}
$$

Matrix form

$$
\frac{1}{h^2}\begin{pmatrix} -2+h^2 q_1 & 1+h\frac{p_1}{2} & 0 & 0 \\ 1-h\frac{p_2}{2} & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & 1+h\frac{p_{N-1}}{2} \\ 0 & 0 & 1-h\frac{p_N}{2} & -2+h^2 q_N \end{pmatrix}\begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{pmatrix}
$$

$$
= \begin{pmatrix} r_1 - \frac{\alpha}{h^2} + p_1\frac{\alpha}{2h} \\ r_2 \\ \vdots \\ r_N - \frac{\beta}{h^2} - p_N\frac{\beta}{2h} \end{pmatrix}
$$

Questions to ask

1. Does this have a solution?
2. How hard is it to solve?
3. How well does $\mathbf{u}$ approximate $u(x)$ for a given h?
4. Does $u_j \to u(x_j)$ as $h \to 0$?

Measure the error

The grid function (i.e. it is only defined on the grid/mesh) norms:

$$\|\Sigma\|_1 = h \sum |\Sigma_i|,$$
$$\|\Sigma\|_2 = \left(h \sum |\Sigma_i|^2\right)^{\frac{1}{2}},$$
$$\|\Sigma\|_\infty = \max |\Sigma_i|.$$

Define the global error:

$$\|E\| = \|\mathbf{u} - \hat{\mathbf{u}}\|.$$

Some terminologies

Definition : Convergence

A method with global error $\|E\| = O(h^p)$ as $h \to 0$ is convergent of order p.

Definition : Local Truncation Error (LTE)

The local truncation error τ_j is the residual at grid point x_j when the true solution is put into the finite difference formula.

Definition : Consistency

A method is consistent of order p if $\|\tau\| = O(h^p)$ as $h \to 0$.

LTE for our system

$$
\begin{aligned}
\tau_j =& \frac{1}{h^2}(u(x_{j+1}) - 2u(x_j) + u(x_{j-1})) + p(x_j)\frac{u(x_{j+1}) - u(x_{j-1})}{2h} + \\
& q(x_j)u(x_j) - r(x_j) \\
=& u''(x_j) + p(x_j)u'(x_j) + q(x_j)u(x_j) - r(x_j) + Ch^2 u^{(3)} + O(h^4) \\
=& Ch^2 u^{(3)}(\xi) + O(h^4),
\end{aligned}
$$

Error equation

Original scheme

$$A\mathbf{u} = \mathbf{b}.$$

Consistency

$$A\hat{\mathbf{u}} = \mathbf{b} + \tau.$$

The global error $\mathbf{E} = \mathbf{u} - \hat{\mathbf{u}}$ satisfies

$$A(\mathbf{u} - \hat{\mathbf{u}}) = -\tau$$

Convergence

Error equation

$$\mathbf{E} = -A^{-1}\tau.$$

Taking norms

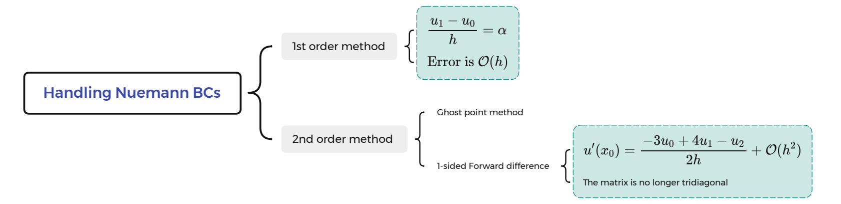
$$\begin{aligned}\|\mathbf{E}\| &= \| - A^{-1}\tau\| \\ &\leq \|A^{-1}\|\|\tau\| \\ &\leq \|A^{-1}\| Ch^2.\end{aligned}$$

Summary

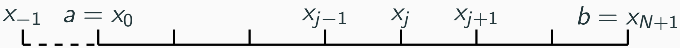
$$\text{Consistency} + \text{Stability} \Rightarrow \text{Convergence}$$

Other type boundary condition

Taking $u'(x_0) = \alpha$ as an exmape.



Ghost point method



Apply ODE at x_0 and use central difference on BC to eliminate u_{-1}:

$$\begin{aligned}\tfrac{u_{-1}-2u_0+u_1}{h^2} + p_0\left(\tfrac{u_1-u_{-1}}{2h}\right) + q_0 u_0 &= r_0\\ \tfrac{u_1-u_{-1}}{2h} &= \alpha\end{aligned}$$

$$\Rightarrow \frac{-2u_0+2u_1}{h^2} + q_0 u_0 = r_0 - p_0\alpha + \frac{2\alpha}{h}$$