Chapter 8: Discretization of Time-Dependent problems

Michael Hanke

Mathematical Models, Analysis and Simulation, Part I

A Parabolic Problem

Let $\Omega \subset \mathbb{R}^2$ be a bounded domain with boundary $\Gamma = \partial\Omega$.

- Differential equation

$$u_t - \mathrm{div}(k\nabla u) = f(x,y), \quad (x,y) \in \Omega.$$

- Boundary condition

$$u|_\Gamma = g(x,y), \quad (x,y) \in \Gamma.$$

- Initial condition

$$u(x,y,0) = u_0(x,y), \quad (x,y) \in \Omega.$$

Classification of Numerical Methods

When constructing numerical methods, we use the idea of separation of variables:

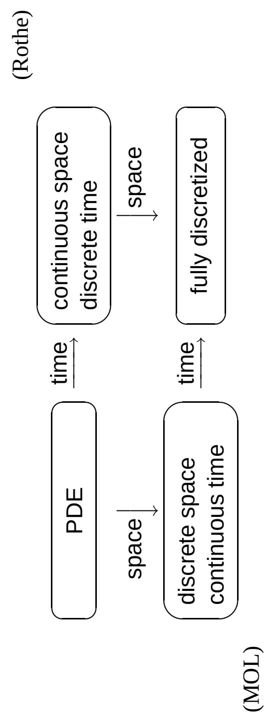
$$u(t,x,y) = T(t)X(x,y)$$

Semi-discretization in space For X, we got an elliptic boundary value problem:
$\Longrightarrow$ Use finite difference or finite element methods.
This is the method of lines (MOL).

Semi-discretization in time For T, we got an initial value problem:
$\Longrightarrow$ Use a finite difference method (e.g., Runge-Kutta or multistep methods).
This method is often called *Rothe's method*.

Complete discretization in space and time Use finite difference methods simultaneously.
This approach will be considered in subsequent lectures.

Classification: Cont



MOL With Finite Elements

- Trial function space: $V_g = \{v | v \in H^1(\Omega), v|_\Gamma = g\}$.
- Test function space: $V_0 = H_0^1(\Omega)$.
- Weak formulation: Find $u \in V_g \times [0,T]$ such that

$$\int_\Omega u_t v d\Omega + \int_\Omega k\nabla u \cdot \nabla v d\Omega = \int_\Omega f v d\Omega \text{ for all } v \in V_0.$$

- Replace V_g by a finite element space V_h (h – discretization parameter) with

$$V_h = \text{lin}\{\underbrace{\phi_1,\ldots,\phi_N}_{\text{nodes inside }\Omega}, \underbrace{\phi_{N+1},\ldots,\phi_{N+M}}_{\text{nodes on }\Gamma}\}.$$

- $u_h \in V_h \times [0,T]$ means

$$u_h(x,y,t) = \sum_{i=1}^{N} \tau_i(t)\phi_i(x,y) + \underbrace{\sum_{i=N+1}^{N+M} g(x_i,y_i)\phi_i(x,y)}_{\text{Dirichlet bc}}.$$

MOL: Cont

- Substituting u_h for u and ϕ_i, $i = 1,\ldots,N$ for v yields

$$\boxed{\mathbf{M}\frac{d}{dt}\tau + \mathbf{A}\tau = \mathbf{f},}$$

where

$$a_{ij} = \int_\Omega k\nabla\phi_i \cdot \nabla\phi_j d\Omega,$$

$$f_i = \int_\Omega f\phi_i d\Omega - \sum_{j=N+1}^{N+M} g(x_j,y_j)\int_\Omega \phi_i\phi_j d\Omega,$$

$$m_{ij} = \int_\Omega \phi_i\phi_j d\Omega.$$

$\mathbf{M}$ is called the *mass matrix*.

A 1D Example

- Consider: $u_t - u_{xx} = 0$, subject to $u(0) = u(1) = 0$.
- P1 ansatz functions:

$$\phi_i(x) = \begin{cases} \frac{x-x_{i-1}}{x_i-x_{i-1}}, & x_{i-1} \le x \le x_i, \\ \frac{x_i-x}{x_{i+1}-x_i}, & x_i \le x \le x_{i+1}, \\ 0, & \text{elsewhere} \end{cases}, \phi_{i,x}(x) = \begin{cases} \frac{1}{x_i-x_{i-1}}, & x_{i-1} \le x \le x_i, \\ \frac{-1}{x_{i+1}-x_i}, & x_i \le x \le x_{i+1}, \\ 0, & \text{elsewhere} \end{cases}.$$

- Equidistant grid: $x_i = ih$ where $h = (N+1)^{-1}$.

A 1D Example: Cont

- Stiffness matrix:

$$\mathbf{A} = \frac{1}{h}\begin{pmatrix} 2 & -1 & 0 & \ldots & \ldots & 0 \\ -1 & 2 & -1 & 0 & \ldots & 0 \\ 0 & -1 & 2 & -1 & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & 0 \\ 0 & \ldots & \ldots & -1 & 2 & -1 \\ 0 & \ldots & \ldots & 0 & -1 & 2 \end{pmatrix}.$$

- Mass matrix:

$$\mathbf{M} = \frac{h}{6}\begin{pmatrix} 4 & 1 & 0 & \ldots & \ldots & 0 \\ 1 & 4 & 1 & 0 & \ldots & 0 \\ 0 & 1 & 4 & 1 & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & 0 \\ 0 & \ldots & \ldots & 1 & 4 & 1 \\ 0 & \ldots & \ldots & 0 & 1 & 4 \end{pmatrix}.$$

A 1D Example: Discretization by Explicit Methods

As a typical example, use the explicit Euler method:

$$\boxed{\mathbf{M}\frac{\tau^{n+1}-\tau^{n}}{\Delta t}+\mathbf{A}\tau^{n}=\mathbf{f}}$$

Hence:

$$\mathbf{M}\tau^{n+1}=\mathbf{M}\tau^{n}+\Delta t(\mathbf{f}-\mathbf{A}\tau^{n}).$$

We must solve a linear system in every time step, even if the method is explicit!

Trick: Modify $\mathbf{M}$ such that the modified matrix $\tilde{\mathbf{M}}$ is diagonal:

$$\tilde{m}_{ij}=\begin{cases}\sum_{k=1}^{N} m_{ik}, & i=j\\ 0 & i\neq j\end{cases}$$

This is called *mass lumping*.

Mass lumping cannot be used for highly oscillatory problems.

A 1D Example: Discretization by Implicit Methods

Implicit Euler:

$$\boxed{\mathbf{M}\frac{\tau^{n+1}-\tau^{n}}{\Delta t}+\mathbf{A}\tau^{n+1}=\mathbf{f}}$$

Hence:

$$(\mathbf{M}+\Delta t\mathbf{A})\tau^{n+1}=\mathbf{M}\tau^{n}+\Delta t\mathbf{f}.$$

Here, a linear system must be solved in every step, even for lumped mass matrices.

Note: $\mathbf{Au}\approx-\mathrm{div}(k\nabla u)$, $\mathbf{Mu}\approx u$. Hence,

$$\left(\frac{1}{\Delta t}\mathbf{M}+\mathbf{A}\right)\tau^{n+1}\approx\frac{1}{\Delta t}u^{n+1}-\mathrm{div}(k\nabla u^{n+1}),$$

such that a steady-state reaction-diffusion problem must be solved in every step.

Note: Stability considerations will be postponed.

Example: Pollution Of Water In A River

- The river is stretched along the x-axis with velocity V in positive direction,

$$Vu_x=\varepsilon(u_{xx}+u_{yy}), V>0.$$

- At time $t=0$, a pollutant is released near the left river side,

$$u(0,y)=H(y)=\begin{cases}1, & y>0\\ 0, & y<0\end{cases}.$$

  H is the *Heaviside function*.
- At least near $x=0$, we can neglect the contribution of u_{xx} since the spreading is dominated by advection,

$$\boxed{u_x=(\varepsilon/V)u_{yy}}$$

  subject to pure intial conditions.

Q: What happens if diffusion is small compared to advection?

A Case Study

Read: Strang, p 538–542

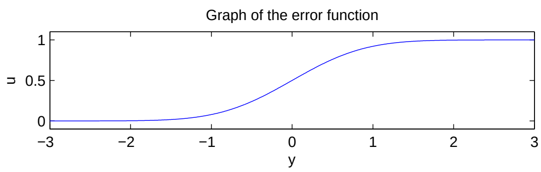
- Consider

$$\boxed{u_x=\beta u_{yy}.}\quad 0<\beta\ll 1$$

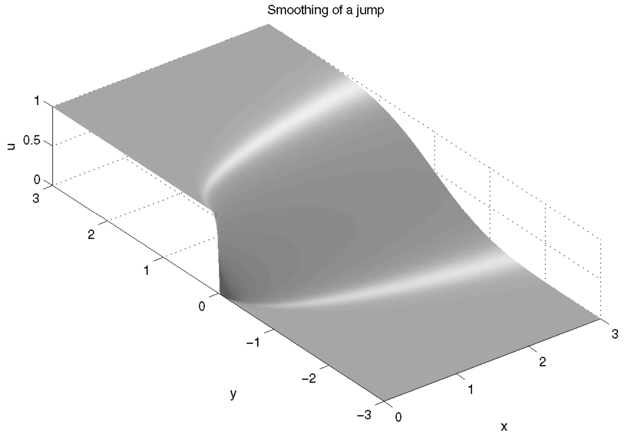
- The solution (found by using the Fourier transform) is,

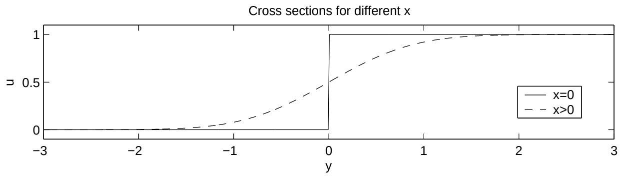
$$\boxed{u(x,y)=\Phi(y/\sqrt{4\beta x})}$$

  where Φ is the distribution function of the normal distribution.



A Case Study: Cont





Conclusion: We need elements of size $O(\sqrt{4\beta x})$ (in y-direction) in order to resolve the gradients.

This is a stringent requirement in many applications.

A Case Study: Boundary Layers

- Consider

$$\boxed{u_x = \beta u_{xx},} \quad 0 < \beta \ll 1.$$

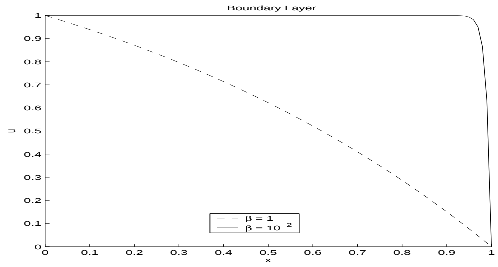
- Boundary conditions of Dirichlet type:

$$u(0) = 1, \quad u(1) = 0.$$

This is a singular perurbation problem (bvp).

Solution

$$u(x) = \frac{1}{1 - e^{-1/\beta}}\left(1 - e^{(x-1)/\beta}\right) \approx 1 - e^{(x-1)/\beta}$$



The gradient at $x = 1$ is $O(1/\beta)$.

Boundary Layers: Cont

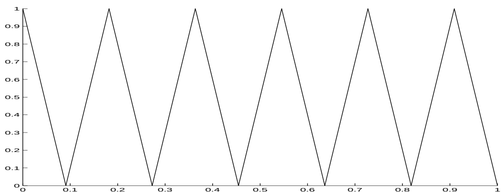
- **Q:** How small must the elements be to resolve this *boundary layer*?
- Use piecewise linear elements. The FEM approximation reads

$$\frac{u_{i+1} - u_{i-1}}{2h} - \frac{\beta}{h^2}(u_{i+1} - 2u_i + u_{i-1}) = 0,$$

  subject to $u_0 = 1$, $u_{N+1} = 0$.
- Consider the reduced problem by taking the limit $\beta \to 0$: $u_{i+1} - u_{i-1} = 0$,

$$u_0 = u_2 = u_4 = \cdots = 1.$$

  - If N is odd: The system contains a contradiction.
  - If N is even: $u_{2j} = 1, u_{2j+1} = 0$.



Boundary Layer Resolution: Quantitative estimation

- The difference equation rewritten:

$$(1 - \frac{2\beta}{h})u_{i+1} + \frac{4\beta}{h}u_i + (-1 - \frac{2\beta}{h})u_{i-1} = 0.$$

- This difference equation admits solutions of the type $u_j = \lambda^j$:

$$(1 - \frac{2\beta}{h})\lambda^2 + \frac{4\beta}{h}\lambda + (-1 - \frac{2\beta}{h}) = 0.$$

  Solutions:

$$\lambda_1 = 1, \quad \lambda_2 = \frac{1 + 2\beta/h}{2\beta/h - 1}.$$

- The general solution

$$u_j = c_1\lambda_1^j + c_2\lambda_2^j = c_1 + c_2\lambda_2^j.$$

  This solution is monotone iff $\lambda_2 > 0$
- We obtain the requirement

$$\boxed{\frac{h}{2\beta} < 1}$$

  i.e., $h = O(\beta)$, which is even more stringent than the requirement before.

Comments, Artificial Diffusion

- This is *not* a stability requirement. It is necessary to reflect a *qualitative* property of the exact solution, namely, to avoid unphysical oscillations.
- If the step size restriction is too hard to fulfill, *increase* β *artificially*:
$$\hat{\beta} := \max(\beta, h/2).$$
Notation: *Artificial diffusion*

Artificial Diffusion In Higher Dimensions

- Consider a non-oscillatory advection-dominated transport problem
$$\mathbf{V} \cdot \nabla u - \varepsilon \Delta u = f.$$
- Discretized by finite elements, *enough diffusion is necessary*. For elements of size h,
  - choose $\hat{\varepsilon} = \max(\varepsilon, |\mathbf{V}|h/2)$.
  - The resulting discretization is first-order accurate.

Artificial Diffusion: How Does It Look Like In 1D?

Discretization:

$$\begin{aligned} f(x_i) &= \frac{V}{2h}(u_{i+1} - u_{i-1}) - \frac{\hat{\varepsilon}}{h^2}(u_{i+1} - 2u_i + u_{i-1}) \\ &= u_{i+1}\left(\frac{V}{2h} - \frac{|V|}{2h}\right) + \frac{|V|}{h}u_i + u_{i-1}\left(-\frac{V}{2h} - \frac{|V|}{2h}\right) \\ &= \begin{cases} \frac{V}{h}(u_i - u_{i-1}), & \text{if } V > 0, \\ \frac{V}{h}(u_{i+1} - u_i), & \text{if } V < 0. \end{cases} \end{aligned}$$

These are one-sided differences depending on the *sign* of V, that is on the direction of the stream.

This is an *upstream* or *upwind* discretization.