Time-dependent problems

- So far, used the finite element framework for discretizing in space
- What about $u_t = \Delta u + f$?
  - Use finite differences in time to obtain a set of recursive spatial problems
  - Solve the spatial problems by the finite element method

Example: diffusion problem

\[ \frac{\partial u}{\partial t} = \alpha \nabla^2 u + f(x,t), \quad x \in \Omega, t \in [0,T] \]
\[ u(x,0) = I(x), \quad x \in \Omega \]
\[ \frac{\partial u}{\partial n} = 0, \quad x \in \partial \Omega, t \in [0,T] \]

A Forward Euler scheme; stages in the discretization

\[ u_n(x,t) \] exact solution of the PDE problem
\[ \Delta u_n(x,t) \] exact solution of time-discrete problem (after applying a finite difference scheme in time)
\[ \Delta u_n(x,t) = \sum_{j=0}^N \psi_j(x), \quad \Delta u_n^{n+1} = \Delta u_n + \Delta t \left( \alpha \nabla^2 u_n + f(x,t_n) \right) \]
\[ u_n^{n+1} \approx u_{n+1} \quad = \sum_{j=0}^N c_n \psi_j(x) \]
\[ R = u_{n+1} - u_n - \Delta t \left( \alpha \nabla^2 u_n + f(x,t_n) \right) \]

A Forward Euler scheme; ideas

\[ D_n u = \alpha \nabla^2 u + f \]
\[ n = 1,2, \ldots, N_t - 1 \]

Solving wrt $u_{n+1}$:
\[ u_{n+1} = u_n + \Delta t \left( \alpha \nabla^2 u_n + f(x,t_n) \right) \]
\[ u_n = \sum_{j=0}^N c_n \psi_j \in V, \quad u_{n+1} = \sum_{j=0}^N c_{n+1} \psi_j \in V \]

Compute $u_n$ from $I$

Compute $u_{n+1}$ from $u_n$ by solving the PDE for $u_{n+1}$ at each time level

A Forward Euler scheme; weighted residual (or Galerkin) principle

\[ R = u_{n+1} - u_n - \Delta t \left( \alpha \nabla^2 u_n + f(x,t_n) \right) \]

The weighted residual principle:
\[ \int_{\Omega} R \phi \, dx = 0, \quad \forall \phi \in W \]

leads to:
\[ \int_{\Omega} \left[ u_{n+1} - u_n - \Delta t \left( \alpha \nabla^2 u_n + f(x,t_n) \right) \right] w \, dx = 0, \quad \forall w \in W \]

Galerkin: $W = V, \quad w = v$
Example using sinusoidal basis functions

A Forward Euler scheme; integration by parts

\[ \int_0^{u_n+1} v dx = \int_0^u \left( u^n + \Delta t \left( \alpha \nabla^2 u^n + f(x, u^n) \right) \right) v dx \]

Integrate by parts of \( f(\nabla^2 u^n) \) v dx:

\[ \int_0^{u_n+1} u x \nabla v dx = - \int_0^u \alpha \nabla u \cdot \nabla v dx + \int_0^u \alpha \Delta u v dx \]

Variational form:

\[ \int_0^{u_n+1} v dx = \int_0^u \left( \alpha \nabla u \cdot \nabla v \right) dx + \Delta t \int_0^u f v dx \quad \forall v \in V \]

Deriving the linear systems

\[
\begin{align*}
\text{At time } n + 1: & \\
\text{or shorter: } & (u, v) = (u_1, v) + \Delta t (\alpha \nabla u_1, \nabla v) + \Delta t (f_n, v) \\
\text{where } & u_1 = \sum_{j=0}^N c_{1,i,j} \psi_j(x) \\
\text{and } & v_1 = \sum_{j=0}^N c_{2,j} \psi_j(x) \\
& \text{for } v = \psi_i, i = 0, \ldots, N
\end{align*}
\]

Structure of the linear systems

\[ M c = M_0 c - \Delta t K_0 c + \Delta t f \]

\[
\begin{align*}
M &= (M_{ij}), \quad M_{ij} = (\psi_i, \psi_j), \quad i, j \in I_n \\
K &= (K_{ij}), \quad K_{ij} = (\nabla \psi_i, \nabla \psi_j), \quad i, j \in I_n \\
f &= ((f(x, \psi_i))_{x \in \Omega}) \\
c &= (c_i)_{i \in I_n} \\
q &= (a_i)_{i \in I_n}
\end{align*}
\]

Computational algorithm

1. Compute \( M \) and \( K \).
2. Interpolate \( u_0 \) by either interpolation or projection.
   - For \( n = 1, 2, \ldots, N \):
     - Compute \( \phi_j = M_{ij} - \Delta t K_{ij} + \Delta t f_{ij} \)
     - Solve \( M c = \phi_j \)
     - Set \( c_i = 0 \)
3. Initial condition:
   - Either interpolation: \( a_j = f(x_j) \) (finite elements)
   - Or projection: Solve \( \sum_{j=0}^N M_{ij} a_j = f(x_j) \), \( i \in I_n \)

Example using sinusoidal basis functions

\[ \frac{\partial u}{\partial x} = 0 \quad x \in [0, 4], \quad t \in [0, 7] \] (1)

\[ u(x, 0) = A \cos(\pi x / 4) + B \cos(20 \pi x / 4), \quad x \in [0, 4] \] (2)

\[ \frac{\partial u}{\partial x} = 0, \quad x = 0, 4, \quad t \in [0, 7] \] (3)

\[ \psi = \cos(\pi x / 4) \]
Approximating the initial condition

\( I(x) \in V \) implies perfect approximation of the initial condition:

\[ c_{1,1} = A, \quad c_{20} = B, \]

while \( c_{ij} = 0 \) for \( i \neq 1, 10 \).

Solving the equation system

\[ D - u = \alpha \Delta^2 u + f \]

\( \Delta^2 = \frac{D^2}{h^2} \Delta = D - \frac{D^2}{h^2} \)

We actually get a closed-form discrete solution:

\[ u_i^n = A(1 - \Delta (\frac{\pi}{L})^2) \cos (\pi i / L) + B (1 - \Delta (\frac{10 \pi}{L})^2) \cos (10 \pi i / L). \]

Computing the \( M \) and \( K \) matrices

Note that \( \psi_i \) and \( \psi'_i \) are orthonormal on \([0, L]\) such that we only need to compute the diagonal elements \( M_{ii} \) and \( K_{ii} \):

\[ M_{0,0} = L, \quad M_{ij} = L/2, \quad i > 0, \quad K_{0,0} = 0, \quad K_{ij} = \frac{\pi^2 i^2}{L^2}, \quad i > 0. \]

Comparing P1 elements with the finite difference method; ideas

- P1 elements in 1D
- Uniform mesh on \([0, L]\) with cell length \( h \)
- No Dirichlet conditions: \( u_i = 0, \ldots, N = N_h - 1 \)
- Have found formulas for \( M \) and \( K \) as the element level
- Have assembled the global matrices
- Have developed corresponding finite difference operator formulas
- \( M : \psi_i \psi_i = \frac{1}{h^2} \Delta \psi_i \psi_i + f \)
- \( K : \psi_i \psi_i = \frac{\pi^2}{L^2} \psi_i \psi_i \)

Discretization in time by a Backward Euler scheme

Backward Euler scheme is time:

\[ (D^\tau - \alpha \Delta^2 u + f(x, t))u^n = u^{n-1} \]

\[ u_i^n = \frac{\Delta t}{\alpha h^2} \psi_i + f(x_i, t_n) \]

\[ u_i^n = \frac{\Delta t}{\alpha h^2} \psi_i + f(x_i, t_n) = u_i^{n-1} - \sum_{j=0}^{N} \frac{\Delta t}{\alpha h^2} \psi_j(x_i), \quad u_i^{n+1} = u_i^{n} - \sum_{j=0}^{N} \frac{\Delta t}{\alpha h^2} \psi_j(x_i) \]
The variational form of the time-discrete problem

\[
\int_{\Omega} \left( u_n^t + \Delta t \nabla u_n^t \cdot \nabla v \right) \, dx = \int_{\Omega} u_{n-1} \, v \, dx + \Delta t \int_{\Omega} f \, v \, dx, \quad \forall v \in V
\]
or

\[
(u_n^t + \Delta t \nabla u_n^t \cdot \nabla v) = (u_{n-1} + \Delta t \nabla v)
\]
The linear system: insert \( u = \sum_{j} c_j \psi_j \) and \( u_1 = \sum_{j} c_1,j \psi_j \),

\[
(M + \Delta t K) c = M c_1 + \Delta t f
\]

Can interpret the resulting equation system as

\[
[D^\gamma (u_1^t + \Delta t \nabla u_1^t \cdot \nabla v)] = \begin{cases}
\alpha \Delta \nabla v, & i \in T_0 \\
\nabla \psi_i, & i \in T_s
\end{cases}
\]

Calculations with P1 elements in 1D

Finite element basis functions

\[
B(x,t_n) = \sum_{j \in I} b_j U_n^j \psi_i = \psi_{\nu(j)}, \quad j \in I_s
\]

Dirichlet boundary conditions

Dirichlet condition at \( x = 0 \) and Neumann condition at \( x = L \):

\[
\begin{align*}
u(x,t) &= \nu_0(x,t), & x \in \partial \Omega_0 \\
\nu_x(x,t) &= g(x,t), & x \in \partial \Omega_N
\end{align*}
\]

Forward Euler time, Galerkin's method, and integrate by parts:

\[
\int_{\Omega} u^{n+1} v \, dx = \int_{\Omega} u^n \cdot \nabla v \, dx + \Delta t \int_{\Omega} f v \, dx + \Delta t \int_{\partial \Omega} g v \, ds
\]

Requirement: \( v = 0 \) on \( \partial \Omega_D \)

Boundary function

\[
u^n(x) = \nu_0(x,t_n) + \sum_{j \in I_s} \phi_j^n(x)
\]

Modification of the linear system; the raw system

- Drop boundary function
- Compute an element-wise Dirichlet conditions
- Modify the linear system to incorporate Dirichlet conditions
- \( T_s \) holds the indices of all nodes \( \{0,1, \ldots, N_n - 1\} \)

\[
\sum_{j \in T_s} \left( \int_{\Omega} \phi_j \, dx \right) q_j = -\sum_{j \in T_s} \left( \int_{\Omega} \phi_j \cdot \nabla \cdot \nabla \psi_i \, dx \right) q_j
\]

- \( \Delta t \) is the time step
Modification of the linear system; setting Dirichlet conditions

\[ Mc = b, \quad b = Mo + \Delta tKo + \Delta t f \]

For each \( k \) where a Dirichlet condition applies, \( u(x_k, t=0) = G_k^{i+1} \).

1. Set row \( k \) to zero and 1 on the diagonal: \( M_k = 0, \quad j \in I_k, M_k,k = 1 \)
2. \( b_k = G_k^{i+1} \)

Or apply the slightly more complicated modification which preserves symmetry of \( M \)

Analysis of the discrete equations

The diffusion equation \( u_t = \alpha u_{xx} \) allows a (Fourier) wave component

\[ u = a e^{ikx}, \quad \phi = e^{-\alpha \Delta x^2} \]

Numerical schemes often allow the similar solution

\[ a_0^i = a e^{ikx} \]

1. \( A \): amplification factor to be computed
2. How good is this \( A \) compared to the exact one?

Amplification factor for the Forward Euler method; results

Introduce \( p = i \Delta x/2 \) and \( C = \alpha \Delta x^2 / 2 \)

\[ A = 1 - 4C \frac{\sin^2 p}{1 - \frac{1}{2} \sin^2 p} \]

(See notes for details)

Stability: \( |A| \leq 1 \)

\[ C \leq \frac{1}{6} \quad \Rightarrow \quad \Delta t \leq \frac{\Delta x^2}{6} \]

Finite differences: \( C \leq \frac{1}{2} \), so finite elements give a stricter stability criterion for the PDE!

Amplification factor for the Backward Euler method; results

Backward Euler discretization in time gives a more complicated coefficients matrix:

\[ At = b, \quad A = M + \Delta t K, \quad b = Mo + \Delta t f \]

1. Set row \( k \) to zero and 1 on the diagonal: \( M_k = 0, j \in I_k, M_k,k = 1 \)
2. \( b_k = G_k^{i+1} \)

Observe: \( A_k = M_k + \Delta t k_{M,k} = 1 + 0, \) so \( q_k = G_k^{i+1} \)

Handy formulas

\[
\begin{align*}
[D^+]A &= \Delta t A \frac{\sin^2 p}{1 - \frac{1}{2} \sin^2 p} A^{-1} \\
[D^-]A &= \Delta t A \frac{\sin^2 p}{1 - \frac{1}{2} \sin^2 p} A^{-1} \\
[D_x, D_x]A &= -A \frac{4}{\Delta x^2} \sin^2 \left( \frac{k_{\Delta x}}{2} \right)
\end{align*}
\]
Amplication factors for smaller time steps; Forward Euler

Amplication factors for smaller time steps; Backward Euler

Method: FE
\[ \frac{C}{2} \text{ or } \frac{1}{6} \]
Method: BE
\[ \frac{C}{2} \text{ or } \frac{1}{6} \]