FEM09 - lecture 4

Johan Jansson

jjan@csc.kth.se

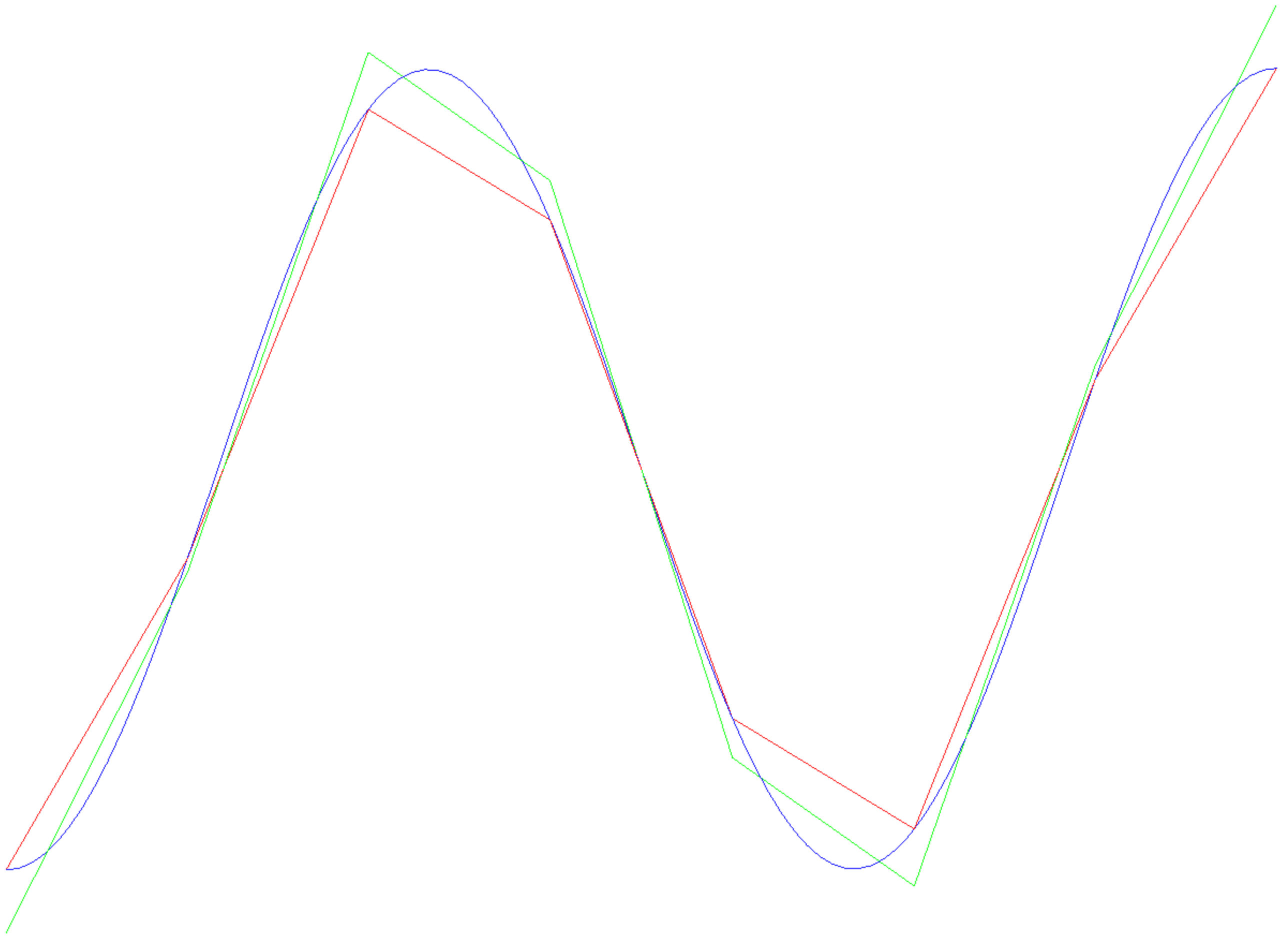
CSC

KTH

Outline

- Computer accounts
- L2-projection, discussion, programming
- Error estimation: a priori, a posteriori
- Poisson in 2D, integration by parts, BCs

L2-projection



Error estimation

Equation: $R(u) = 0$.
Compute solution U, exact solution u.
Error: $e = u - U$
u is unknown, what can we say about the computational error e?

In science/engineering we want to guarantee our solution is acceptable, give a tolerance on error:

$$\|e\| < TOL$$

We also want to compute U with as little computational power as possible.
We will see how a posteriori error estimates/bounds can make this possible.

Error estimation: example

We want to build a cooking pan. Assume the temperature of a hot stove is 400C.
Model the cooking pan with heat eq. (Poisson) and compute an approximate solution. The temperature at the handle of the cooking pan in our computation is 70C (not dangerous). Assume second degree burns start at 80C.

What is the error of the solution?

If the error is 10%, then at most the temperature can be 77C.
If the error is 50%, then at most the temperature can be 105C.

We would like to guarantee that the computational error is smaller than 10%!: $\frac{||e||}{||u||} < 0.1$ ($||e|| < TOL$)

Function space

Remember: A vector space can be constructed with set of

polynomials on domain (a, b) as basis vectors, where function addition and scalar multiplication satisfy the requirements for a vector space.

We can also define an inner product space with the L_2 inner product defined as:

$$(f, g)_{L_2} = \int_\Omega f(x)g(x)dx$$

Lagrange (nodal) Basis

We will use the Lagrange basis: $\{\lambda_i\}_{i=0}^{q}\}$ associated to the distinct points $\xi_0 < \xi_1 < ... < \xi_q$ in (a, b), determined by the requirement that $\lambda_i(\xi_j) = 1$ if $i = j$ and 0 otherwise.

$$\lambda_i(x) = \prod_{j \neq i} \frac{x - \xi_j}{\xi_i - \xi_j}$$

$$\lambda_0(x) = (x - \xi_1)(\xi_0 - \xi_1)$$

$$\lambda_1(x) = (x - \xi_0)/(\xi_1 - \xi_0)$$

Polynomial interpolation

We assume that f is continuous on $[a, b]$ and choose distinct interpolation nodes $a \leq \xi_0 < \xi_1 < \cdots < \xi_q \leq b$ and define a polynomial interpolant $\pi_q f \in \mathcal{P}^q(a, b)$, that interpolates $f(x)$ at the nodes $\{\xi_i\}$ by requiring that $\pi_q f$ take the same values as f at the nodes, i.e. $\pi_q f(\xi_i) = f(\xi_i)$ for $i = 0, ..., q$. Using the Lagrange basis corresponding to the ξ_i, we can express $\pi_q f$ using “Lagrange’s formula”:

$$\pi_q f(x) = f(\xi_0)\lambda_0(x) + f(\xi_1)\lambda_1(x) + \cdots + f(\xi_q)\lambda_q(x) \quad \text{for } a \leq x \leq b$$

Interpolation error

Mean value theorem/Taylor:

$$f(x) = f(\xi_0) + f'(\eta)(x - \xi_0) = \pi_0 f(x) + f'(\eta)(x - \xi_0)$$

for some η between ξ_0 and x, so that

$$|f(x) - \pi_0 f(x)| \leq |x - \xi_0| \max_{[a,b]} |f'| \quad \text{for all } a \leq x \leq b$$

Or:

$$\|f - \pi_0 f\|_{L_2(a,b)} \leq C_i(b - a)\|f'\|_{L_2(a,b)}$$

L_2 projection

We seek a polynomial approximate solution $U \in P^q(a,b)$ to the equation:

$$R(u) = u - f = 0, \quad x \in (a,b)$$

where f in general is not polynomial, i.e. $f \notin P^q(a,b)$.
This means $R(U)$ can in general not be zero. The best we can hope for is that $R(U)$ is orthogonal to $P^q(a,b)$ which means solving the equation:

$$(R(U), v)_{L_2} = (U - f, v)_{L_2} = 0, \quad x \in \Omega, \quad \forall v \in P^q(a,b)$$

Error estimate

The orthogonality condition means the computed L_2 projection U is the best possible approximation of f in $P^q(a,b)$ in the L_2 norm:

$$\|f-U\|^2 = (f-U, f-U) =$$
$$(f-U, f-v) + (f-U, v-U) =$$
$$[v-U \in P^q(a,b)] = (f-U, f-v) \leq \|f-U\|\|f-v\|$$
$$\Rightarrow$$
$$\|f-U\| \leq \|f-v\|, \quad \forall v \in P^q(a,b)$$

Error estimate

Since $\pi f \in P^q(a, b)$, we can choose $v = \pi f$ which gives:

$$\|f - U\| \leq \|f - \pi f\| \leq C_i(b - a)\|f'\|_{L_2(a,b)}$$

i.e. we can use an interpolation error estimate since it bounds the projection error.

A priori for Poisson

$C_i(b-a)\|f'\|_{L_2(a,b)}$ vs. TOL

Integration by parts in 2D

$$\int_\Omega D_{x_i} vw dx = \int_\Gamma vwn_i ds - \int_\Omega vD_{x_i} w dx, \quad i = 1, 2, ..., d$$

$$\int_a^b D_x vw dx = [vw]_a^b - \int_a^b vD_x w dx \tag{1}$$

$$\int_\Omega D_{x_i} v D_{x_i} w dx = \int_\Gamma vD_{x_i} wn_i ds - \int_\Omega vD_{x_i} D_{x_i} w dx \tag{2}$$

$$\int_\Omega \nabla v \cdot \nabla w dx = \int_\Gamma v(\nabla w \cdot n) ds - \int_\Omega v\Delta w dx \tag{3}$$