

Diffusion/Elliptic Problems

Weak Formulation. Given V , $F : V \rightarrow \mathbb{R}$ functional and $a : V \times V \rightarrow \mathbb{R}$ bilinear form,

$$\text{find } u \in V : a(u, v) = F(v) \quad \forall v \in V$$

Lax-Milgram Lemma. Assume that:

1. V Hilbert space with $\|\cdot\|_V$ and $\langle \cdot, \cdot \rangle$
2. $F \in V'$, i.e. $|F(v)| \leq \|F\|_{V'} \|v\|_V$
3. a cont., i.e. $|a(u, v)| \leq M \|u\|_V \|v\|_V$
4. a coercive, i.e. $a(v, v) \geq \alpha \|v\|_V^2$

Then: $\exists!$ u sol. di WF, and $\|u\|_V \leq \|F\|_{V'}/\alpha$

Galerkin Approximation. If you can build $V_h \subset V$ s.t. $\dim V_h = N_h < \infty$ ($\Rightarrow V_h$ closed subspace), then WF becomes G:

$$\text{find } u_h \in V_h : a(u_h, v_h) = F(v_h) \quad \forall v_h \in V_h$$

- *well-posedness* follows from LM
- *stability* is the continuous dependence from data in LM
- *consistency* \equiv **Galerkin Orthog.** GO:

$$a(u - u_h, v_h) = 0 \quad \forall v_h \in V_h$$

- if we assume **space saturation**

$$\lim_{h \rightarrow 0} \inf_{v_h \in V_h} \|v - v_h\|_V = 0 \quad \forall v \in V$$

then *convergence* \equiv **Céa Lemma**:

$$\|u - u_h\|_V \leq \frac{M}{\alpha} \inf_{v_h \in V_h} \|u - v_h\|_V$$

Proof. See whiteboards.

Last but not least: Problem G is equivalent to the following linear system LS of equations:

$$\text{find } \mathbf{u} \in \mathbb{R}^{N_h} : A\mathbf{u} = \mathbf{F}$$

where $A \in \mathbb{R}^{N_h \times N_h}$, $\mathbf{F} \in \mathbb{R}^{N_h}$.

Proof. See whiteboards.

Moral of the story: V_h must be chosen to ensure the saturation assumption and the computation of the integrals $A_{ij} = a(\phi_j, \phi_i)$ and $F_i = F(\phi_i)$.

The Finite Element Method. We consider a domain Ω with polygonal shape (we do not take into account error due to the approximation of a non-polygonal domain with a FE grid). A **mesh** is a finite cover of K non-overlapping triangles, also called *triangulation*: $\mathcal{T} = \bigcup K$. Actually: in 1D, intervals; in 2D, triangles but we'll see also quadrilaterals; in 3D, tetrahedra but we'll see also hexahedra.