Courses/FEM/modules/functions

From Icarus
< Courses | FEM

Contents

Function Approximation

Precondition

- Science

Theory

Function space

A vector space can be constructed with functions (a set of polynomials on a domain Ω for example) as vectors, where function addition and scalar multiplication satisfies the requirements for a vector space.

We can also define an inner product space with the L2 inner product defined as

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

The inner product generates the norm

$$\|f\|_{L_2} = \sqrt{(f, f)}$$

Just like in R^d we define orthogonality between two vectors as

$$(f, g)_{L_2} = 0$$

[Polynomial space]

[Piecewise polynomials]

Polynomial interpolation

We can construct an interpolant $\pi f \in P^q(a,b)$ of the function f by requiring that

$$\begin{gathered}\pi f(\xi_i) = f(\xi_i), \quad i \in [0,\dots,q] \\ a \leq \xi_0 < \dots < \xi_q \leq b\end{gathered}$$

Assume that f has $q+1$ continuous derivatives in (a,b) and let $\pi f \in P^q(a,b)$ interpolate f at the points $a \leq \xi_0 < \dots < \xi_q \leq b$. Then for $a \leq x \leq b$

$$|f(x) - \pi f(x)| \leq |\frac{(x-\xi_0)\dots(x-\xi_q)}{(q+1)!}| max_{[a,b]}|D^{q+1}f|$$

L2 (orthogonal) projection

An L2 projection Pf is a projection of a function f in the function space A to the function space B. We can think of this problem as solving the equation

$$R(Pf) = Pf - f = 0, \quad x \in \Omega$$

However, since Pf and f belong to different function spaces, the residual $R(Pf)$ can in general not be zero. The best we can hope for is that $R(Pf)$ is orthogonal to B, which means solving the equation

$$(Pf - f, v) = 0, \quad x \in \Omega, \quad \forall v \in B$$

The L_2 projection is the best possible approximation

The orthogonality condition means that Pf is the best possible approximation in B, i.e. we cannot pick an object $v \in B$ which is a better approximation that Pf in the L_2 norm

$$\|f - Pf\|^2 = (f - Pf, f - Pf) = (f - Pf, f - v) + (f - Pf, v - Pf) = [v - Pf \in B] = (f - Pf, f - v) \leq \|f -$$
$$\Rightarrow$$
$$\|f - Pf\| \leq \|f - v\|, \quad \forall v \in B$$

L_2 projection error estimate

Since $\pi f \in B$, we can choose $v = \pi f$ which gives

$$\|f - Pf\| \leq \|f - \pi f\|$$

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

Software

To compute the L2 projection we want to solve the equation

$$(R(u), v) = (u, v) - (f, v) = 0$$

. We identify the terms with u and v as **bilinear forms**

$$a(u, v)$$

and the terms with only v as **linear forms**

$$L(v)$$

.

We can thus define the representation of the equation in FEniCS as:

```
V = FunctionSpace(mesh, "CG", 1)
u = TrialFunction(V)
v = TestFunction(V)

a = (u * v) * dx
L = (f * v) * dx
```

Construct discrete functions and compute functionals. Compute an L_2 projection of a given function and compute the functional (norm) $\|f - Pf\|$.

A form without u and v is identified as M (a functional), and can be used to compute a norm of a known function for example:

```
f = Function(V)

M = (f * f) * dx
```

Assembly:

A form `q` (a, L or M in this case) is assembled by:

```
T = assemble(q)
```

This is the same basic algorithm for assembling a matrix (from a bilinear form), vector (from a linear form) or scalar (from a functional).

Vector indexing:

You can get and set values of a vector by a standard bracket notation:

```
x = Vector(3)

a = x[1]
x[2] = a + 4
```

To get the vector of coefficients ξ in the linear combination $Pf = \sum_j^M \xi_j \phi_j$ you do:

```
x = Pf.vector()
```

Postcondition

You should now be familiar with:

- L2 inner product
- interpolation
- L2 projection
- Mesh size function h

Exercises

- CDE: 5.7, 5.8, 5.14, 5.17, 5.1/5.21

Problem

We let $\mathcal{P}^q(a, b)$ denote the set of polynomials $p(x) = \sum_{i=0}^q c_i x^i$ of degree at most q on an interval (a, b), where the $c_i \in R$ are called the coefficients of $p(x)$. We recall that two polynomials $p(x)$ and $r(x)$ may be added to give a polynomial $p + r$ defined by $(p + r)(x) = p(x) + r(x)$ and a polynomial $p(x)$ may be multiplied by a scalar α to give a polynomial αp defined by

$(\alpha p)(x) = \alpha p(x)$. Similarly, $\mathcal{P}^q(a, b)$ satisfies all the requirements to be a vector space where each ``vector *is a* particular polynomial function $p(x)$.

Prove this claim.

Problem

Compute formulas for the linear interpolant of a continuous function f through the points a and (b + a)/2. Plot the corresponding Lagrange basis functions.

Problem

Write down the polynomial of degree 3 that interpolates $sin(x)$ at $\xi_0 = 0, \xi_1 = \pi/6, \xi_2 = \pi/4$, and $\xi_3 = \pi/3$, and plot p_3 and sin on $[0, \pi/2]$.

Examination

1.1

Construct linear Lagrange basis functions with the nodes in the vertices on a tetrahedron in 3D.

1.2

Using FEniCS:

Compute the L2 projection Pf of a function f (a trigonometric function for example) in 1D or 2D on a space of piecewise linear polynomials with just a few points. Compute the L2 norm of the error $||f - Pf||$ as a functional in FEniCS. Try to choose better values of the coefficients ξ_j in $Pf = \sum_j^M \xi_j \phi_j$. Are you able to reduce the error?

Note: Expressions with f will be integrated with quadrature (f is represented as a finite element function on each cell). Choose a higher order representation for f to remove the effect of the quadrature error on the result, i.e.:

```
V3 = FunctionSpace(mesh, "CG", 3)
f = Expression("0.1*pi*cos(3.0*pi*x[0])", element=V3.ufl_element())
```

represents f as a cubic rather than linear function when integrating.

TODO

- Polynomial space
- Piecewise polynomials
- Software part