1. *hp*-refinements

In Assignment 3, we have solved the 1d Poisson problem in just one element with different polynomial degree p. We have seen that, when we increase p (also denoted by N somewhere), the solution converges to the exact solution fast. This refinement is called p-refinement.

Another common refinement is the h-refinement. h usually refers to the element size. For example, if the domain is $[-1, 1]$ and it is uniformly divided into K elements, we have $h = 2/K$. If we increase K, h gets smaller, and the mesh is refined. Therefore, the solution will converge as well. This is the h-refinement.

An hp-refinement scheme is such a scheme that we can do both h- and p-refinement.

2. 1d Poisson problem with two different boundary conditions on the left and right boundaries

We still consider the exact solutions:

$$\phi = -\sin(\alpha\pi x),$$

$$u = \nabla\phi = -\alpha\pi\cos(\alpha\pi x),$$

$$f = -\nabla\cdot u = -\alpha^2\pi^2\sin(\alpha\pi x),$$

where $\alpha \in \{1, 2, 3, 4, ...\}$. The domain this time is set to $\Omega = [0, 1]$, not $[-1, 1]$ anymore, for simplicity.

The problem is then given as: In Ω, given f all over the domain and boundary conditions:

$$\phi(0) = 0,$$

and

$$u(1) = -\alpha\pi\cos(\alpha\pi),$$

solve the Poisson equation (See Assignment 3).

3. Transformations

Let the domain $\Omega = [0, 1]$ be divided into K uniform elements:

$$\Omega_k = [kh, (k+1)h], \quad k \in \{0, 1, ..., K-1\},$$

where $h = 1/K$ is the element size. We can easily derive the linear transformation, Φ_k, from the reference domain $\Omega_{\text{ref}} := \xi \in [-1, 1]$ to Ω_k:

$$\Phi_k := \Omega_{\text{ref}} \to \Omega_k.$$

With this transformation, we can switch between the element Ω_k and the reference domain Ω_{ref}.

3.1. From Ω_k to Ω_{ref}

Start from this moment, we use a bit differential forms to simplify the derivations. We first see how to transform from element Ω_k to the reference domain. In Ω_k, the ϕ is interpreted as a 1-form:

$$\phi^{(1)} = \phi(x)\ \mathrm{d}x,$$

and the u is interpreted as a 0-form:

$$u^{(0)} = u(x).$$

We can transform them back to the reference domain by:

$$\phi_{\text{ref}}^{(1)} = \left[\phi(\Phi_k(\xi))\ \frac{\mathrm{d}x}{\mathrm{d}\xi}\right] \mathrm{d}\xi,$$

$$u_{\text{ref}}^{(0)} = u(\Phi_k(\xi)).$$

3.2. From Ω_{ref} to Ω_k

Suppose we have $u_{\text{ref}}^{(0)}$ and $\phi_{\text{ref}}^{(1)}$ in Ω_{ref}:

$$u_{\text{ref}}^{(0)} = u_{\text{ref}}(\xi)$$

and

$$\phi_{\text{ref}}^{(1)} = \phi_{\text{ref}}(\xi)\ \mathrm{d}\xi.$$

Their transformation in Ω_k are

$$u^{(0)} = u_{\text{ref}}(\Phi_k^{-1}(x))$$

and

$$\phi^{(1)} = \left[\phi_{\text{ref}}(\Phi_k^{-1}(x))\frac{\mathrm{d}\xi}{\mathrm{d}x}\right] \mathrm{d}x.$$

4. Discrete 1d Poisson problem

Suppose in all elements, we use polynomial basis functions at $p = N$. In Ω_k, we can discretize f onto

$$f_k^h = \sum_{i=1}^{N} f_k^i e_i^k(x)\ \mathrm{d}x,$$

where

$$f_k^i = \int_{\Phi_k(\xi_{i-1})}^{\Phi_k(\xi_i)} f(x)\ \mathrm{d}x.$$

With the transformations, we can get the discrete 1d Poisson problem in Ω_k:

$$\begin{Bmatrix} \mathbb{M}_k^u & (\mathbb{M}_k^\phi \mathbb{E})^\mathsf{T} \\ \mathbb{M}_k^\phi \mathbb{E} & \end{Bmatrix} \begin{Bmatrix} \vec{u}_k \\ \vec{\phi}_k \end{Bmatrix} = \begin{Bmatrix} 0 \\ -\mathbb{M}_k^\phi \vec{f}_k \end{Bmatrix}, \tag{1}$$

where $\vec{f}_k = \left\{f_k^1, f_k^2, \cdots, f_k^N\right\}^\mathsf{T}$,

$$\mathbb{M}^u_{k;i,j} = \int_{-1}^{1} \frac{\mathrm{d}x}{\mathrm{d}\xi} h_i(\xi) h_j(\xi)\ \mathrm{d}\xi,$$

$$\mathbb{M}^\phi_{k;i,j} = \int_{-1}^{1} \frac{\mathrm{d}\xi}{\mathrm{d}x} e_i(\xi) e_j(\xi)\ \mathrm{d}\xi,$$

and $\mathbb{E}$ is the incidence matrix we have in assignment 3.

We can do this for all elements, Ω_k, $k \in \{0, 1, \cdots, K-1\}$ and get following K local systems.

$$A_k \vec{x}_k = b_k.$$

We them can assemble all these local matrices A_k into a global one A:

$$A = \begin{Bmatrix} A_0 & 0 & 0 & \cdots & 0 \\ 0 & A_1 & 0 & \cdots & 0 \\ 0 & 0 & A_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & A_{K-1} \end{Bmatrix}.$$

and assemble local vectors $\vec{b}_k$ into a global one $\vec{b}$:

$$\vec{b} = \begin{Bmatrix} \vec{b}_0 \\ \vec{b}_1 \\ \vdots \\ \vec{b}_{K-1} \end{Bmatrix}.$$

And the first block of the final global system is

$$A\vec{x} = \vec{b}.$$

Recall the process that how to connect two neighboring elements and to impose boundary condition. We get the second block of the global system:

$$C \left\{\vec{x}, \vec{\lambda}\right\}^\mathsf{T} = \vec{d}. \tag{2}$$

For example, if the domain have 3 elements and $N = 2$, and the global numbering of all dofs are as shown in Fig. 1. Then, we will have

$$u_2 - u_5 = 0,$$

$$u_7 - u_{10} = 0,$$

$$\lambda_0 = \phi(0),$$

$$u_{12} = u(1).$$

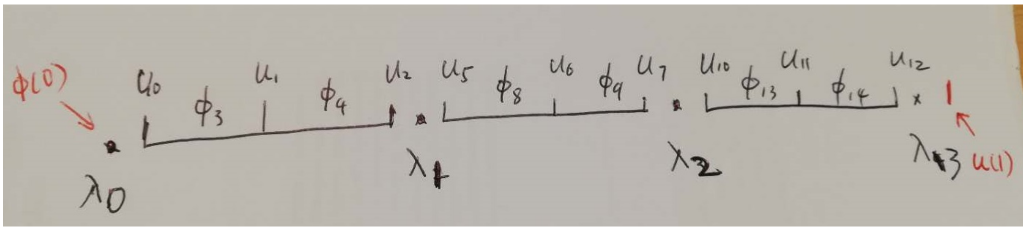

Figure 1: The elements and connections.

Then, see (2), C is

$$\begin{Bmatrix} 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{Bmatrix},$$

and $\left\{\vec{x}, \vec{\lambda}\right\}^{\mathsf{T}}$ is

$$\begin{Bmatrix} u_0 & u_1 & u_2 & \phi_3 & \phi_4 & u_5 & u_6 & u_7 & \phi_8 & \phi_9 & u_{10} & u_{11} & u_{12} & \phi_{13} & \phi_{14} & \lambda_0 & \lambda_1 & \lambda_2 & \lambda_3 \end{Bmatrix}^{\mathsf{T}},$$

and $\vec{d}$ is

$$\begin{Bmatrix} 0 & 0 & \phi(0) & u(1) \end{Bmatrix}^{\mathsf{T}}.$$

Let $C_1 = C[:,\ 0:-4]$, $C_2 = C[:,\ -4:]$. We can get the global system:

$$\begin{Bmatrix} A & C_1^{\mathsf{T}} \\ C_1 & C_2 \end{Bmatrix} \begin{Bmatrix} \vec{x} \\ \vec{\lambda} \end{Bmatrix} = \begin{Bmatrix} \vec{b} \\ \vec{d} \end{Bmatrix}.$$

If we solve this system, we obtain all unknowns, u_i, ϕ_i and λ_i, then we can reconstruct u^h and ϕ^h in all elements.