Ecole d'ingénieurs Sup Galilée
Energétique - Informatique - Instrumentation
Mathématiques Appliquées et Calcul Scientifique
Télécommunications et Réseaux

Ingénieurs MACS 2 - TP F.E.M. (S8)

Eléments finis $\mathbb{P}_1$-Lagrange en dimension $\geqslant 1$: épisode 4

Local vs Global

François Cuvelier

Laboratoire d'Analyse Géométrie et Applications
Institut Galilée
Université Paris XIII.

2024/03/08

Prérequis

épisodes 1 à 3.

Objectif :

- maitriser la notion de numérotation locale et globale des points d'un maillage élémentaire,
- passage des matrices de Masse locales aux matrices de Masse globales,
- maitriser l'écriture de fonctions Matlab/Octave avec des paramètres optionnels en entrée et un nombre variable d'arguments en sortie.

Plan

E_h maillage élémentaire d'un maillage Ω_h.

- **restriction :** Soit $v_h \in V(\Omega_h)$, comment construire $u_h \in V(E_h)$ tel que $u_h = v_{h|E_h}$
- **relèvement/prolongement :** Soit $u_h \in V(E_h)$, comment construire $v_h \in V(\Omega_h)$ tel que $v_{h|E_h} = u_h$

Rappel, épisode 1 : numérotation **globale** d'un maillage

Soit E_h est un maillage élémentaire d'un maillage Ω_h, et on note

$$n_q = E_h.n_q,\ q = E_h.q, me = E_h.me,\ toGlobal = E_h.toGlobal, \ldots$$

On a alors

$$q = \Omega_h.q(:, toGlobal) \quad \Omega_h.n_q = E_h.nqGlobal$$

Soit $i \in [\![1, E_h.n_q]\!]$ et $r = toGlobal(i)$, $(r \in [\![1, \Omega_h.n_q]\!])$.

$$E_h.\varphi_i = \Omega_h.\varphi_{r|E_h} \quad \Leftrightarrow \quad E_h.\varphi_i(\boldsymbol{x}) = \Omega_h.\varphi_r(\boldsymbol{x}),\ \forall \boldsymbol{x} \in E_h$$

Notons $\mathcal{I}_G = E_h.toGlobal$ (indices globaux), $\mathcal{I}_L = [\![1, E_h.n_q]\!]$ (indices locaux) et $\mathcal{F}$ la **bijection**

$$\begin{array}{rccc} \mathcal{F} & : & \mathcal{I}_L & \longrightarrow & \mathcal{I}_G \\ & & i & \longmapsto & \mathcal{I}_G(i) \end{array}$$

$$\forall i \in \mathcal{I}_G,\quad \Omega_h.\varphi_{i|E_h} = E_h.\varphi_j, \quad \text{avec } j = \mathcal{F}^{-1}(i).$$

$$\forall i \in \mathcal{I}_G,\quad \Omega_h.\varphi_{i|E_h} = E_h.\varphi_j, \quad \text{avec } j = \mathcal{F}^{-1}(i).$$

Soient $\boldsymbol{U} \in \mathbb{R}^{E_h.n_q}$ et $u_h \in V(E_h)$. Alors $\forall \boldsymbol{x} \in E_h$,

$$\begin{aligned} u_h(\boldsymbol{x}) &= \sum_{j=1}^{E_h.n_q} U_j(E_h.\varphi_j)(\boldsymbol{x}) = \sum_{j \in \mathcal{I}_L} U_j(E_h.\varphi_j)(\boldsymbol{x}) \\ &= \sum_{i \in \mathcal{I}_G} U_{\mathcal{F}^{-1}(i)}(\Omega_h.\varphi_{\mathcal{F}^{-1}(i)})(\boldsymbol{x}) \end{aligned}$$

On note $\mathcal{I}_G^c = [\![1, \Omega_h.n_q]\!] \backslash \mathcal{I}_G$.

$$\boldsymbol{V} \in \mathbb{R}^{\Omega_h.n_q} : \quad \forall i \in \mathcal{I}_G,\quad V_i = U_{\mathcal{F}^{-1}(i)}, \quad \text{et} \quad \forall i \in \mathcal{I}_G^c,\quad V_i = 0$$

$$v_h(\boldsymbol{x}) = \sum_{i=1}^{\Omega_h.n_q} V_i(\Omega_h.\varphi_i)(\boldsymbol{x}) = \sum_{i \in \mathcal{I}_G \cup \mathcal{I}_G^c} V_i(\Omega_h.\varphi_i)(\boldsymbol{x}) = \sum_{i \in \mathcal{I}_G} V_i(\Omega_h.\varphi_i)(\boldsymbol{x})$$

$$\forall \boldsymbol{x} \in E_h,\quad v_h(\boldsymbol{x}) = u_h(\boldsymbol{x}) \qquad \rightarrow v_h \text{ prolongement par 0 de } u_h \text{ dans } V(\Omega_h)$$

Exo: Soient Th un maillage (siMesh object) et Eh un de ses maillages élémentaires (siMeshElt object). On a u=@(x,y) cos(pi*x).*sin(pi/3*y), Ug=Th.eval(u) et Ul=Eh.eval(u)

1. Quelles sont les dimension de Ug et Ul?
2. Quelle est la relation entre Ul et Ug?
3. Comment s'écrit l'interpolation u_h de u sur Ω_h = Th en fonction de Ug?
4. Déterminer la restriction w_h de u_h à $V(E_h)$ à partir de Ug.
5. Comment s'écrit l'interpolation v_h de u sur Eh en fonction de Ul?
6. Déterminer le prolongement par 0 dans Th de la fonction de v_h à partir de Ul.

☞ voir ep04.ResProDom et ep04.ResProBd

Correction: Soient Th un maillage (siMesh object) et Eh un de ses maillages élémentaires. On a u=@(x,y) cos(pi*x).*sin(pi/3*y), Ug=Th.eval(u) et Ul=Eh.eval(u)

1. Quelles sont les dimension de Ug et Ul?
   ☞ resp. $(\mathrm{Th}.n_q, 1)$ et $(\mathrm{Eh}.n_q, 1)$
2. Quelle est la relation entre Ul et Ug?
   ☞ Ul==Ug(Eh.toGlobal)
3. Comment s'écrit l'interpolation u_h de u sur Ω_h = Th en fonction de Ug?
   ☞ $u_h(\boldsymbol{x}) = \sum_{i=1}^{\mathrm{Th.nq}} \mathrm{Ug(i)} \times \mathrm{Th}.\varphi_i(\boldsymbol{x})$
4. Déterminer la restriction w_h de u_h à $V(E_h)$ à partir de Ug.
   ☞ W=Ug(Eh.toGlobal) et $w_h(\boldsymbol{x}) = \sum_{j=1}^{\mathrm{Eh.nq}} \mathrm{W(j)} \times \mathrm{Eh}.\varphi_j(\boldsymbol{x})$
5. Comment s'écrit l'interpolation v_h de u sur Eh en fonction de Ul?
   ☞ $v_h(\boldsymbol{x}) = \sum_{j=1}^{\mathrm{Eh.nq}} \mathrm{Ul(j)} \times \mathrm{Eh}.\varphi_j(\boldsymbol{x})$
6. Déterminer p_h, le prolongement par 0 dans Th de la fonction de v_h, à partir de Ul.
   ☞ P=zeros(Th.nq,1);P(Eh.toGlobal)=Ul; et $p_h(\boldsymbol{x}) = \sum_{i=1}^{\mathrm{Th.nq}} \mathrm{P(i)} \times \mathrm{Th}.\varphi_i(\boldsymbol{x})$

On note $P^0_{E_h} : V(E_h) \longrightarrow V(\Omega_h)$ et $P^{0,D}_{E_h} : \mathbb{R}^{E_h.n_q} \longrightarrow \mathbb{R}^{\Omega_h.n_q}$ respectivement le prolongement/relèvement par 0 d'une fonction de $V(E_h)$ dans $V(\Omega_h)$ et le relèvement discret associé. Soit $v_h \in V(E_h)$ donné par $\boldsymbol{V} \in \mathbb{R}^{E_h.n_q}$:

$$v_h = \sum_{i=1}^{E_h.n_q} V_i.(E_h.\varphi_i)$$

alors

$$P^0_{E_h}(v_h) = \sum_{j=1}^{\Omega_h.n_q} U_j(\Omega_h.\varphi_j) \quad \text{et} \quad P^{0,D}_{E_h}(\boldsymbol{V}) = \boldsymbol{U}$$

avec $\boldsymbol{U} \in \mathbb{R}^{\Omega_h.n_q}$ tel que $U_{\mathcal{I}_G} = V$ et $U_{\mathcal{I}^c_G} = 0$. $\quad (\mathcal{I}_G = E_h.\mathrm{toGlobal},\ \ \mathcal{I}^c_G = [\![1, \Omega_h.n_q]\!] \backslash \mathcal{I}_G)$

Exo:

1. Ecrire la fonction relevement retournant le vecteur $\boldsymbol{U}$ à partir du $\boldsymbol{V}$ associé à un maillage élémentaire Eh provenant d'un maillage Th.
2. Ecrire la fonction restriction , réciproque de la fonction précédente.

Ces fonctions seront enregistrées dans le répertoire `+siMeshElt`.

Plan

rappels : $\mathcal{I}_G = E_h.\mathrm{toGlobal}$ (indices globaux), $\mathcal{I}_L = [\![1, E_h.n_q]\!]$ (indices locaux) et $\mathcal{F}$ la **bijection**

$$\begin{array}{rccc} \mathcal{F} & : \mathcal{I}_L & \longrightarrow & \mathcal{I}_G \\ & i & \longmapsto & \mathcal{I}_G(i) \end{array}$$

$$\forall i \in \mathcal{I}_G,\quad \Omega_h.\varphi_{i|E_h} = E_h.\varphi_j, \quad \text{avec } j = \mathcal{F}^{-1}(i).$$

Soit $\mathbb{A}^{h,L} \in \mathcal{M}_{E_h.n_q}(\mathbb{R})$ une matrice d'assemblage (locale), $\forall (i,j) \in \mathcal{I}_L^2$,

$$\mathbb{A}^{h,L}_{i,j} \stackrel{\text{def}}{=} \mathcal{A}_h(E_h.\varphi_j, E_h.\varphi_i) = \mathcal{A}_h(\Omega_h.\varphi_{\mathcal{F}(j)|E_h}, \Omega_h.\varphi_{\mathcal{F}(i)|E_h})$$

$\mathbb{A}^{h,G} \in \mathcal{M}_{\Omega_h.n_q}(\mathbb{R})$ la matrice d'assemblage (globale)

$$\mathbb{A}^{h,G}_{r,s} \stackrel{\text{def}}{=} \begin{cases} \mathcal{A}_h(\Omega_h.\varphi_{s|E_h}, \Omega_h.\varphi_{r|E_h}), & \forall (r,s) \in \mathcal{I}_G^2, \\ 0 & \text{sinon.} \end{cases}$$

Exo: Avec les notations précédentes, soient $\boldsymbol{U}^G \in \mathbb{R}^{\Omega_h.n_q}$, $\boldsymbol{U}^L = \boldsymbol{U}^G(\mathcal{I}_G) \in \mathbb{R}^{E_h.n_q}$ et $\boldsymbol{V}^G = \mathbb{A}^{h,G}\boldsymbol{U}^G$.

1. Montrer que $\boldsymbol{V}^G(\mathcal{I}^c_G) = 0$.
2. Montrer que $\boldsymbol{V}^G(\mathcal{I}_G) = \mathbb{A}^{h,L}\boldsymbol{U}^L$.

☞ voir ep04.MatGloDom et ep04.MatGloBd

Sous Matlab/Octave

```
% Matrices locales
Ml=FEM.Masse(Eh.nq,Eh.me,Eh.vols);
Kl=FEM.Rigidite(Eh.nq,Eh.me,Eh.vols,Eh.gradBaCo);
% Matrices globales
Ig=Eh.toGlobal;
Mg=sparse(Th.nq,Th.nq); Mg(Ig,Ig)=Ml;
Kg=sparse(Th.nq,Th.nq); Kg(Ig,Ig)=Kl;
```

Exo: Ecrire les fonctions d'assemblage Masse et Rigidite permettant de retourner, au choix, les matrices de Masse et de Rigidité **locales** ou **globales** associées à un maillage élémentaire Eh provenant d'un maillage Th. Ces fonctions utiliseront un inputParser voir code ci-dessous. Ces fonctions seront enregistrées dans le répertoire `+siMeshElt`.

```
function A=Masse(Eh,varargin)
  % M=siMeshElt.Masse(Th.sTh{3},'local',false)
  assert(fc_simesh.is_siMeshElt(Eh))
  p = inputParser;
  p.addParameter('local', true, @islogical );
  p.parse(varargin{:});
  if p.Results.local   % optional parameter
  ...
```

Plan

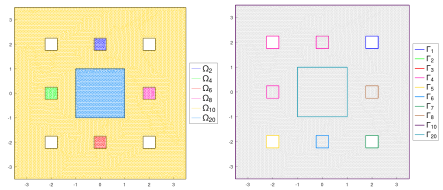


Figure: Exemple de maillage 2D: fichier `square6dom4holes.geo`

Le maillage est donné par Th (objet fc_siMesh.siMesh). L'objectif ici est d'utiliser des matrices de Masse pour approcher des intégrales.

Exo: Pour chacune des intégrales, proposer une méthode utilisant des matrices globales et une autre utilisant des matrices locales.

$$\int_{\Omega_{20}} u(\mathrm{q})v(\mathrm{q})d\mathrm{q}, \quad \int_{\Omega_{[20,2]}} u(\mathrm{q})v(\mathrm{q})d\mathrm{q}, \quad \int_{\Omega} u(\mathrm{q})v(\mathrm{q})d\mathrm{q},$$

$\Omega_{[20,2]} = \Omega_{20} \cup \Omega_2 \quad \Omega = \Omega_{[2:2:10,20]}$ (notation Matlab)

On pourra prendre, par ex., $u(x,y) = \cos(x+y-\pi/3)$ et $v(x,y) = \sin(x-y+1)$, dont l'intégrale exacte sur $[a,b]\times[c,d]$ est donnée ci-dessous.

```
Iexfun=@(a,b,c,d) -c*(cos(1-pi/3+2*a)-cos(1-pi/3+2*b))/4 + d*(cos(1-pi/3+2*a)-cos(1-pi/3+2*b))/4 + (a-b)*cos(2*c-1-pi/3)/4 - (a-b)*cos(2*d-1-pi/3)/4;
```

☞ voir ep04.IntegralDom01

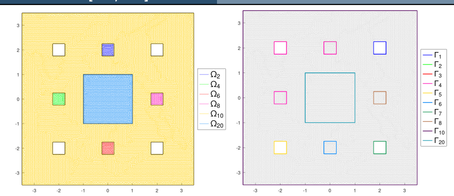


Figure: Exemple de maillage 2D: fichier `square6dom4holes.geo`

Le maillage est donné par Th (objet fc_siMesh.siMesh). L'objectif ici est d'utiliser des matrices de Masse pour approcher des intégrales sur les bords.

Exo: Pour chacune des intégrales, proposer une méthode utilisant des matrices globales et une autre utilisant des matrices locales.

$$\int_{\Gamma_{20}} u(\boldsymbol{\sigma})v(\boldsymbol{\sigma})d\boldsymbol{\sigma}, \quad \int_{\Gamma_{[10,20]}} u(\boldsymbol{\sigma})v(\boldsymbol{\sigma})d\boldsymbol{\sigma}, \quad \int_{\Gamma} u(\boldsymbol{\sigma})v(\boldsymbol{\sigma})d\boldsymbol{\sigma},$$

$\Gamma_{[10,20]} = \Gamma_{10} \cup \Gamma_{20} \quad \Gamma = \Gamma_{[1:2:7,10,20]}$ (notation Matlab)

On pourra prendre, par ex., $u(x,y) = \cos(x+y-\pi/3)$ et $v(x,y) = \sin(x-y+1)$.

$$\int_a^b u(x,y)v(x,y)dx \Longrightarrow$$ `Ixfun=@(a,b,y) (a-b)*sin(2*y-1-pi/3)/2 + cos(2*a+1-pi/3)/4 - cos(2*b+1-pi/3)/4;`

$$\int_c^d u(x,y)v(x,y)dy \Longrightarrow$$ `Iyfun=@(c,d,x) (d-c)*sin(2*x+1-pi/3)/2 - cos(2*c-1-pi/3)/4 + cos(2*d-1-pi/3)/4;`

☞ voir ep04.IntegralBd01

Plan

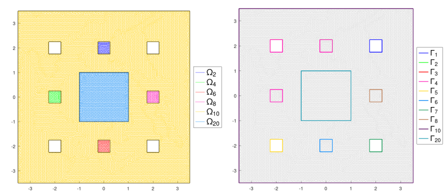


Figure: Exemple de maillage 2D: fichier `square6dom4holes.geo`

Le maillage est donné par Th (objet fc_siMesh.siMesh). L'objectif ici est d'utiliser des matrices de Masse pour approcher des intégrales de fonctions non continues sur Ω mais continues dans chaque sous-domaines.

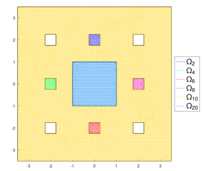
Exo: Soient $v(x, y) = \sin(x - y + 1)$ et $u : \Omega \longrightarrow \mathbb{R}$, tel que $u(\mathrm{q}) = 1$, $\forall \mathrm{q} \in \Omega_{10}$ et $u(\mathrm{q}) = 2$ sinon. (u **n'est pas continue** sur Ω!) Pour chacune des intégrales, proposer une méthode utilisant des matrices globales et une autre utilisant des matrices locales:

$$\int_{\Omega_{[2,20]}} u(\mathrm{q})v(\mathrm{q})d\mathrm{q}, \quad \int_{\Omega_{[10,20]}} u(\mathrm{q})v(\mathrm{q})d\mathrm{q}, \quad \int_{\Omega} u(\mathrm{q})v(\mathrm{q})d\mathrm{q},$$

$\Omega_{[2,20]} = \Omega_2 \cup \Omega_{20}$ $\quad \Omega = \Omega_{[2:2:10,20]}$ (notation Matlab)

Iexfun=@(a,b,c,d) −sin(−a+c−1) + sin(−a+d−1) + sin(−b+c−1) − sin(−b+d−1);

☞ voir ep04.IntegralDomDiscont



$\mathbb{M}^{L,\mathrm{lab}}$ et $\mathbb{M}^{G,\mathrm{lab}}$ resp. les matrices de Masse **locale** et **global** sur Ω^h_{lab}.
$\boldsymbol{U}^{L,\mathrm{lab}} \in \mathbb{R}^{\Omega^h_{\mathrm{lab}}.n_{\mathrm{q}}}$ et $\boldsymbol{U}^{G,\mathrm{lab}} \in \mathbb{R}^{\Omega^h.n_{\mathrm{q}}}$ vecteurs associés à la fonction u.
$\boldsymbol{V}^{L,\mathrm{lab}} \in \mathbb{R}^{\Omega^h_{\mathrm{lab}}.n_{\mathrm{q}}}$ et $\boldsymbol{V}^{G,\mathrm{lab}} \in \mathbb{R}^{\Omega^h.n_{\mathrm{q}}}$ vecteurs associés à la fonction v.

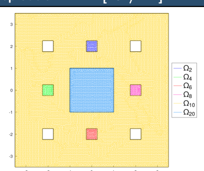
$$\int_{\Omega_{[2,20]}} u(\mathrm{q})v(\mathrm{q})d\mathrm{q} \approx \left\langle \mathbb{M}^{L,2}\boldsymbol{U}^{L,2}, \boldsymbol{V}^{L,2}\right\rangle + \left\langle \mathbb{M}^{L,20}\boldsymbol{U}^{L,20}, \boldsymbol{V}^{L,20}\right\rangle = \left\langle \mathbb{M}^{G,2}\boldsymbol{U}^{G,2}, \boldsymbol{V}^{G,2}\right\rangle + \left\langle \mathbb{M}^{G,20}\boldsymbol{U}^{G,20}, \boldsymbol{V}^{G,20}\right\rangle$$

Comme $\Omega^h_2.toGlobal \cap \Omega^h_{20}.toGlobal = \varnothing$ (i.e. $\overline{\Omega^h_2} \cap \overline{\Omega^h_{20}} = \varnothing$)

$$\left\langle \mathbb{M}^{G,2}\boldsymbol{U}^{G,2}, \boldsymbol{V}^{G,2}\right\rangle + \left\langle \mathbb{M}^{G,20}\boldsymbol{U}^{G,20}, \boldsymbol{V}^{G,20}\right\rangle = \left\langle (\mathbb{M}^{G,2} + \mathbb{M}^{G,20})(\boldsymbol{U}^{G,2} + \boldsymbol{U}^{G,20}), (\boldsymbol{V}^{G,2} + \boldsymbol{V}^{G,20})\right\rangle$$

En posant poser $\mathbb{M} = \mathbb{M}^{G,2} + \mathbb{M}^{G,20}$, $\boldsymbol{U} = \boldsymbol{U}^{G,10} + \boldsymbol{U}^{G,20}$ et $\boldsymbol{V} = \boldsymbol{V}^{G,10} + \boldsymbol{V}^{G,20}$

$$\int_{\Omega_{[2,20]}} u(\mathrm{q})v(\mathrm{q})d\mathrm{q} \approx \langle \mathbb{M}\boldsymbol{U}, \boldsymbol{V}\rangle.$$



$\mathbb{M}^{L,\mathrm{lab}}$ et $\mathbb{M}^{G,\mathrm{lab}}$ resp. les matrices de Masse **locale** et **global** sur Ω^h_{lab}.
$\boldsymbol{U}^{L,\mathrm{lab}} \in \mathbb{R}^{\Omega^h_{\mathrm{lab}}.n_{\mathrm{q}}}$ et $\boldsymbol{U}^{G,\mathrm{lab}} \in \mathbb{R}^{\Omega^h.n_{\mathrm{q}}}$ vecteurs associés à la fonction u.
$\boldsymbol{V}^{L,\mathrm{lab}} \in \mathbb{R}^{\Omega^h_{\mathrm{lab}}.n_{\mathrm{q}}}$ et $\boldsymbol{V}^{G,\mathrm{lab}} \in \mathbb{R}^{\Omega^h.n_{\mathrm{q}}}$ vecteurs associés à la fonction v.

$$\int_{\Omega_{[10,20]}} u(\mathrm{q})v(\mathrm{q})d\mathrm{q} \approx \left\langle \mathbb{M}^{L,10}\boldsymbol{U}^{L,10}, \boldsymbol{V}^{L,10}\right\rangle + \left\langle \mathbb{M}^{L,20}\boldsymbol{U}^{L,20}, \boldsymbol{V}^{L,20}\right\rangle = \left\langle \mathbb{M}^{G,10}\boldsymbol{U}^{G,10}, \boldsymbol{V}^{G,10}\right\rangle + \left\langle \mathbb{M}^{G,20}\boldsymbol{U}^{G,20}, \boldsymbol{V}^{G,20}\right\rangle$$

Comme $\Omega^h_{10}.toGlobal \cap \Omega^h_{20}.toGlobal = \mathcal{I} \neq \varnothing$ (i.e. $\overline{\Omega^h_{10}} \cap \overline{\Omega^h_{20}} \neq \varnothing$)

$$\left\langle \mathbb{M}^{G,10}\boldsymbol{U}^{G,10}, \boldsymbol{V}^{G,10}\right\rangle + \left\langle \mathbb{M}^{G,20}\boldsymbol{U}^{G,20}, \boldsymbol{V}^{G,20}\right\rangle \neq \left\langle (\mathbb{M}^{G,10} + \mathbb{M}^{G,20})(\boldsymbol{U}^{G,10} + \boldsymbol{U}^{G,20}), (\boldsymbol{V}^{G,2} + \boldsymbol{V}^{G,20})\right\rangle$$

Toutefois si u et v sont **continues** sur $\overline{\Omega_{10}} \cup \overline{\Omega_{20}}$, on a $\boldsymbol{U}^{G,10}_{\mathcal{I}} = \boldsymbol{U}^{G,20}_{\mathcal{I}}$.

Avec $\mathbb{M} = \mathbb{M}^{G,10} + \mathbb{M}^{G,20}$, et $\boldsymbol{U} \in \mathbb{R}^{\Omega^h.n_{\mathrm{q}}}$, tel que

$$\boldsymbol{U}(\Omega^h_{10}.toGlobal) = \boldsymbol{U}^{L,10} \quad \text{et} \quad \boldsymbol{U}(\Omega^h_{20}.toGlobal) = \boldsymbol{U}^{L,20}, \quad \text{(autres termes nuls)}$$

$$\int_{\Omega_{[10,20]}} u(\mathrm{q})v(\mathrm{q})d\mathrm{q} \approx \langle \mathbb{M}\boldsymbol{U}, \boldsymbol{V}\rangle.$$

Plan

Etude de la fonction fournie siMesh.Masse (fichier `Masse.m`, répertoire `+siMesh`) retournant une ou plusieurs matrices de Masse.

. ☞ voir ep04.IntegralDom02 et ep04.IntegralBd02

Exemple d'appels sur un maillage Th généré à partir du fichier `square6dom4holes.geo`:

- M=siMesh.Masse(Th);
- M=siMesh.Masse(Th,'labels',[10,20]);
- M=siMesh.Masse(Th,'labels',10,'local',true);
- [M,idx,labels]=siMesh.Masse(Th,'local',true,'labels',[20,10,4]);
- M=siMesh.Masse(Th,'d',1);
- M=siMesh.Masse(Th,'labels',[10,20],'d',1);
- M=siMesh.Masse(Th,'d',1,'labels',[10,20],'local',true);
- ...

Exo: Ecrire une fonction similaire pour la matrice de Rigidité.

La fonction fournie siMesh.eval (fichier `eval.m`, répertoire `+siMesh`) fonctionne sur le même principe (evaluation d'une fonction aux noeuds d'un maillage ou ...):

U=siMesh.eval(Th,u);

Fonctions Matlab/Octave écrites ou fournies

- siMeshElt.relevement ☞ fichier `relevement.m` dans répertoire `+siMeshElt`
- siMeshElt.restriction ☞ fichier `restriction.m` dans répertoire `+siMeshElt`
- siMeshElt.Masse ☞ fichier `Masse.m` dans répertoire `+siMeshElt`
- siMeshElt.Rigidite ☞ fichier `Rigidite.m` dans répertoire `+siMeshElt`
- siMesh.Masse ☞ fichier `Masse.m` dans répertoire `+siMesh` (fournie)
- siMesh.eval ☞ fichier `eval.m` dans répertoire `+siMesh` (fournie)
- siMesh.Rigidite ☞ fichier `Rigidite.m` dans répertoire `+siMesh`