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 3

Intégration et assemblage

François Cuvelier

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

Prérequis

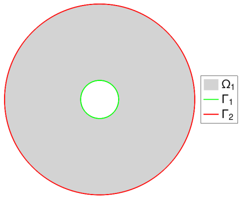
- Notion de maillages,
- Eléments finis $\mathbb{P}_1$-Lagrange en dimension 1,
- Coordonnées barycentriques sur un d-simplexe,

☞ voir vidéos dédiées

Objectif : calculer des approximations d'intégrales sur un maillage élémentaire, par ex.: $\boldsymbol{x} = (x_1, x_2)$

$$\int_{\Omega_1} u(\boldsymbol{x})v(\boldsymbol{x})d\boldsymbol{x}, \int_{\Omega_1} \langle \boldsymbol{\nabla}\, u(\boldsymbol{x}), \boldsymbol{\nabla}\, v(\boldsymbol{x})\rangle\, d\boldsymbol{x}$$

$$\int_{\Omega_1} \frac{\partial u}{\partial x}(x,y)v(x,y)dxdy, \quad \int_{\Gamma_2} u(\sigma)v(\sigma)d\sigma$$



Plan

$E \subset \mathbb{R}^n$ borné. (E est sous-variété de dimension d de $\mathbb{R}^n$)
Soient $u, v \in V(E)$ données, ($V(E) = L^2(E;\mathbb{R})$ par ex.), comment calculer

$$\mathcal{A}(u,v) = \int_E u(\boldsymbol{x})v(\boldsymbol{x})d\boldsymbol{x}?$$

ou, plus généralement,

$$\mathcal{A}(u,v) = \int_E \mathcal{D}(u,v)(\boldsymbol{x})d\boldsymbol{x}$$

avec

- $\mathcal{D}(\bullet,\bullet)$ bilinéaire
- $\operatorname{supp}(\mathcal{D}(u,v)) \subset \operatorname{supp}(u) \cap \operatorname{supp}(v)$.

Par exemples,

- $\mathcal{D}(u,v) = u.v$, ($\Rightarrow$ *matrice de Masse*)
- $\mathcal{D}(u,v) = w.u.v$, avec w une fonction poids donnée ($\Rightarrow$ *matrice de Masse avec poids*)
- $\mathcal{D}(u,v) = \frac{\partial u}{\partial x_i}.v$,
- $\mathcal{D}(u,v)(\boldsymbol{x}) = \langle \boldsymbol{\nabla}\, u(\boldsymbol{x}), \boldsymbol{\nabla}\, v(\boldsymbol{x})\rangle$ ($\Rightarrow$ *matrice de Rigidité*)

Episode 1: rappels

- E_h un maillage élémentaire de type d-simplexes de E:
  - $E_h.\mathrm{q}$, `size` $= (n, n_\mathrm{q})$; $E_h.\mathrm{me}$, `size` $= (\mathrm{d}+1, n_\mathrm{me})$; $E_h.\mathrm{toGlobal}$, `size` $= (1, n_\mathrm{q})$
    $\to$ avec $n_\mathrm{q} = E_h.n_\mathrm{q}$, $n_\mathrm{me} = E_h.n_\mathrm{me}$, $n = E_h.n$, ...
- $V(E_h)$, espace des éléments finis $\mathbb{P}_1$-Lagrange sur E_h :

$$\begin{aligned} V(E_h) &\stackrel{\text{def}}{=} \left\{ v \in \mathcal{C}^0(E_h;\mathbb{R}) \text{ tq } \forall k \in [\![1, n_\mathrm{me}]\!],\ v_{|K_k} \in \mathbb{R}_1[X_1, \dots, X_n] \right\} \\ &= \mathrm{Vect}\left\{ \varphi_1, \dots, \varphi_{n_\mathrm{q}} \right\} \end{aligned}$$

Les $\varphi_i \in V(E_h)$ vérifient
- $\varphi_i(\mathrm{q}^j) = \delta_{i,j}, \quad \forall (i,j) \in [\![1, n_\mathrm{q}]\!]^2$
- $\varphi_i \equiv 0$ sur K_k, si $q^i \notin K_k \quad \Rightarrow \quad \mathrm{supp}(\varphi_i) = \displaystyle\bigcup_{\{k \in [\![1, n_\mathrm{me}]\!];\ q^i \in K_k\}} K_k$

- Opérateur d'interpolation:

$$\left\{ \begin{array}{rcccl} \pi_h & : & V(E) & \longrightarrow & V(E_h) \\ & & w & \longmapsto & \pi_h(w) = \displaystyle\sum_{i=1}^{n_\mathrm{q}} w(q^i)\varphi_i \end{array} \right.$$

Epidodes 1 et 2: medley

Soit K un d-simplexe de $\mathbb{R}^n$ dont les $(\mathrm{d}+1)$ sommets sont les colonnes de q.

- vol=simplex.volume(q) retourne le volume d'un d-simplexe $K \subset \mathbb{R}^n$ dont les $(\mathrm{d}+1)$ sommets sont les colonnes de q
- G=simplex.gradBaCo(q) retourne les gradients des coordonnées barycentriques de K

☞ voir épidode 2

Exo: Soit E_h un maillage élémentaire de type d-simplexes de $E \subset \mathbb{R}^n$. On dispose du tableau de point $\mathrm{q} = E_h.\mathrm{q}$, (n, n_q), et du tableau de connectivité $\mathrm{me} = E_h.\mathrm{me}$, $(\mathrm{d}+1, n_\mathrm{me})$.

1. Ecrire la fonction Matlab/Octave, volumes, permettant de calculer tous les volumes des d-simplexes de E_h. Cette fonction sera enregistrée dans le répertoire +FEM.
2. Ecrire la fonction Matlab/Octave, gradBaCo, permettant de calculer tous les gradients des coordonnées barycentriques de tous les d-simplexes de E_h. On stokera le résultat dans un tableau 3D initialisé par zeros(nme,d+1,dim). Cette fonction sera enregistrée dans le répertoire +FEM.

Soient $(u, v) \in V(E)^2$. On pose $u_h = \pi_h(u) \stackrel{\text{def}}{=} \displaystyle\sum_{i=1}^{n_\mathrm{q}} u(q^i)\varphi_i$ et $v_h = \pi_h(v) \stackrel{\text{def}}{=} \displaystyle\sum_{i=1}^{n_\mathrm{q}} v(q^i)\varphi_i$ alors

$$\mathcal{A}(u, v) = \int_E \mathcal{D}(u, v)(\boldsymbol{x})d\boldsymbol{x} \ \approx\ \mathcal{A}_h(u_h, v_h) = \int_{E_h} \mathcal{D}(u_h, v_h)(\boldsymbol{x})d\boldsymbol{x}$$

Exo: Montrer que $\mathcal{A}_h(u_h, v_h) = \left\langle \mathbb{A}^h \boldsymbol{U}, \boldsymbol{V} \right\rangle$ où l'on explicitera $\mathbb{A}^h \in \mathcal{M}_{n_\mathrm{q}}(\mathbb{R})$, $\boldsymbol{U} \in \mathbb{R}^{n_\mathrm{q}}$, et $\boldsymbol{V} \in \mathbb{R}^{n_\mathrm{q}}$

- **rappel:** $\mathcal{D}(\bullet, \bullet)$ bilinéaire ...

Comme $\mathcal{A} : V(E) \times V(E) \longrightarrow \mathbb{R}$ et $\mathcal{A}_h : V(E_h) \times V(E_h) \longrightarrow \mathbb{R}$ sont **bilinéaires**:

$$\begin{aligned} \mathcal{A}_h(u_h, v_h) &= \mathcal{A}_h\left(\sum_{j=1}^{n_\mathrm{q}} u(q^j)\varphi_j, \sum_{i=1}^{n_\mathrm{q}} v(q^i)\varphi_i \right) \\ &= \sum_{i=1}^{n_\mathrm{q}} \sum_{j=1}^{n_\mathrm{q}} u(q^j)v(q^i)\mathcal{A}_h(\varphi_j, \varphi_i) \\ &= \sum_{i=1}^{n_\mathrm{q}} \left(\sum_{j=1}^{n_\mathrm{q}} \underbrace{\mathcal{A}_h(\varphi_j, \varphi_i)}_{\mathbb{A}^h_{i,j}} \underbrace{u(q^j)}_{\boldsymbol{U}_j} \right) v(q^i) \\ &= \sum_{i=1}^{n_\mathrm{q}} \left(\mathbb{A}^h \boldsymbol{U}\right)_i \underbrace{v(q^i)}_{\boldsymbol{V}_i} \\ &= \left\langle \mathbb{A}^h \boldsymbol{U}, \boldsymbol{V} \right\rangle, \quad \mathbb{A}^h \in \mathcal{M}_{n_\mathrm{q}}(\mathbb{R}),\ \boldsymbol{U} \in \mathbb{R}^{n_\mathrm{q}},\ \boldsymbol{V} \in \mathbb{R}^{n_\mathrm{q}} \end{aligned}$$

avec $\mathbb{A}^h_{i,j} = \mathcal{A}_h(\varphi_j, \varphi_i)$, $\boldsymbol{U}_j = u(\mathrm{q}^j)$ et $\boldsymbol{V}_i = u(\mathrm{q}^i)$

Exo: On note $\mathbb{M} \in \mathcal{M}_{n_q}(\mathbb{R})$ et $\mathbb{K} \in \mathcal{M}_{n_q}(\mathbb{R})$, respectivement les matrices de Masse et de Rigidité, définies par

$$\mathbb{M}_{i,j} = \int_{E_h} \varphi_j(\mathrm{q})\varphi_i(\mathrm{q})d\mathrm{q} \quad \text{et} \quad \mathbb{K}_{i,j} = \int_{E_h} \langle \nabla \varphi_j(\mathrm{q}), \nabla \varphi_i(\mathrm{q}) \rangle d\mathrm{q}.$$

Soit $\boldsymbol{W} \in \mathbb{R}^{n_q}$, $W_i = 1$, $\forall i \in [\![1, n_q]\!]$.

1. Montrer que $\langle \mathbb{M}\boldsymbol{W}, \boldsymbol{W} \rangle = |E_h|$.
2. Montrer que $\mathbb{K}\boldsymbol{W} = \boldsymbol{0}$ et $\langle \mathbb{K}\boldsymbol{U}, \boldsymbol{W} \rangle = 0$, $\forall \boldsymbol{U} \in \mathbb{R}^{n_q}$.

- On a $\mathcal{A}_h(u_h, v_h) = \langle \mathbb{A}^h \boldsymbol{U}, \boldsymbol{V} \rangle$ ...

Exo: Soit $k \in [\![1, n_{\mathrm{me}}]\!]$. Déterminer des conditions sur i et j pour que $\mathcal{D}(\varphi_j, \varphi_j) \equiv 0$ sur K_k.

- $\mathcal{D}(\bullet, \bullet)$ bilinéaire et $\mathrm{supp}\left(\mathcal{D}(u_h, v_h)\right) \subset \mathrm{supp}(u_h) \cap \mathrm{supp}(v_h)$.
- $\varphi_i \equiv 0$ sur K_k, si $q^i \notin K_k \quad \Rightarrow \quad \mathrm{supp}(\varphi_i) = \bigcup_{\{k \in [\![1, n_{\mathrm{me}}]\!];\ \mathrm{q}^i \in K_k\}} K_k$

Plan

$$\mathcal{D}(\varphi_j, \varphi_j) \equiv 0 \text{ sur } K_k \text{ si } \mathrm{q}^i \notin K_k \text{ ou } \mathrm{q}^j \notin K_k, \qquad E_h = \bigcup_{k=1}^{n_{\mathrm{me}}} K_k, \quad \mathring{K}_r \cap \mathring{K}_s = \emptyset \text{ si } r \neq s,$$

$$\mathbb{A}^h_{i,j} = \mathcal{A}_h(\varphi_j, \varphi_i) = \int_{E_h} \mathcal{D}(\varphi_j, \varphi_i)(\boldsymbol{x})d\boldsymbol{x} = \sum_{k=1}^{n_{\mathrm{me}}} \int_{K_k} \mathcal{D}(\varphi_j, \varphi_i)(\boldsymbol{x})d\boldsymbol{x}$$

Algorithme 1 Naive finite element assembly algorithm (version 1)

```
A^h ← 0
Pour i ← 1 to n_q faire
  Pour j ← 1 to n_q faire
    Pour k ← 1 to n_me faire
      A^h_{i,j} ← A^h_{i,j} + ∫_{K_k} D(φ_j, φ_i)(x) dx
    Fin Pour
  Fin Pour
Fin Pour
```

Algorithme 2 Naive finite element assembly algorithm (version 2)

```
A^h ← 0
Pour k ← 1 to n_me faire
  Pour i ← 1 to n_q faire
    Pour j ← 1 to n_q faire
      A^h_{i,j} ← A^h_{i,j} + ∫_{K_k} D(φ_j, φ_i)(x) dx
    Fin Pour
  Fin Pour
Fin Pour
```

$$\mathcal{D}(\varphi_j,\varphi_j) \equiv 0 \text{ sur } K_k \text{ si } \mathrm{q}^i \notin K_k \text{ ou } \mathrm{q}^j \notin K_k, \qquad E_h = \bigcup_{k=1}^{n_{\mathrm{me}}} K_k, \quad \mathring{K}_r \cap \mathring{K}_s = \emptyset \text{ si } r \neq s,$$

$$\mathbb{A}^h_{i,j} = \mathcal{A}_h(\varphi_j,\varphi_i) = \int_{E_h} \mathcal{D}(\varphi_j,\varphi_i)(\boldsymbol{x})d\boldsymbol{x} = \sum_{k=1}^{n_{\mathrm{me}}} \int_{K_k} \mathcal{D}(\varphi_j,\varphi_i)(\boldsymbol{x})d\boldsymbol{x}$$

Algorithme 2 Naive finite element assembly algorithm (version 2)

$\mathbb{A}^h \leftarrow 0$
Pour $k \leftarrow 1$ to n_{me} **faire**
  Pour $i \leftarrow 1$ to n_{q} **faire**
    Pour $j \leftarrow 1$ to n_{q} **faire**
      $\mathbb{A}^h_{i,j} \leftarrow \mathbb{A}^h_{i,j} + \int_{K_k} \mathcal{D}(\varphi_j,\varphi_i)(\boldsymbol{x})d\boldsymbol{x}$
    Fin Pour
  Fin Pour
Fin Pour

Algorithme 3 classic finite element assembly algorithm

$\mathbb{A}^h \leftarrow 0$
Pour $k \leftarrow 1$ to n_{me} **faire**
  Pour $\alpha \leftarrow 1$ to $\mathrm{d}+1$ **faire**
    $i \leftarrow \mathrm{me}(\alpha,k)$
    Pour $\beta \leftarrow 1$ to $\mathrm{d}+1$ **faire**
      $j \leftarrow \mathrm{me}(\beta,k)$
      $\mathbb{A}^h_{i,j} \leftarrow \mathbb{A}^h_{i,j} + \int_{K_k} \mathcal{D}(\varphi_j,\varphi_i)(\boldsymbol{x})d\boldsymbol{x}$
    Fin Pour
  Fin Pour
Fin Pour

Algorithme 3 classic finite element assembly algorithm

$\mathbb{A}^h \leftarrow 0$
Pour $k \leftarrow 1$ to n_{me} **faire**
  Pour $\alpha \leftarrow 1$ to $\mathrm{d}+1$ **faire**
    $i \leftarrow \mathrm{me}(\alpha,k)$
    Pour $\beta \leftarrow 1$ to $\mathrm{d}+1$ **faire**
      $j \leftarrow \mathrm{me}(\beta,k)$
      $\mathbb{A}^h_{i,j} \leftarrow \mathbb{A}^h_{i,j} + \int_{K_k} \mathcal{D}(\varphi_j,\varphi_i)(\boldsymbol{x})d\boldsymbol{x}$
    Fin Pour
  Fin Pour
Fin Pour

Algorithme 4 classic finite element assembly algorithm with elementary matrices

$\mathbb{A}^h \leftarrow 0$
Pour $k \leftarrow 1$ to n_{me} **faire**
  $\mathbb{A}^e(K_k) \leftarrow \textsc{ElemMat}(\ldots)$ // $\mathbb{A}^e(K_k) \in \mathcal{M}_{\mathrm{d}+1}(\mathbb{R})$
  Pour $\alpha \leftarrow 1$ to $d+1$ **faire**
    $i \leftarrow \mathrm{me}(\alpha,k)$
    Pour $\beta \leftarrow 1$ to $d+1$ **faire**
      $j \leftarrow \mathrm{me}(\beta,k)$
      $\mathbb{A}^h_{i,j} \leftarrow \mathbb{A}^h_{i,j} + \mathbb{A}^e_{\alpha,\beta}(K_k)$
    Fin Pour
  Fin Pour
Fin Pour

$$\mathbb{A}^e_{\alpha,\beta}(K_k) = \int_{K_k} \mathcal{D}(\varphi_j,\varphi_i)(\boldsymbol{x})d\boldsymbol{x},$$

avec $i = \mathrm{me}(\alpha,k)$ et $j = \mathrm{me}(\beta,k)$.

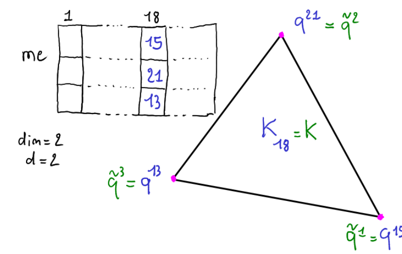
Plan

k fixé. Les $(\mathrm{d}+1)$ sommets de $K = K_k$ sont

$$\tilde{\mathrm{q}}^\alpha = \mathrm{q}(:,\mathrm{me}(\alpha,k)), \ \forall \alpha \in [\![1,\mathrm{d}+1]\!]$$

Les α correspondent à une **numérotation locale des sommets** du d-simplexe K.
Soit $\alpha \in [\![1,d+1]\!]$, $i = \mathrm{me}(\alpha,k)$, on note

$$\tilde{\varphi}_\alpha = \varphi_{i|K_k}.$$



- $\tilde{\varphi}_\alpha : K \longrightarrow \mathbb{R}$ un polynôme de degré 1
- $\tilde{\varphi}_\alpha(\tilde{\mathrm{q}}^\beta) = \delta_{\alpha,\beta}, \ \ \forall \beta \in [\![1,\mathrm{d}+1]\!]$.

Ce sont donc les **coordonnées barycentriques du d-simplexe K_k** :

$$\tilde{\varphi}_\alpha = \lambda_{\alpha-1}, \ \ \forall \alpha \in [\![1,\mathrm{d}+1]\!].$$

$$\mathbb{A}^e(K) \in \mathcal{M}_{\mathrm{d}+1}(\mathbb{R}), \quad \mathbb{A}^e_{\alpha,\beta}(K) = \int_K \mathcal{D}(\tilde{\varphi}_\beta,\tilde{\varphi}_\alpha)(\boldsymbol{x})d\boldsymbol{x}, \ \forall (\alpha,\beta) \in [\![1,\mathrm{d}+1]\!]^2,$$

$\mathbb{A}^e(K)$ est donc la matrice élémentaire de $\mathcal{D}$ sur K ☞ voir épisode 2!

Plan

Algorithme 4 classic finite element assembly algorithm with elementary matrices

$\mathbb{A}^h \leftarrow 0$
Pour $k \leftarrow 1$ to n_{me} **faire**
  $\mathbb{A}^e(K_k) \leftarrow \textsc{ElemMat}(\ldots)$ $\quad$ // $\mathbb{A}^e(K_k) \in \mathcal{M}_{d+1}(\mathbb{R})$
  Pour $\alpha \leftarrow 1$ to $\mathrm{d}+1$ **faire**
    $i \leftarrow \mathrm{me}(\alpha, k)$
    Pour $\beta \leftarrow 1$ to $\mathrm{d}+1$ **faire**
      $j \leftarrow \mathrm{me}(\beta, k)$
      $\mathbb{A}^h_{i,j} \leftarrow \mathbb{A}^h_{i,j} + \mathbb{A}^e_{\alpha,\beta}(K_k)$
    Fin Pour
  Fin Pour
Fin Pour

On dispose (voir épidode 2) des fonctions

Ae=simplex.MasseElem(d,vol)

et

Ae=simplex.RigiditeElem(vol,G)

permettant de calculer respectivement les matrices élémentaires de Masse et de Rigidité sur un d-simplexe.

Exo:

1. Ecrire les fonctions d'assemblage Masse et Rigidite permettant de calculer respectivement les matrices de Masse et de Rigidité associées à un maillage élémentaire E_h. Ces matrices sont creuses! Ces fonctions seront enregistrées dans le répertoire `+FEM`.
2. Proposer une première approche simple pour tester ces deux fonctions. ☞ voir ep03.test01

Fonctions Matlab/Octave écrites

- FEM.volumes ☞ fichier `volumes.m` dans répertoire `+FEM`
- FEM.gradBaCo ☞ fichier `gradBaCo.m` dans répertoire `+FEM`
- FEM.Masse ☞ fichier `Masse.m` dans répertoire `+FEM`
- FEM.Rigidite ☞ fichier `Rigidite.m` dans répertoire `+FEM`