

# Quelques améliorations 6

---

## Objectifs de la séance :

- Utiliser des maillages créés par **Gmsh**.
- Utiliser le format de matrice creuse de **SciPy**.

Dans cette séance, nous voulons améliorer la structure des matrices utilisées jusque là. Dans le cas des éléments finis, remarquez que la matrice contient un grand nombre de coefficients nuls, de telle sorte qu'il existe des structures de matrice plus adaptées qui améliorent notamment la complexité de produit matrice vecteur. Cela nous permettra par exemple d'inverser la matrice plus rapidement en utilisant des méthodes itératives. Ces structures de matrices permettent également d'alléger l'utilisation de la mémoire lors du stockage de la matrice.

Par ailleurs, nous voulons aussi pouvoir résoudre des problèmes sur des géométries plus complexes que celle utilisées jusque là, et avec des maillages moins réguliers (angle droit, segment parallèle, similitude dans les triangles...). Mais dans ce cas, nous ne pouvons pas mailler à la main comme nous l'avons fait pour un rectangle, il nous faut utiliser un mailleur externe comme **Gmsh**. Nous devons alors récupérer les nœuds sur les bords pour pouvoir appliquer les conditions de Dirichlet.

## 1 Création d'un maillage avec Gmsh

Nous allons maintenant mailler quelques géométries simples à l'aide de **Gmsh**. Tout d'abord, il faut créer un fichier `.geo` qui va contenir la description de la géométrie. Puis, il faut appeler **Gmsh** en lui donnant ce fichier en argument pour qu'il maille la géométrie. Le maillage résultant est alors contenu dans un fichier `.msh` du même nom. Son format similaire est à celui que nous avons utilisé jusque là. Voici un exemple de fichier `.geo` pour un carré de rayon  $R$  et de finesse  $lc$  :

```
1 R=1;
2 lc = 0.1;
3 Point(1) = { 0 , 0 , 0 , lc};
4 Point(2) = { R , 0 , 0 , lc};
5 Point(3) = { R , R , 0 , lc};
6 Point(4) = { 0 , R , 0 , lc};
7
8 Line(1) = {1,2} ;
9 Line(2) = {2,3} ;
10 Line(3) = {3,4} ;
11 Line(4) = {4,1} ;
12
13 Line Loop(0) ={ 1 , 2 , 3 , 4};
14 Plane Surface(0) = {0};
15 Physical Line(0) ={ 0 };
```

et pour un disque de rayon  $R$  et de finesse  $lc$  :

```

1 R=1;
2 lc = 0.1;
3 Point(0) = { 0 , 0 , 0 , lc};
4 Point(1) = { R , 0 , 0 , lc};
5 Point(2) = { 0 , R , 0 , lc};
6 Point(3) = { -R , 0 , 0 , lc};
7 Point(4) = { 0 , -R , 0 , lc};
8
9 Circle(1) = {1 , 0 , 2 };
10 Circle(2) = {2 , 0 , 3 };
11 Circle(3) = {3 , 0 , 4 };
12 Circle(4) = {4 , 0 , 1 };
13
14 Line Loop(0) ={ 1 , 2 , 3 , 4};
15 Plane Surface(0) = {0};
16 Physical Line(0) ={ 0 };

```

**Exercice 1.**

1. Implémentez une fonction qui crée un fichier `.geo` pour un carré et qui prend en argument deux réels, la taille du carré et la finesse du maillage, ainsi qu'une chaîne de caractère, le nom du fichier. Faites de même pour un disque.
2. Créez une fonction qui prend en argument un nom de fichier `.geo` et qui appelle `Gmsh` pour mailler la géométrie du fichier en argument. (cf. module `subprocess`).
3. Adaptez la fonction de lecture de maillage au format de `Gmsh`. Attention aux tags pour les éléments (cf. section 9.1 de la documentation de `Gmsh`). On prendra soin à aussi extraire du fichier de maillage le tableau des arêtes sur le bord.
4. Vérifiez la lecture du maillage avec vos fonctions de visualisation.

**2 Utilisation des matrices creuses**

Nous allons maintenant utiliser les outils mis en place précédemment pour reprendre l'équation suivante :

$$-\Delta u + u = f \text{ dans } \Omega \quad (6.1)$$

où  $f \in L^2(\Omega)$  et  $\Omega \subset \mathbb{R}^2$ . On pourra considérer un domaine carré ou cylindrique.

Pour cela, vous pouvez utiliser le module de `SciPy` qui gère divers formats de matrices creuses tels que Compressed Sparse Row (CSR), Compressed Sparse Column et Row-based Linked List (LIL) qui présentent tout les 3 des avantages et des inconvénients précis, que vous devez choisir en fonction de l'utilisation que vous avez de la matrice.

**Exercice 2** (Condition de Neumann homogène). Nous allons résoudre (6.1) avec une condition de Neumann homogène.

1. Résolvez le problème considéré en utilisant des matrices creuses.
2. Tracez des graphes du temps d'exécution de la résolution du problème, par rapport à la taille de la matrice, avec et sans matrices creuses, en utilisant une méthode directe et une méthode itérative. On fera attention à se placer en échelle logarithmique.

**Exercice 3** (Condition de Dirichlet homogène). Faites la même chose qu'à l'exercice précédent pour une condition de Dirichlet homogène.

**3 Analyse numérique de l'erreur d'approximation de la méthode**

On avait vu en exercice dans les TP4 et 5, la théorie pour obtenir une estimation d'erreur d'ordre 1 en semi-norme  $H^1 : |\cdot|_{H^1}$ . Cependant, vous avez pu remarquer que l'on obtient un ordre 2 numériquement pour la résolution de notre problème. Nous allons tenter de mettre en lumière ce qu'il se passe.

**Exercice 4** (Analyse d'erreur numérique). Pour chacun des problèmes considérés en TP4 et 5, càd 6.1 & CN et 6.1 & CD :

1. Tracer  $\|u - u_h\|_{L^2(\Omega)}$  et  $|u - u_h|_{H^1(\Omega)}$  en échelle logarithmique pour différentes valeurs du paramètre  $h$ , en exploitant les structures creuses des matrices et pour des maillages générés avec GMSH. Pour les conditions de Dirichlet, on essaiera avec la méthode de pénalisation exacte et la méthode de pseudo-élimination.
2. Comparez alors les ordres numériques obtenues avec les maillages de GMSH et le maillage obtenu grâce à la fonction du TP1.
3. Essayez d'expliquer ce qui se passe.

Un début de réponse : <https://en.wikipedia.org/wiki/Superconvergence>

## 4 Transformation affine et élément de référence

Lorsque l'on veut coder une librairie élément fini, on veut généralement que celle-ci soit capable de faire plus que des éléments finis  $\mathbb{P}^1$ -Lagrange. Or nous avons jusqu'à là, pour calculer les contributions locales de la forme bilinéaire du problème, utiliser des formules qui ne sont valides que pour des éléments finis  $\mathbb{P}^1$ -Lagrange. La stratégie classique, pour faire des éléments finis d'ordre plus élevé, disons d'ordre  $\mathbb{P}^k$ , c'est de ramener les calculs à un élément de référence puis d'utiliser des formules de quadrature, qui calculent les contributions de cet élément de référence de la forme bilinéaire issue du problème étudié. Ces contributions sont alors modifiées pour devenir les contributions locales d'un triangle quelconque du maillage par le biais d'applications affines.

**Exercice 5** (Transformation affine). Soit  $\Omega$  un domaine maillé par la triangulation  $\mathcal{T}_h$ . Soit  $\hat{T}$  le triangle de référence, composé des sommets  $\hat{s}_1 = (0, 0)$ ,  $\hat{s}_2 = (1, 0)$  et  $\hat{s}_3 = (0, 1)$ . Soit  $T$  un triangle quelconque de  $\mathcal{T}_h$ , de sommet  $s_1, s_2$  et  $s_3$ .

1. Déterminer l'expression de l'unique transformation affine  $F_T : \mathbb{R}^2 \rightarrow \mathbb{R}^2$  qui envoie le triangle de référence  $\hat{T}$  sur le triangle  $T$ , qui préserve l'ordre de numérotation locale des sommets.
2. Calculer la Jacobienne de cette transformation.
3. Implémenter une fonction qui prends en argument un triangle du maillage, un point de  $\mathbb{R}^2$ , et qui renvoie l'image de ce point par la fonction affine  $F_T$ . Implémenter également son inverse.

En notant,  $\hat{\varphi}_i$  les fonctions de formes sur le triangle de référence, on peut définir les fonctions de formes sur n'importe quel triangle par la transformation affine de la manière suivante :

$$\begin{cases} \varphi_i(x) = \hat{\varphi}_i(\hat{x}) \\ x = F_T(\hat{x}) \end{cases}$$

**Exercice 6** (Problème modèle). On va maintenant utiliser ceci dans le cadre de notre problème modèle 6.1. Soit  $T \in \mathcal{T}_h$  un triangle quelconque du maillage.

1. Calculer les contributions locales  $\int_T \varphi_i(x) \varphi_j(x)$  de la masse élémentaire en se ramenant à des calculs sur le triangle de référence.
2. Faire de même pour les contributions locales de la rigidité élémentaire :  $\int_T \nabla \varphi_i(x) \nabla \varphi_j(x)$ .

Une formule de quadrature à  $l$  points est la donnée d'une suite de points de l'espace  $(a_k)_{k=1 \dots l} \in \mathbb{R}^{n_N}$  et poids  $(w_k)_{k=1 \dots l} \in \mathbb{R}^N$  qui permet de calculer une quantité  $\int_D f(x)$  de la manière suivante :

$$\left| \int_D f(x) - \sum_{k=1}^l w_k f(a_k) \right| < \varepsilon$$

avec  $\varepsilon$  qui dépend de la régularité de  $f$ , de l'aire du domaine  $D$  et de la quadrature utilisée, mais qui est naturellement typiquement petit. On donne en annexe, un tableau avec des quadratures pour le triangle de référence choisi.

**Exercice 7** (Formule de quadrature).

1. Choisir une formule de quadrature qui convient pour calculer les contributions locales vu à l'exercice précédent du problème modèle 6.1 pour des éléments finis  $\mathbb{P}^1$ -Lagrange.
2. Ecrire les formules permettant alors de calculer les contributions locales à un triangle :  $\int_T \varphi_i(x) \varphi_j(x)$  et  $\int_T \nabla \varphi_i(x) \nabla \varphi_j(x)$  comme somme sur les points de quadratures.

**Exercice 8** (Implémentation).

1. Donner l'expression des fonctions de formes associées au triangle de référence, ainsi que l'expression de leur gradient.
2. Implémenter alors une fonction qui prend un point de  $\mathbb{R}^2$  et un indice  $i = 1 \dots 3$  et qui renvoie la valeur de la fonction de forme associée au nœud  $i$  dans la numérotation local du triangle de référence. Faites de même pour le gradient des fonctions de formes associées au triangle de référence.
3. Modifier alors votre code pour que le calcul des contributions locales à chaque triangle soit fait avec la transformation affine et des formules de quadratures.
4. Vérifiez que vous obtenez bien la même chose qu'avant en testant avec les différentes conditions de bord que l'on a vu jusque là.

## Annexe

Quadature formula on a triangle					
$L$	qft=	qforder=	point in $T_k$	$\omega_\ell$	exact on $P_k, k =$
1	qf1pT	2	$\left(\frac{1}{3}, \frac{1}{3}\right)$	$ T_k $	1
3	qf2pT	3	$\left(\frac{1}{2}, \frac{1}{2}\right)$ $\left(\frac{1}{2}, 0\right)$ $\left(0, \frac{1}{2}\right)$	$ T_k /3$ $ T_k /3$ $ T_k /3$	2
7	<b>qf5pT</b>	6	$\left(\frac{1}{3}, \frac{1}{3}\right)$ $\left(\frac{6-\sqrt{15}}{21}, \frac{6-\sqrt{15}}{21}\right)$ $\left(\frac{6-\sqrt{15}}{21}, \frac{9+2\sqrt{15}}{21}\right)$ $\left(\frac{9+2\sqrt{15}}{21}, \frac{6-\sqrt{15}}{21}\right)$ $\left(\frac{6+\sqrt{15}}{21}, \frac{6+\sqrt{15}}{21}\right)$ $\left(\frac{6+\sqrt{15}}{21}, \frac{9-2\sqrt{15}}{21}\right)$ $\left(\frac{9-2\sqrt{15}}{21}, \frac{6+\sqrt{15}}{21}\right)$	$0.225 T_k $ $\frac{(155-\sqrt{15}) T_k }{1200}$ $\frac{(155-\sqrt{15}) T_k }{1200}$ $\frac{(155-\sqrt{15}) T_k }{1200}$ $\frac{(155+\sqrt{15}) T_k }{1200}$ $\frac{(155+\sqrt{15}) T_k }{1200}$ $\frac{(155+\sqrt{15}) T_k }{1200}$	5
3	qf1pTlump		$(0, 0)$ $(1, 0)$ $(0, 1)$	$ T_k /3$ $ T_k /3$ $ T_k /3$	1
9	qf2pT4P1		$\left(\frac{1}{4}, \frac{3}{4}\right)$ $\left(\frac{3}{4}, \frac{1}{4}\right)$ $\left(0, \frac{1}{4}\right)$ $\left(0, \frac{3}{4}\right)$ $\left(\frac{1}{4}, 0\right)$ $\left(\frac{3}{4}, 0\right)$ $\left(\frac{1}{4}, \frac{1}{4}\right)$ $\left(\frac{1}{4}, \frac{1}{2}\right)$ $\left(\frac{1}{2}, \frac{1}{4}\right)$	$ T_k /12$ $ T_k /12$ $ T_k /12$ $ T_k /12$ $ T_k /12$ $ T_k /12$ $ T_k /6$ $ T_k /6$ $ T_k /6$	1

FIGURE 6.1 –