

Gestion des maillages

1

Objectifs de la séance :

- Comprendre la structure des fichiers de maillage
- Lire et écrire des fichiers de maillage
- Pouvoir visualiser des fichiers de maillage

1 Fichier de maillage

Il existe de nombreuses manières de stocker les différentes données relatives à un maillage. Dans cette séance et celles qui suivront, nous prendrons comme convention le format suivant :

```
1 $Noeuds
2 nbr-de-noeuds
3 numero-noeud x-coord y-coord z-coord
4 ...
5 $FinNoeuds
6 $Elements
7 nbr-de-elements
8 numero-elements numero-noeuds-1 numero-noeuds-2 numero-noeuds-3
9 ...
10 $FinElements
```

Ce format nous permettra de stocker des maillages formés d'éléments triangulaires. Celui-ci est fortement inspiré du format utilisé par *Gmsh*¹ qui est un générateur de maillage gratuit. Ce dernier permet notamment de construire des géométries avec une interface graphique et comporte un certain nombre d'outils de visualisation.

Dans la suite, les exercices vous feront lire et écrire des fichiers de maillage. Il vous est demandé d'utiliser le package *NumPy* pour les tableaux et *matplotlib* pour la visualisation. Voici quelques liens utiles :

- NumPy
 - <https://docs.scipy.org/doc/numpy-1.15.0/user/quickstart.html>
 - <http://math.mad.free.fr/depot/numpy/essai.html>
- matplotlib
 - http://matplotlib.org/users/pyplot_tutorial.html
 - <http://matplotlib.free.fr/index.html>
- De façon plus générale :
 - <http://cs231n.github.io/python-numpy-tutorial/>

2 écriture et lecture de fichiers de maillage

Exercice 1 (Lecture d'un fichier de maillage).

1. <http://gmsh.info/>

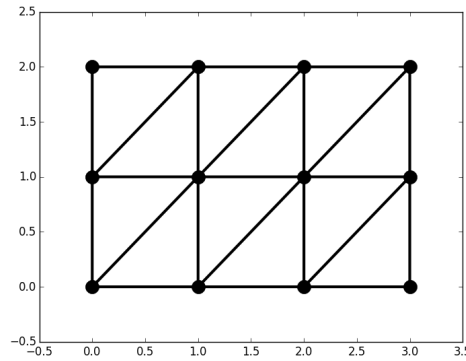


FIGURE 1.1 – Exemple de maillage pour un domaine rectangulaire

- Définissez une fonction qui prend en argument le nom d'un fichier et qui affiche dans le terminal toutes les lignes de ce fichier.
- Définissez une fonction qui prend en argument le nom d'un fichier de maillage du type de celui décrit précédemment et qui renvoie les nœuds du maillage sous forme d'un tableau. Dans notre cas, un nœud est un tableau de trois réels. (on pourra utiliser la fonction `split` par exemple)
- Définissez une autre fonction qui prend en argument le nom d'un fichier de maillage et qui renvoie les éléments d'un maillage sous forme d'un tableau. Dans notre cas, un élément est un tableau de trois entiers correspondant aux numéros des nœuds dans le tableau que renvoie la fonction précédente (remarquez que la numérotation des tableaux commence à zéro).

Un fichier de maillage sera fourni pour que vous puissiez tester les fonctions définies dans l'Exercice 1. Et il faudra les tester ! N'oubliez pas qu'il faut toujours, autant que possible, tester son code. Pour l'instant, nous pouvons nous contenter d'afficher les tableaux lus dans le terminal par exemple.

Exercice 2 (Génération d'un type de maillage).

- Définissez une fonction qui crée un fichier de maillage associé à un domaine rectangulaire. Plus précisément, la fonction doit prendre en argument :
 - le nom du fichier
 - le nombre de nœuds selon l'horizontal
 - le nombre de nœuds selon la verticale
 - la longueur horizontale de domaine
 - la longueur verticale de domaine

Le maillage créé par cette fonction est constitué de rectangles coupés en deux selon leur diagonale. Un exemple est donné Figure 1.1.

De même, on pourra tester notre fonction générant des maillages avec les fonctions de lecture des nœuds et des triangles.

3 Visualisation

Exercice 3 (Visualisation d'un maillage).

- Définissez une fonction qui crée une image représentant le maillage (voir `triplot` dans `matplotlib`). Cette fonction doit permettre d'obtenir une image du type de celle dans la Figure 1.1. N'hésitez pas à regarder les différentes options des fonctions de visualisation fournies par `matplotlib` (gestion des tailles des lignes, des markers, des couleurs ...).

Exercice 4 (Visualisation d'une fonction sur le maillage).

- Soit un tableau de valeurs associées à une fonction interpolée sur les nœuds du maillage. Définissez une fonction qui crée une image à partir de ce tableau de valeurs pour représenter

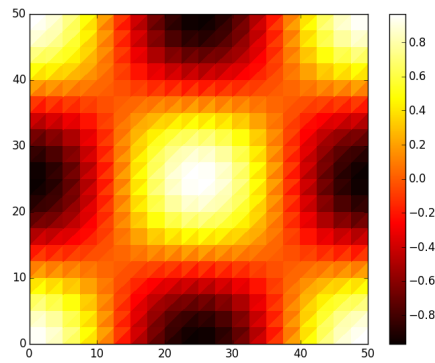


FIGURE 1.2 – Exemple de visualisation avec la fonction $\cos(\frac{(x-25)\pi}{25}) \cos(\frac{(y-25)\pi}{25})$ associée aux positions des nœuds du maillage

la fonction sur le maillage (cf. Figure 1.2 par exemple). Remarquez que le module NumPy définit des fonctions qui s'appliquent directement sur des tableaux et pensez à regarder la fonction `tricontour` et ses options dans matplotlib. Je vous laisse le choix des fonctions à représenter, faites une jolie image !