TP 1 :
Initiation à FreeFem++

Exercice 1

Introduction. La syntaxe de FreeFem++ ressemble beaucoup à celle du C ! Toutes vos variables doivent donc avoir un type. Pour les nombres, il y a essentiellement deux types différents : les `int` et les `real` qui correspondent respectivement à des entiers (*int* en C) et à des réels (*double* en C).

Dans un premier temps, le but de résoudre un problème de Poisson,

$$\left\{\begin{array}{rcll} -\Delta u & = & f & \text{dans } \Omega \setminus \bar{\omega}, \\ u & = & 0 & \text{sur } \partial\Omega, \\ u & = & 0 & \text{sur } \partial\omega, \end{array}\right.$$

où Ω est le carré unité, ω est un disque dans Ω et f est une fonction dans $L^2(\Omega \setminus \bar{\omega})$. Il y a plusieurs étapes :

1. création du maillage,
2. création des espaces éléments finis,
3. écriture de la formulation variationelle,
4. résolution du problème.

Création du maillage. Pour mailler un domaine borné, il suffit de définir une équation paramétrique pour chacun de ses cotés. Il y a donc deux étapes :

- ⋆ création des bords pour définir le domaine à mailler,
- ⋆ création du maillage avec les bords.

1. On suppose que le bord est défini par une fonction paramétrique $(X(t), Y(t))$ avec t variant

de t_1 à t_2. La syntaxe pour créer un tel bord est la suivante :

$$\texttt{border B}(\texttt{t} = \texttt{t}_1, \texttt{t}_2)\{\texttt{x} = \texttt{X}(\texttt{t}); \texttt{y} = \texttt{Y}(\texttt{t}); \texttt{label} = \texttt{n}; \};$$

où `B` est le nom de la variable qui représente le bord (peut être n'importe quoi) et `n` est un entier qui sera utilisé pour identifier le bord. Deux bords avec le même label seront vu comme un seul bord.

$\longrightarrow$ Créer un carré en construisant 4 bords appelés `B1`, `B2`, `B3` et `B4`.

2. On veut maintenant mailler l'intérieur du carré. Pour cela, on défini un maillage `Th` :

$$\texttt{mesh Th} = \texttt{buildmesh}(\texttt{B1}(\texttt{np}) + \texttt{B2}(\texttt{np}) + \texttt{B3}(\texttt{np}) + \texttt{B4}(\texttt{np}));$$

Cette commande construit un maillage à partir des 4 bords (il faut donc que les bords forment un domaine bornés) avec `np` points sur chaque bord.

$\longrightarrow$ Construire le maillage d'un carré et le représenté avec la commande `plot` :

$$\texttt{plot}(\texttt{Th}, \text{wait} = \texttt{1});$$

3. Essayer maintenant de créer un cercle C.

4. Pour faire le maillage d'un carré troué, il suffit de combiner le carré et le cercle dans le buildmesh en mettant un nombre négatif pour le cercle :

$$\texttt{buildmesh}(\texttt{B1}(\texttt{np}) + \texttt{B2}(\texttt{np}) + \texttt{B3}(\texttt{np}) + \texttt{B4}(\texttt{np}) + \texttt{C}(-\texttt{np}));$$

$\longrightarrow$ Essayer de faire le maillage d'un carré contenant un trou. Que se passe t-il si on met un nombre positif pour C à la place d'un nombre négatif ?

Choix des espaces d'éléments finis. Une fois que le maillage est créé, il faut dire à FreeFem quels sont vos espaces d'éléments finis. Il en existe au moins 3 : $P0$, $P1$ et $P2$. Pour notre problème, on va avoir besoin de fonctions $P1$ pour la solution et $P0$ pour le second membre f.

1. La commande

$$\texttt{fespace V1}(\texttt{Th}, \texttt{P1});$$

définit un espace appelé `V1` (c'est juste un nom de variable) qui contient les élément $P1$ sur le maillage `Th`. Faire la même chose pour l'espace $P0$ (on l'appellera `V0`).

2. On peut maintenant définir nos fonctions. Par exemple pour définir la solution u et la

fonction test v on pourra écrire

$$\texttt{V1 u}, \texttt{v};$$

On définit de même le terme source f, par exemple la fonction constante égale à 1 :

$$\texttt{V0 f} = \texttt{1};$$

Formulation variationnelle.

1. Écrire la formulation variationelle du problème.
2. Pour définir un problème dans FreeFem on utilise le mot clé problem

$$\texttt{problem Poisson}(\texttt{u}, \texttt{v}) = \ldots$$

Ici on appelle notre problème "Poisson" et on dit que `u` est la solution et `v` la fonction test. Il faut ensuite écrire la formulation variationelle. Plusieurs commandes pour cela :

- ⋆ `int2d`(`Th`)(expr) : permet de calculer une intégrale en 2d de l'expression expr.
- ⋆ `dx`(func) et `dy`(func) permettent d'utiliser les dérivés de la fonction func.
- ⋆ `on`(`label1`, `label2`, . . . , func=g) permet d'imposer la condition de Dirichlet g sur les bords numéros `label1`, `label2`, etc.

$\longrightarrow$ Écrire la formulation variationelle du problème dans FreeFem.

Résolution du problème et visualisation.

1. Pour résoudre le problème, il suffit d'écrire le nom du problème. Ici on écrira donc :

$$\texttt{Poisson};$$

2. On utilise la fonction plot pour visualiser la solution. Essayer les différentes options (`fill=1, value=1, dim=3`, etc.).

Exercice 2 *Vitesse de convergence sur un carré.*
On considère le problème de Poisson

$$\begin{cases} -\triangle u = f & \text{dans } \Omega = [0,1]^2 \\ u = 0 & \text{sur } \partial\Omega, \end{cases}$$

Afin de retrouver les ordres de convergences obtenus en cours, nous allons utilisé la "méthode de la solution manufacturée". Pour cela, nous nous donnons une solution du problème, par exemple

$$u_{p,q}(x,y) = \sin(p\pi x)\sin(q\pi y), \quad (p,q) \in (\mathbb{N}^*)^2.$$

On puet vérifier que toutes ces fonctions sont effectivement nulles sur le bord $\partial\Omega$. En calculant maintenant $-\triangle u_{p,q}$, nous obtenons le second membre

$$f_{p,q}(x,y) = \left[(p\pi)^2 + (q\pi)^2\right]\sin(p\pi x)\sin(q\pi y).$$

L'idée de méthode de la solution manufacturée est donc d'introduire ce second membre $f_{n,k}$ et de comparer la solution numérique $u^h_{n,k}$ avec la solution exacte $u_{n,k}$ décrite ci-dessus.

1. Écrire la formulation variationnelle correspondante à ce problème.
2. Dans le cas de la méthode des éléments finis P1 Lagrange, nous savons que l'erreur en norme en norme H^1 est d'ordre 1 et en norme L^2 est d'ordre 2. Nous allons retrouver ces ordres. Pour cela, écrivons

   $$E_{H^1}(h) = \|u^h_{p,q} - u_{p,q}\|_{H^1(\Omega)} = Ch^\alpha,$$

   avec h est la taille du maillage. En échelles logarithmiques, nous obtenons alors l'égalité

   $$\ln E_{H^1}(h) = \ln C + \alpha \ln h,$$

   et donc la courbe $\ln E_{H^1}(h)$ en fonction de $\ln h$ est une droite de pente α. Appliquons cela à la méthode des éléments finis P1 pour ce problème en chsoississant $n = 2^k$ points sur chacun des bords de Ω, avec $k \in [\![3,7]\!]$,

   - résoudre le problème
   - calculer l'erreur commise en norme $\mathbb{L}^2$ puis en norme H^1.
   - afficher les erreurs en fonction de h, en échelles logarithmiques. On pourra utiliser la commande suivante pour calculer le pas du maillage :

     ```
     fespace Ph(Th, P0);
     Ph h = hTriangle;
     real hmax = h[].max ;
     ```

   Quel est l'ordre de convergence de la méthode suivant la norme utilisée ?

Exercice 3 *Vitesse de convergence sur un domaine non convexe.*
On considère maintenant le problème

$$\begin{cases} -\triangle u = 1 & \text{dans } \Omega, \\ u = 0 & \text{sur } \partial\Omega, \end{cases}$$

où Ω est un domaine en forme de ⅃ (voir Figure 1).

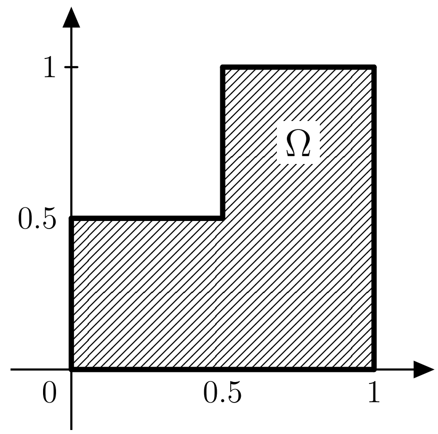

FIGURE 1 – Domaine Ω.

1. Ne connaissant pas la solution exacte de ce problème, quelle solution de référence peut-on prendre pour le calcul des erreurs $\mathbb{L}^2$ et H^1 ?
2. Reprendre l'étude numérique de la convergence dans le cas d'un domaine en ⅃. Qu'observe-t-on ?

Quelques commandes utiles...

- Vecteur de réels : `real[int] Nom(`n`);`, où n désigne la taille du vecteur.
- Boucle "for" :
```
int k;
int n0; // n0 indice de début,
int n1; // n1 indice de fin,
for (k=n0;k<n1+1;k++){
...
};
```
- Créer (ou écraser existant) et écrire dans une fichier :
```
{
ofstream file("Nom.txt");
```

```
file « "Indice" « " " « k «endl;
};
}
```

Entre guillemets, la chaîne de caractère "Indice" est écrite dans le fichier, sans les guillemets, la valeur de `k` est écrite dans le fichier ; la commande `endl` permet de changer de ligne. Il est possible d'insérer le `file` dans une boucle pour écrire plusieurs données (comme par exemple les valeurs d'un vecteur).

⋆ Liste non exhaustive d'options de `plot` :

`wait=1` // demande une action pour passer à la figure suivante,
`ps="Nom.eps"` // enregistre la figure au format .eps,
`value=1` // affiche les valeurs en légende,
`fill=0` // n'affiche que les courbes d'isovaleur,
`fill=1` // colore en plein la figure,
`dim=3` // affiche la figure en 3 dimensions.

⋆ Pour plus de commandes,

https://doc.freefem.org/references/