

INTRODUCTION - OPTIMISATION DE FORMES ET TOPOLOGIQUE

BENIAMIN BOGOSEL

CONTENTS

1. Introduction <code>FreeFem++</code>	1
1.1. Maillage et espace éléments finis	2
1.2. Équation de Laplace	2
1.3. L'élasticité linéarisée	3
1.4. Problèmes aux valeurs propres	5

L'objectif des travaux pratiques est de se familiariser avec les outils théoriques et numériques présentées dans ce cours : utilisation des éléments finis pour approcher la solution d'une EDP, comprendre le fonctionnement et saisir les difficultés et restrictions en travaillant avec les diverses variantes d'optimisation de formes et topologique numériques (paramétrisation avec des fonctions densité, optimisation géométrique, optimisation topologique).

Les sessions pratiques sont basées sur le logiciel libre `FreeFem++` qui permet de traiter de manière efficace tout le contenu du cours. Un manuel détaillé est disponible en ligne à l'adresse suivante

<http://www3.freefem.org/ff++/ftp/freefem++doc.pdf>

et une documentation en ligne peut être trouvée à l'adresse

<https://freefem.org/index.html>

Pour ceux n'ayant jamais travaillé en `FreeFem++` avant, une introduction des principales fonctionnalités est donnée dans la première section ci-dessous.

Un grand merci à Charles Dapogny et Grégoire Allaire pour leur aide et conseils concernant ce cours.

1. INTRODUCTION `FreeFem++`

Le logiciel `FreeFem++` permet de résoudre des équations aux dérivées partielles avec un niveau minimum de codage. Le langage de programmation respecte les mêmes règles que `C++` : il faut définir chaque variable utilisée, finir chaque commande par `;`, etc. Un programme `FreeFem++` contient trois parties essentielles :

- (1) Maillage : le domaine de calcul D doit être partitionné en triangles en 2D ou tétraèdres en 3D.
- (2) Espace éléments finis : une fois le maillage créé, il est possible de créer un espace éléments finis
- (3) Assemblage et résolution du système linéaire associé

1.1. Maillage et espace éléments finis. `FreeFem++` est capable de construire un maillage à partir d'une paramétrisation de la frontière en 2D. Le cas 3D est beaucoup plus difficile et va être évoqué vers la fin du cours.

Les mots clés pour construire le maillage sont `border` et `buildmesh`. La commande `border` permet de construire une courbe paramétrée. Exemple :

```
border left(t=0,1){x=0;y=1;label=1;};
border bot(t=0,1){x=t;u=0;label=1;}
```

Notez que dans la définition d'une frontière on prend en compte l'intervalle de variation du paramètre, des formules pour les coordonnées $x(t), y(t)$ et un entier pour pouvoir faire une référence directe quand on veut imposer une condition au bord ou calculer une intégrale sur un bord. Il est possible d'afficher une partie du bord avec la commande `plot`. Pour chaque variable type `border` il y a besoin d'un entier qui précise combien de points on prend sur la courbe associée. On peut afficher plusieurs parties de la frontière en utilisant l'opérateur `+`. Exemple :

```
plot(left(10)+bot(20));
```

La commande qui permet de construire le maillage est `buildmesh`. Un exemple d'usage est

```
mesh Th = buildmesh(left(10)+bot(20)+right(10)+top(20));
```

Il est essentiel que la famille des courbes font une courbe fermée sans auto-intersections orientée dans le sens inverse des aiguilles du montre. Pour debugger il est possible d'utiliser `plot` avant, pour se convaincre que les courbes se relient bien et que les orientations sont correctes. Si jamais une courbe n'a pas la bonne orientation, il suffit de changer le signe de l'entier qui donne le nombre des points sur la courbe. Pour plus de détails et exemples voir les codes fournis.

Une fois le maillage créé il est possible de créer un espace éléments finis. La commande est `fespace` et un exemple d'usage est

```
fespace Vh(Th,P1)
```

Observez que le premier argument de `fespace` est le maillage `Th` et le deuxième est le type d'espace éléments finis. `FreeFem++` permet d'utiliser une multitude d'éléments finis, mais pour ce cours et pour commencer on pourra se restreindre aux éléments finis `P0, P1, P2`.

Une fois l'espace éléments finis créée, il est possible de définir des fonctions éléments finis :

```
Vh u,v,w;
```

Il est possible de modifier ces fonctions éléments finis en faisant des opérations usuelles et de les afficher, toujours avec la commande `plot`. Exemple :

```
u = x; // ceci est un commentaire
v = y;
w = u+v;
plot(w,fill=1); // pour d'autres options d'affichage voir le manuel
```

Question 1. *Construire et affichez des maillages pour des géométries simples : disque, carré, etc. Testez la possibilité d'associer des différents labels aux différents parties du bord et observez si ces modifications sont visibles au moment de l'affichage.*

1.2. Équation de Laplace. Pour un domaine D considérons l'équation de la chaleur

$$\begin{cases} -\operatorname{div}(h\nabla u) = f & \text{dans } D \\ u = 0 & \text{sur } \partial D \end{cases}$$

où h représente la conductivité et f est une source ou puits de chaleur dans D . Si $h \equiv 1$ alors on obtient l'équation de Laplace $-\Delta u = f$ dans D .

L'aspect clé pour appliquer la méthode des éléments finis pour résoudre une telle équation est d'associer une formulation variationnelle. Pour obtenir la formulation variationnelle on multiplie l'équation par une fonction test v et on intègre sur D . La dernière étape est d'appliquer une formule d'intégration par parties ou la formule de Green pour

obtenir une expression symétrique en u et en v . Voir : https://en.wikipedia.org/wiki/Integration_by_parts#Higher_dimensions

Pour l'équation ci-dessus la formulation variationnelle associée est

$$\int_D h \nabla u \nabla v = \int_D f v, \quad \forall v \in H_0^1(D).$$

Question 2. Vérifiez bien que cette formulation est équivalente à l'équation de la chaleur.

L'implémentation en FreeFem++ est faite de la manière suivante :

```
// Definition d'un probleme variationnel
problem heat(u,v) = int2d(Th)(h*(dx(u)*dx(v)+dy(u)*dy(v)))
                  -int2d(Th)(f*v)
                  +on(labels,u=0);
// Resolution, une fois le probleme defini
heat;
// Affichage de la solution
plot(u,fill=1);
```

Observez que les opérateurs `dx`, `dy` permettent de calculer les dérivées partielles d'une fonction éléments finis. L'opérateur `+on` permet d'ajouter des conditions type Dirichlet $u=f$ sur certaines parties du bord définies par un label.

Question 3. Résolvez l'équation en chaleur en FreeFem++ pour des diverses géométries, avec des sources variables et en changeant les labels pour avoir des diverses conditions au bord.

Qu'observez vous sur les parties où on n'impose aucune condition de Dirichlet ?

Question 4. (supplémentaire) Il est possible de résoudre des problèmes plus complexes avec des conditions de type Robin sur certaines parties du bord : $\partial_n u + cu = 0$.

(a) Comment la formulation variationnelle change si vous considérez des conditions de type Robin ?

(b) Implémentez la formulation variationnelle obtenue et interprétez les résultats numériques.

La qualité de la solution dépend de la qualité du maillage. Il est possible de raffiner le maillage en utilisant la commande `adaptmesh`. Un usage habituel est :

```
Th = adaptmesh(Th,0.08,IsMetric=1,nbvx=20000);
```

et d'autres options sont disponibles si vous consultez le manuel. Sur un maillage plus fin l'approximation sera plus juste. Il est aussi possible d'augmenter le degré des fonctions dans l'espace d'éléments finis en considérant des éléments \mathbb{P}_2 au lieu de \mathbb{P}_1 , mais le succès de cette approche dépend de la régularité théorique de la solution u . En pratique, utiliser des éléments \mathbb{P}_1 et des maillages fins suffit pour atteindre la précision souhaitée. L'usage des espaces éléments finis plus complexe peut être justifié par un travail mathématique. Il y a des cas où il est nécessaire d'avoir des espaces éléments finis mieux adaptés, mais ceci sort du cadre de ce cours introductif.

1.3. L'élasticité linéarisée. Le deuxième modèle utilisé couramment dans l'optimisation des structures est l'élasticité linéarisée. L'équation correspondante relie les déplacements de la structure sous diverses forces externes sous l'hypothèse que les déformations sont petites.

La structure Ω est considérée bi-dimensionnelle, pour simplifier les coûts de calcul. Sa frontière $\partial\Omega$ contient une région d'accrochage Γ_D où les déplacements sont égaux à zéro. Des chargements surfaciques sont appliqués sur une autre partie $\Gamma_N \subset \partial\Omega$. Sur le reste de la frontière $\Gamma = \partial\Omega \setminus (\Gamma_D \cup \Gamma_N)$ il n'y a pas de chargement.

Le système d'élasticité linéarisée s'écrit sous la forme :

$$\begin{cases} -\operatorname{div}(Ae(u)) = f & \text{dans } \Omega \\ u = 0 & \text{sur } \Gamma_D \\ Ae(u)n = g & \text{sur } \Gamma_N \\ Ae(u)n = 0 & \text{sur } \Gamma \end{cases}$$

où :

- f est un chargement volumique dans Ω
- le déplacement $u = (u_1, u_2) : \Omega \rightarrow \mathbb{R}^2$ appartient à l'espace $H_{\Gamma_D}^1(\Omega)^2$
- le tenseur des déformations $e(u)$ est défini par :

$$e(u) = \frac{1}{2}(\nabla u + \nabla^T u), \quad (e(u))_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right), \quad i, j = 1, 2$$

- le tenseur des contraintes $\sigma = Ae(u)$ est relié à $e(u)$ par la loi de Hooke

$$\sigma = Ae(u) = 2\mu e(u) + \lambda \operatorname{tr}(e(u)) Id$$

et λ et μ sont les coefficients de Lamé qui caractérisent le comportement élastique du matériau. Ils sont liés aux quantités physiques E (le module de Young) et ν (le coef. de Poisson) par les relations

$$\lambda = \frac{E}{2(1+\nu)}, \quad \mu = \frac{\nu E}{2(1+\nu)(1-\nu)}$$

Pour les tests numériques on prendra $E = 1$ et $\nu = 1/3$. En pratique il faut choisir les coefficients associées au matériel présent dans Ω .

Pour résoudre le système d'élasticité linéarisée en **FreeFem++** il faut transformer l'EDP dans une formulation variationnelle.

Question 5. Vérifiez que la formulation variationnelle pour le système de l'élasticité linéarisée est donnée par

$$\int_D Ae(u) : e(v) dx = \int_D f \cdot v dx + \int_{\Gamma_N} g \cdot v d\sigma$$

pour tout $v \in H_{\Gamma_D}^1(\Omega)^2$.

Question 6. Utilisez le code `elas.edp` pour vérifier le comportement de l'algorithme **FreeFem++** pour résoudre le système d'élasticité linéarisée. Notez l'utilisation de la commande `movemesh` pour observer la structure déformée.

(a) Vérifiez le comportement du résultat par rapport aux paramètres du problème.

(b) Construisez votre propre structure avec les chargements souhaitées et observez la déformation finale donnée par l'EDP.

Parfois la structure n'est pas connue avec un maillage, mais comme un sous-ensemble d'un maillage donné. Il est souvent convenable de représenter la forme Ω comme un sous-niveau d'une fonction de niveau φ définie sur le domaine de référence D . La fonction φ , qui en pratique peut être une fonction \mathbb{P}_1 vérifie

$$\begin{cases} \varphi(x) < 0 & \text{pour } x \in \Omega \\ \varphi(x) = 0 & \text{pour } x \in \partial\Omega \\ \varphi(x) > 0 & \text{pour } x \in D \setminus \bar{\Omega} \end{cases}$$

Il est possible d'approcher la solution du système d'élasticité linéarisée sur Ω en résolvant une EDP sur D en changeant tout simplement la loi de Hooke A :

$$A_\varepsilon(x) = \begin{cases} A & \text{si } x \in \Omega \Leftrightarrow \varphi(x) < 0 \\ \varepsilon A & \text{si } x \in D \setminus \Omega \Leftrightarrow \varphi(x) \geq 0 \end{cases}$$

Cette méthode s'appelle la *méthode du matériau ersatz*, et elle est très utile quand on veut optimiser une structure en gardant un maillage fixe.

Question 7. Utilisez le code `elasers.edp` pour vérifier le comportement de la méthode du matériau ersatz. Sauvegardez les résultats et créez un script qui compare la différence entre les deux solutions obtenues sur le maillage de Ω . Les fonctions `readsol` et `volfrac` incluses dans le fichier `tools.idp` peuvent être utiles.

Question 8. (codage + quelques notions de géométrie, facultatif) Observez le rôle de la fonction `volfrac` et son usage. Proposez une alternative plus simple.

1.4. Problèmes aux valeurs propres. Un autre aspect important en mécanique structurelle sont les modes de vibrations ou les modes propres. On reprend le système d'élasticité avec un chargement volumique particulier : $f = \lambda u$, i.e. on cherche les chargements volumiques qui produisent des déplacements proportionnels.

$$\begin{cases} -\operatorname{div}(Ae(u)) & = \lambda u & \text{dans } \Omega \\ u & = 0 & \text{sur } \Gamma_D \\ Ae(u)n & = 0 & \text{sur } \Gamma_N \\ Ae(u)n & = 0 & \text{sur } \Gamma \end{cases}$$

Un résultat classique nous assure que les valeurs propres forment une suite (λ_n) de nombres positives tels que $\lim_{n \rightarrow \infty} \lambda_n = +\infty$. Les fonctions propres u_n associées sont normalisées : par exemple $\int_D (u_1^2 + u_2^2) dx = 1$.

Du point de vue physique, ces modes de résonance sont importantes car ils peuvent générer des réponses importantes de la structure, affectant son intégrité. Cherchez sur internet des informations sur *Tacoma bridge* : [https://en.wikipedia.org/wiki/Tacoma_Narrows_Bridge_\(1940\)](https://en.wikipedia.org/wiki/Tacoma_Narrows_Bridge_(1940)).

Question 9. Trouver une formulation variationnelle pour le problème aux valeurs propres ci-dessus. Écrivez la variante discrète du problème en considérant un espace d'éléments finis V_h . Observez l'implémentation en `FreeFem++` donnée dans `elasev.edp`.

Question 10. (plus simple! mais pas essentiel) Répétez le même exercice pour le problème de Laplace : écrivez un algorithme `FreeFem++` qui vous permet de trouver numériquement les premières 10 valeurs propres pour l'équation de Laplace sur un domaine Ω . Vous pouvez prendre les exemples classiques : le carré unité, le disque unité et comparer vos résultats avec les valeurs analytiques connues. Observez ce qui se passe si :

- (a) Le domaine est transformé par une homothétie.
- (b) Le domaine devient presque plat.
- (c) Le domaine n'est pas connexe.