Approximation numérique des EDP

Cours EDP : approches variationnelles (ENPC 1A)

DM distribué le 12 mai 2025, à rendre pour le 2 juin 2025

Le code associé (fourni sous la forme de Jupyter notebook) est disponible sur le site web du cours : `http://cermics.enpc.fr/~legoll/edpef.html`

Nous avons vu en cours comment étudier le caractère bien posé d'une équation aux dérivées partielles : on établit l'équivalence entre cette EDP et une formulation variationnelle, et on établit le caractère bien posé de celle-ci, par exemple à l'aide du théorème de Lax-Milgram. Ceci donne donc un résultat (d'existence et d'unicité) abstrait, mais ne permet pas de calculer la solution. En dehors de cas très particuliers (de géométrie du domaine dans lequel on travaille, de coefficients dans l'équation, etc), il n'existe en fait pas de formule explicite pour la solution d'une EDP générale.

L'objectif de ce devoir est d'étudier une méthode d'approximation. On va introduire un problème approché dont on va pouvoir calculer explicitement la solution, et on va montrer que celle-ci est une bonne approximation de la solution du problème de référence.

On se donne

— un espace de Hilbert V (on note $\langle \cdot, \cdot \rangle$ son produit scalaire et $\| \cdot \|$ la norme associée) ;

— une forme linéaire b qu'on suppose être continue sur V, au sens où il existe une constante C telle que
$$\forall v \in V, \quad |b(v)| \leq C \, \|v\|; \tag{1}$$

— une forme bilinéaire a qu'on suppose être continue sur $V \times V$, au sens où il existe une constante M telle que
$$\forall u, v \in V, \quad |a(u,v)| \leq M \, \|u\| \, \|v\|, \tag{2}$$
et qu'on suppose aussi être coercive sur V, au sens où il existe une constante $\alpha > 0$ telle que
$$\forall v \in V, \quad a(v,v) \geq \alpha \, \|v\|^2. \tag{3}$$

On considère la formulation variationnelle suivante :
$$\left\{ \begin{array}{l} \text{Trouver } u \in V \text{ tel que} \\ \forall v \in V, \qquad a(u,v) = b(v). \end{array} \right. \tag{4}$$

Le théorème de Lax-Milgram indique que le problème (4) est bien posé.

1 Approche de Galerkin

L'approche de Galerkin [1] consiste à remplacer, dans (4), l'espace V (qui est typiquement de dimension infinie, penser au cas vu en cours où $V = H^1_0(\Omega)$) par un espace V_h de dimension finie. L'approximation numérique u_h de $u \in V$ va être cherchée dans V_h. L'espace V_h est ainsi appelé *espace d'approximation*. L'indice h symbolise le fait que V_h est de dimension finie (cf. la section 2 ci-dessous pour un exemple explicite).

1. du nom du mathématicien russe Boris Grigorievich Galerkin, 1871–1945.

Dans ce travail, on suppose que

$$V_h \subset V. \tag{5}$$

On dit alors que la méthode d'approximation est *conforme*. La méthode de Galerkin consiste à approcher le problème (4) par le problème suivant :

$$\left\{ \begin{array}{l} \text{Trouver } u_h \in V_h \text{ tel que} \\ \forall v_h \in V_h, \qquad a(u_h, v_h) = b(v_h). \end{array} \right. \tag{6}$$

Question 1. Montrer que le problème (6) est bien posé (i.e. qu'il admet une et une seule solution).

1.1 Ecriture du système linéaire

L'espace V_h est de dimension finie. On note N sa dimension et on se donne une base $(\varphi_1, \ldots, \varphi_N)$ de V_h. On va écrire la solution $u_h \in V_h$ de (6) sous la forme

$$u_h = \sum_{j=1}^{N} u_j \, \varphi_j.$$

Question 2. Montrer que résoudre (6) est équivalent à trouver $U = \begin{pmatrix} u_1 \\ \vdots \\ u_N \end{pmatrix} \in \mathbb{R}^N$ tel que

$$A\,U = B, \tag{7}$$

où A est une matrice de taille $N \times N$, dite matrice de raideur (ou de rigidité), donnée par

$$\forall 1 \leq i, j \leq N, \quad A_{ij} = a(\varphi_j, \varphi_i),$$

et où $B \in \mathbb{R}^N$ est le vecteur défini par

$$\forall 1 \leq i \leq N, \quad B_i = b(\varphi_i).$$

Question 3. On montre que la matrice A possède de bonnes propriétés.
3a. Montrer que la matrice A est définie positive, au sens où

$$\forall \xi \in \mathbb{R}^N, \qquad \xi \cdot A\xi \geq 0$$

et, si $\xi \in \mathbb{R}^N$ est tel que $\xi \cdot A\xi = 0$, alors $\xi = 0$.
3b. On suppose dans cette question uniquement que la forme bilinéaire a est symétrique, au sens où $a(u, v) = a(v, u)$ pour tout $u, v \in V$. Vérifier que la matrice A est alors symétrique.

On voit donc que la résolution de (6) revient à résoudre le système linéaire (7), tâche qu'on peut maintenant faire exécuter par un code informatique. La matrice A qu'il faut inverser possède de bonnes propriétés, directement héritées des propriétés de la forme bilinéaire a.

1.2 Estimation d'erreur

Dans cette partie, on va majorer l'erreur $u - u_h$.

Question 4. Montrer que l'erreur $u - u_h$ satisfait la propriété suivante :

$$\forall v_h \in V_h, \quad a(u - u_h, v_h) = 0. \tag{8}$$

Cette propriété serait-elle vraie en l'absence de (5) ?

Question 5. Il est évident que l'erreur entre u et u_h dépend de la "distance" entre u et l'espace V_h, au sens où

$$\inf_{v_h \in V_h} \|u - v_h\| \leq \|u - u_h\|.$$

On va maintenant montrer que l'inégalité inverse est aussi vraie, à une constante multiplicative près indépendante de V_h.

5a. Montrer que

$$\alpha \|u - u_h\|^2 \leq a(u - u_h, u - u_h).$$

5b. Montrer que, pour tout $v_h \in V_h$, on a

$$\alpha \|u - u_h\|^2 \leq a(u - u_h, u - v_h).$$

5c. En déduire que

$$\|u - u_h\| \leq \frac{M}{\alpha} \inf_{v_h \in V_h} \|u - v_h\|. \tag{9}$$

La quantité $\inf_{v_h \in V_h} \|u - v_h\|$ est appelée erreur de meilleure approximation. Elle est indépendante de la formulation variationnelle étudiée, et ne dépend que des qualités d'approximation de V_h. Si cette erreur de meilleure approximation tend vers 0 lorsque la dimension de V_h tend vers $+\infty$, alors l'erreur $u - u_h$ tend aussi vers 0, et avec le même taux de convergence.

Question 6. On suppose que la forme bilinéaire a est symétrique. On rappelle alors que le problème (4) revient à minimiser l'énergie $J(v) = \dfrac{1}{2} a(v, v) - b(v)$ sur l'ensemble des $v \in V$.

6a. Montrer que $J(u) = -\dfrac{1}{2} a(u, u) = -\dfrac{1}{2} b(u)$ et que $J(u) \leq 0$.

6b. Montrer que $J(u) \leq J(u_h)$ et qu'on a aussi $J(u_h) \leq 0$.

6c. Montrer que

$$J(u_h) - J(u) = \frac{1}{2} a(u - u_h, u - u_h). \tag{10}$$

2 Mise en oeuvre en dimension $d = 1$

On va mettre en oeuvre dans le cas le plus simple la méthode décrite ci-dessus. On se place en dimension $d = 1$, on considère l'ouvert $\Omega = (0, 1)$, une fonction $f \in L^2(\Omega)$, et le problème

$$-u'' = f \ \text{ dans } \mathcal{D}'(\Omega), \qquad u(0) = 0, \qquad u(1) = 0. \tag{11}$$

La formulation variationnelle associée est le problème (4) avec $V = H_0^1(\Omega)$,

$$a(u, v) = \int_\Omega u'\, v', \qquad b(v) = \int_\Omega f\, v.$$

Pour tout entier N, on pose $h = 1/(1+N)$ et on introduit les points $x_j = j\,h$, pour tout $0 \le j \le 1+N$. Ces points forment une grille régulière de Ω, et on a en particulier $x_0 = 0$, $x_1 = h$, ..., $x_N = 1-h$ et $x_{1+N} = 1$.

Pour tout $1 \le j \le N$, on considère[2] la fonction φ_j dont le support est $[x_{j-1}, x_{j+1}]$, qui est affine sur $[x_{j-1}, x_j]$ et $[x_j, x_{j+1}]$ et qui est telle que $\varphi_j(x_{j-1}) = \varphi_j(x_{j+1}) = 0$ et $\varphi_j(x_j) = 1$.

Question 7. Pour un $1 \le j \le N$, dessiner la fonction φ_j et montrer que $\varphi_j \in H^1_0(\Omega)$. Montrer aussi que la famille $\{\varphi_j\}_{1 \le j \le N}$ est libre. Dans toute la suite, on prend

$$V_h = \mathrm{Vect}\{\varphi_j,\ 1 \le j \le N\}.$$

Question 8. Compléter le code fourni pour implémenter le calcul de la matrice de raideur A.

Question 9. On s'intéresse maintenant au calcul du vecteur B.
9a. On pourrait utiliser la formule d'intégration numérique suivante :

$$B_j = b(\varphi_j) = \int_0^1 f\,\varphi_j \approx h\,f(x_j)\,\varphi_j(x_j) = h\,f(x_j).$$

Expliquer les étapes permettant d'aboutir à cette approximation.
9b. En pratique, cette approximation n'est pas assez précise et une autre approximation est implémentée dans le code. Quelle règle d'intégration numérique a été employée ?

Question 10. On considère deux choix pour le membre de droite : $f(x) = \sin(2\,\pi\,x)$ et $f(x) = 1$. Dans les deux cas, déterminer la solution exacte de (11) et calculer l'énergie $J(u)$ associée. Implémenter ces expressions dans le code fourni.

Question 10. On rappelle que $U = \begin{pmatrix} u_1 \\ \vdots \\ u_N \end{pmatrix} \in \mathbb{R}^N$ est le vecteur qui rassemble les degrés de liberté de $u_h \in V_h$: $u_h(x) = \displaystyle\sum_{j=1}^N u_j\,\varphi_j(x)$. En utilisant la Question 6, montrer que l'énergie $J(u_h)$ associée à la solution numérique u_h est directement liée à la quantité $U \cdot AU$, où A est la matrice de raideur introduite ci-dessus. Implémenter le calcul de $J(u_h)$ dans le code.

Question 11. On s'intéresse à l'erreur entre u et u_h. Plus précisément, on va évaluer la quantité $\sqrt{J(u_h) - J(u)}$, qui est directement liée (sur la base de (10)) à l'erreur $u - u_h$. Afin de s'intéresser à une erreur relative, on considère

$$\mathrm{err} = \sqrt{\frac{J(u_h) - J(u)}{|J(u)|}}. \tag{12}$$

Implémenter le calcul de (12). Pour les deux choix de membre de droite, calculer (12) pour plusieurs valeurs de la taille de grille h. Vérifier la convergence de cette erreur vers 0 lorsque h décroit et estimer empiriquement un taux de convergence.

2. On prendra garde au fait que la numérotation dans ce document ne correspond pas forcément à la numérotation dans le code (ceci est lié au fait que Python numérote à partir de 0 et que la première fonction φ_j utile ici correspond à l'indice $j = 1$).