TP informatique du cours d'Analyse spectrale et applications aux équations aux dérivées partielles à rendre pour le 9 mai 2017

Virginie Ehrlacher et Frédéric Legoll

18 avril 2017

Vous pouvez télécharger les documents relatifs à ce TP sur le lien suivant :

http ://cermics.enpc.fr/∼ehrlachv/Cours_AnaSpec/TP_spectral.zip

Préliminaires : Résolution d'un problème aux valeurs propres généralisé

Soit $V \subset H$ deux espaces de Hilbert tel que l'injection de V dans H soit compacte, et que V soit dense dans H. Soit $a : V \times V \to \mathbb{R}$ une forme bilinéaire symétrique continue et coercive sur V. Le but de ce TP est d'étudier des méthodes numériques pour résoudre le problème aux valeurs propres : trouver $(\lambda, u) \in \mathbb{R} \times V$ solution de

$$\forall v \in V, \quad a(u, v) = \lambda \langle u, v \rangle_H. \tag{1}$$

On rappelle alors le résultat suivant (Théorème 3.1 p.60 du cours). Les valeurs propres du problème (1) forment une suite croissante de réels $(\lambda_k)_{k \geqslant 1}$ tendant vers $+\infty$ et il existe alors une base hilbertienne $(u_k)_{k \geqslant 1}$ de H de vecteurs propres associés, i.e. tels que

$$\forall v \in V, \quad a(u_k, v) = \lambda_k \langle u_k, v \rangle_H. \tag{2}$$

On a vu en cours qu'une méthode classique pour approcher numériquement les solutions du problème aux valeurs propres (1) était d'utiliser une méthode d'approximation de Galerkin. Plus précisément, soit $V_h = \mathrm{Vect}\{\phi_1, \cdots, \phi_n\}$ un sous-espace de dimension finie de V, dont $(\phi_i)_{1 \leqslant i \leqslant n}$ forme une base (ce sont par exemple les fonctions de base d'une méthode d'éléments finis). La méthode d'approximation de Galerkin consiste à approcher $(\lambda_k, u_k) \in \mathbb{R} \times V$ par $(\lambda_{k,h}, u_{k,h}) \in \mathbb{R} \times V_h$ solutions du problème aux valeurs propres discrétisés :

$$\forall v \in V_h, \quad a(u_{k,h}, v) = \lambda_{k,h} \langle u_{k,h}, v \rangle_H. \tag{3}$$

On note dans la suite $K_h, M_h \in \mathbb{R}^{n \times n}$ les matrices définies par : pour tout $1 \leqslant i, j \leqslant n$,

$$(K_h)_{i,j} = a(\phi_i, \phi_j) \quad \text{et } (O_h)_{i,j} = \langle \phi_i, \phi_j \rangle_H.$$

1. On cherche $u_{k,h}$ solution de (3) sous la forme $u_{k,h} = \sum_{j=1}^n U_{k,h}^j \phi_j$ avec $U_{k,h} = \left(U_{k,h}^j\right)_{1 \leqslant j \leqslant n} \in \mathbb{R}^n$. Rappeler le problème matriciel dont $(\lambda_{k,h}, U_{k,h}) \in \mathbb{R} \times \mathbb{R}^n$ sont solutions.

1 Approximation des valeurs propres et vecteurs propres d'un opérateur sous forme divergence

Le but de cette première partie du TP est d'observer les vitesses de convergence d'une méthode d'approximation en éléments finis $\mathbb{P}_1$ pour un problème aux valeurs propres d'un opérateur sous forme divergence.

Plus précisément, soit $d \in \mathbb{N}^*$, $\Omega = (0,1)^d$ et $c : \Omega \to \mathbb{R}$ une fonction mesurable telle qu'il existe $0 < \gamma \leqslant \Gamma < +\infty$ tels que

$$\gamma \leqslant c(x) \leqslant \Gamma, \quad \text{pour presque tout } x \in \Omega.$$

On considère alors le problème aux valeurs propres suivant : Trouver $(u_k, \lambda_k) \in H_0^1(\Omega) \times \mathbb{R}$ tels que

$$\begin{cases} -\mathrm{div}\,(c\nabla u_k) = \lambda_k u_k & \text{dans } \mathcal{D}'(\Omega), \\ u_k = 0 & \text{sur } \partial\Omega. \end{cases} \tag{4}$$

1. Ecrire le problème (4) sous forme variationnelle, i.e. sous la forme (2) avec $V = H_0^1(\Omega)$, $H = L^2(\Omega)$ et
$$\forall v, w \in H_0^1(\Omega) \quad a(v,w) := \int_\Omega c\nabla v \cdot \nabla w.$$
Montrer que ces espaces et cette forme bilinéaire vérifient bien les hypothèses du Théorème 3.1 du cours.

Dans la suite de cette partie, nous supposerons que $d = 1$ et $\Omega = (0,1)$. Nous allons considérer une méthode d'approximation du problème (4) par des éléments finis $\mathbb{P}_1$. Nous considérons dans toute la suite que la fonction $c : (0,1) \to \mathbb{R}$ est définie par : pour tout $x \in (0,1)$,

$$c(x) := \begin{cases} 2 \text{ si } 0.75 \leqslant x \leqslant 1, \\ 3 \text{ si } 0.125 \leqslant x \leqslant 0.375, \\ 1 \text{ sinon.} \end{cases}$$

Pour $N \in \mathbb{N}^*$, on définit $h_N = \frac{1}{N}$ une subdivision régulière de $(0,1)$ définie comme suit : pour tout $0 \leqslant i \leqslant N$, $x_i^N = ih_N$ de telle sorte que $x_0^N = 0 < x_1^N < \cdots < x_N^N = 1$. De plus, pour tout $1 \leqslant i \leqslant N-1$, on définit $\phi_i^N(x)$ la fonction chapeau centrée en x_i^N par :

$$\phi_i^N(x) = \begin{cases} 1 + \frac{x - x_i^N}{h_N} \text{ si } x_{i-1}^N \leqslant x \leqslant x_i^N, \\ 1 + \frac{x_i^N - x}{h_N} \text{ si } x_i^N \leqslant x \leqslant x_{i+1}^N. \\ 0 \text{ sinon.} \end{cases}$$

On notera dans la suite $V_N = \mathrm{Vect}\left\{\phi_1^N, \cdots, \phi_{N-1}^N\right\} \subset H_0^1(0,1)$.

Remarquons ici que si $N = 2^p$ pour un certain entier $p \geqslant 3$, alors la fonction $c(x)$ défini ci-dessus est constante sur chaque intervalle de la forme (x_{i-1}^N, x_i^N) pour tout $1 \leqslant i \leqslant N$. Dans toute la suite de cette partie, nous allons supposer que N prendra toujours des valeurs de cette forme. On notera dans la suite $\lambda_{k,N}$ et $u_{k,N}$ les approximations de Galerkin des valeurs propres λ_k et u_k du problème (4) obtenues avec l'espace de discrétisation V_N.

Entrer dans le répertoire **Vitesse_convergence_1D** . Ouvrir les fichiers *Diffusion.py*, *Matrices.py* et *resolution_ref.py* (mais ne les modifiez pas !). Le fichier *Diffusion.py* contient la définition du coefficient de diffusion $c(x)$. Le fichier *Matrices.py* contient les routines qui vont permettre d'assembler les matrices de masse et de rigidité des problèmes aux valeurs propres que nous allons considérer. Enfin, le fichier *resolution_ref.py* contient un code qui permet de résoudre le problème discrétisé (3) associé au problème (4) avec une méthode d'éléments finis $\mathbb{P}_1$ associé à une grille de discrétisation très fine ($N_{ref} = 2048 = 2^{11}$).

3. Faire tourner ce code en utilisant la commande

   python resolution_ref.py

   Tracer la fonction $x \in (0,1) \mapsto c(x)$. Quelle est sont les valeurs des premières valeurs propres $\lambda_{k,N_{ref}}$ pour $1 \leqslant k \leqslant 5$? Est-ce que la première valeur propre $\lambda_{1,N_{ref}}$ est simple ? Tracer la fonction $x \in (0,1) \mapsto u_{1,N_{ref}}(x)$.

4. Soit $u, w \in V_{N_{ref}}$ tels que $u(x) = \sum_{j=1}^{N_{ref}-1} U^j \phi_j^{N_{ref}}(x)$ et $w(x) = \sum_{j=1}^{N_{ref}-1} W^j \phi_j^{N_{ref}}(x)$. Notons $U = (U^j)_{1\leqslant j\leqslant N_{ref}-1}$ et $W = (W^j)_{1\leqslant j\leqslant N_{ref}-1}$. Montrer que

   $$\|u-w\|^2_{L^2(0,1)} = (U-W)^T M_{ref}(U-W) \quad \text{et } \|u'-w'\|^2_{L^2(0,1)} = (U-W)^T D_{ref}(U-W)$$

   où $M_{ref}, D_{ref} \in \mathbb{R}^{(N_{ref}-1)\times(N_{ref}-1)}$ sont des matrices symétriques dont on donnera les expressions en fonction des $\left(\phi_j^{N_{ref}}(x)\right)_{1\leqslant j\leqslant N_{ref}-1}$.

5. Ouvrir le fichier *convergence.py*. Les premières lignes du code permettent de charger les résultats du calcul de référence que vous venez de lancer. Remplir la partie du code correspondant à la définition des matrices M_{ref} et D_{ref}.

6. Remplir la partie du code qui correspond à la résolution du problème discrétisé pour des valeurs de $N = 2^p$ plus petite que $N_{ref} = 2^{11}$.

7. La fonction $u_{1,N}(x)$ vérifie $u_{1,N}(x) = \sum_{j=1}^{N-1} U^j_{1,N}\phi^N_j(x)$. Montrer que comme $N = 2^p$ avec $p < 11$, alors $\phi^N_j \in V_{N_{ref}}$ pour tout $1 \leqslant j \leqslant N-1$, et que par conséquent $u_{1,N} \in V_{N_{ref}}$. Dans la suite, on note $\left(W^j_{1,N}\right)_{1\leqslant j\leqslant N_{ref}-1}$ les coordonnées de $u_{1,N}$ dans la base $\phi_1^{N_{ref}}, \cdots, \phi^{N_{ref}}_{N_{ref}-1}$, si bien que

   $$u_{1,N}(x) = \sum_{j=1}^{N_{ref}-1} W^j_{1,N}\phi_j^{N_{ref}-1}(x).$$

   Attention ! On ne demande pas de calculer le vecteur $W_{1,N}$ en fonction de $U_{1,N}$!

8. Remplir la partie du code pour calculer les erreurs

   $$\|u_{1,N}-u_{1,N_{ref}}\|^2_{L^2(0,1)}, \quad \|u_{1,N}+u_{1,N_{ref}}\|^2_{L^2(0,1)}, \quad \|u'_{1,N}-u'_{1,N_{ref}}\|^2_{L^2(0,1)}, \quad \|u'_{1,N}+u'_{1,N_{ref}}\|^2_{L^2(0,1)},$$

   puis en déduire que l'erreur en norme L^2 (au carré) est donnée par

   $$\mathrm{err}_{L^2} = \min\left(\|u_{1,N}-u_{1,N_{ref}}\|^2_{L^2(0,1)}, \|u_{1,N}+u_{1,N_{ref}}\|^2_{L^2(0,1)}\right),$$

   et l'erreur en norme H^1 (au carré) est donnée par

   $$\mathrm{err}_{H^1} = \min\left(\|u'_{1,N}-u'_{1,N_{ref}}\|^2_{L^2(0,1)}, \|u'_{1,N}+u'_{1,N_{ref}}\|^2_{L^2(0,1)}\right).$$

   Utiliser pour cela les matrices M_{ref} et D_{ref} définies précédemment.

9. Que peut-on dire du signe de $\lambda_N - \lambda_{N_{ref}}$ si $N = 2^p$ pour un certain $p \in N^*$ et $N \leqslant N_{ref}$? Que peut-on dire de la monotonie de la suite $\left(\lambda_N - \lambda_{N_{ref}}\right)$ pour de telles valeurs de N ? Remplir la partie du code pour caluler l'erreur entre les valeurs propres $\mathrm{err}_\lambda = |\lambda_{1,N} - \lambda_{1,N_{ref}}|$.

10. Tracer le logarithme (en base 2) des différentes erreurs err_{L^2}, err_{H^1} et err_λ en fonction du logarithme de N. Que constatez-vous ?

11. Modifier légèrement les fonctions d'affichage de sorte à afficher sur le même graphique le logarithme de err_{H^1}, err_λ et $\sqrt{\mathrm{err}_{L^2}}$. Que constatez-vous ? Calculer les pentes des droites obtenues. Conclure quant au résultat d'approximation vu en cours. Au vu de vos résultats numériques, comment pensez-vous que l'erreur

    $$\min\left(\|u_k - u_{k,N}\|_{L^2(0,1)}, \|u_k + u_{k,N}\|_{L^2(0,1)}\right)$$

    se comporte en fonction de h_N ?

12. (Facultatif : à ne faire qu'après avoir répondu à toutes les autres questions) Tracer de même les courbes d'évolution des erreurs correspondant à l'approximation de la deuxième valeur propre λ_2 et du vecteur propre associé u_2.

2 Perturbation de premier ordre des valeurs propres d'un opérateur

Dans cette section, nous nous intéressons toujours au problème aux valeurs propres (4) mais cette fois-ci en dimension $d = 2$ dans le cas où $\Omega = (0,1)^2$. On introduit une fonction c_0 définie comme suit : pour tout $(x, y) \in (0,1)^2$,

$$c_0(x,y) = \begin{cases} 3 \text{ si } |x - 0.5| < 0.25 \text{ et } |y - 0.5| < 0.25, \\ 2 \text{ si } |x - 0.125| < 0.125, \\ 1 \text{ sinon.} \end{cases}$$

On définit également un coefficient de diffusion *de perturbation* $c_p : (0,1)^2 \to \mathbb{R}$ comme suit : pour tout $(x, y) \in (0,1)^2$,

$$c_p(x,y) = \begin{cases} 1 \text{ si } |x - 0.5| < 0.25 \text{ et } |y - 0.5| < 0.25, \\ 0 \text{ sinon.} \end{cases}$$

Pour tout $\delta \geqslant 0$, on introduit $c_\delta(x) = c_0(x) + \delta c_p(x)$. On note également a_δ la forme bilinéaire associé au problème (4) lorsque $c(x) = c_\delta(x)$. Nous noterons dans la suite $(\lambda_k(\delta))_{k\geqslant 1}$ les valeurs propres associées à la forme bilinéaire a_δ et $(u_k(\delta))_{k\geqslant 1}$ une base orthonormale de $L^2(\Omega)$ composée de vecteurs propres associés.

Soit $k \in \mathbb{N}^*$. Le but de cette partie est d'étudier comment les valeurs propres $\lambda_k(\delta)$ évoluent avec le paramètre de perturbation δ au voisinage de 0. Plus précisément, nous supposerons dans la suite qu'il existe $\delta_0 > 0$ suffisamment petit tel que $\lambda_k(\delta)$ est une valeur propre simple pour tout $\delta \leqslant \delta_0$. Nous admettrons dans la suite que les applications $[0, \delta_0] \ni \delta \mapsto \lambda_k(\delta) \in \mathbb{R}$ et $[0, \delta_0] \ni \delta \mapsto u_k(\delta) \in H_0^1(\Omega)$ sont différentiables.

On notera dans la suite $b : H_0^1(\Omega) \times H_0^1(\Omega) \to \mathbb{R}$ la forme bilinéaire définie paramètre

$$\forall v, w \in H_0^1(\Omega), \quad b(v, w) := \int_\Omega c_p \nabla v \cdot \nabla w.$$

On considère par ailleurs une méthode de Galerkin pour l'approximation de ce problème aux valeurs propres. Soit $W \subset V$ un sous-espace vectoriel fermé de V, et on note $(\lambda_{k,W}(\delta), u_{k,W}(\delta))_{k\geqslant 1}$ les solutions du problème aux valeurs propres :

$$\forall w \in W, \quad a_\delta(u_{k,W}(\delta), w) = \lambda_{k,W}(\delta)\langle u_{k,W}(\delta), w\rangle_H.$$

Noter que l'espace W défini précédemment peut être de dimension finie ou infinie. Par exemple, si $W = V$, $\lambda_{k,W}(\delta) = \lambda_k(\delta)$ et $u_{k,W}(\delta) = u_k(\delta)$. Si W est un sous-espace vectoriel de dimension finie, par exemple engendré par des fonctions éléments finis, le problème ci-dessus correspond à la méthode d'approximation de Galerkin que nous avons vu en cours.

1. Soit $\delta \in [0, \delta_0)$. Montrer que pour tout $v, w \in H_0^1(\Omega)$, $a_\delta(v, w) = a_0(v, w) + \delta b(v, w)$.
2. Montrer que $\lambda_{k,W}(\delta) = a_\delta(u_{k,W}(\delta), u_{k,W}(\delta))$.
3. En déduire que

$$\frac{d}{d\delta}\lambda_{k,W}(\delta) = 2a_\delta\left(u_{k,W}(\delta), \frac{d}{d\delta}u_{k,W}(\delta)\right) + b\left(u_{k,W}(\delta), u_{k,W}(\delta)\right).$$

4. En utilisant le fait que $\|u_{k,W}(\delta)\|^2_{L^2(\Omega)} = 1$ pour tout $\delta \in [0, \delta_0]$, montrer que $\langle u_{k,W}(\delta), \frac{d}{d\delta} u_{k,W}(\delta)\rangle_{L^2(\Omega)} = 0$. En déduire finalement que

$$\frac{d}{d\delta}\lambda_{k,W}(\delta) = b\left(u_{k,W}(\delta), u_{k,W}(\delta)\right). \tag{5}$$

Nous allons vérifier cette formule numériquement. Entrer dans le répertoire **Perturbation_2D**. Pour $N \in \mathbb{N}^*$, les problèmes aux valeurs propres associés aux formes bilinéaires a_δ sont approchés par une méthode de Galerkin en utilisant comme espace de discrétisation

$$V_N = \mathrm{Vect}\{\phi_i^N(x)\phi_j^N(y), \quad 1 \leqslant i, j \leqslant N-1\}$$

où les fonctions ϕ_i^N ont été définies dans la première partie.

Les routines qui définissent les coefficients de diffusion et les matrices de discrétisation se trouvent dans les fichiers *Diffusion.py* et *Matrices.py*. Regarder ces fichiers mais ne pas les modifier !

5. Ouvrir le fichier *2D_divergence.py*. Construire les matrices de discrétisation suivantes :
   - K_0 est la matrice de discrétisation associée à a_0 ;
   - B est la matrice de discrétisation associée à b ;
   - M est la matrice de discrétisation associée au produit scalaire $\langle \cdot, \cdot \rangle_{L^2(\Omega)}$.

   Le paramètre de discrétisation N est choisi à être égal à 40. Nous conserverons cette valeur dans toute la suite.
6. Tracer la valeur du coefficient c_0 dans $(0,1)^2$. Quelle est la valeur de $\lambda_{1,V_N}(0)$? Tracer la valeur de $u_{1,V_N}(0)$ dans $(0,1)^2$.
7. Décommenter la partie du code correspondant à la question. Nous souhaiterions d'abord obtenir une approximation de $\frac{d}{d\delta}\lambda_{1,V_N}(0)$ par une méthode de différences finies. Pour ce faire, nous allons calculer $\lambda_{1,V_N}(\delta)$ pour plusieurs valeurs de δ puis obtenir une approximation en utilisant la formule
$$\frac{d}{d\delta}\lambda_{k,V_N}(0) \approx \frac{\lambda_{1,V_N}(\delta) - \lambda_{1,V_N}(0)}{\delta}.$$
   Remplir les parties du code à compléter. Tracer les différentes valeurs de la dérivée approchée en fonction de δ. Que constate-t-on ?
8. Décommenter la partie du code correspondant à la question. Les coordonnées du vecteur propre $u_{1,V_N}(0)$ dans la base $\left(\phi_i^N(x)\phi_j^N(y)\right)_{1\leqslant i,j\leqslant N-1}$ sont stockées dans le vecteur U. Réécrire la formule (5) en fonction de U et des matrices discrétisées associées au problème. Implémenter cette formule discrète dans le code. Tracer les courbes d'erreur entre les formules approchées pour la dérivées et la formule analytique en fonction de δ. Que constate-t-on ?