

Modes propres de vibration d'une membrane

Le but de l'exercice est de caractériser les modes de vibration $u_k(x, y)$ selon Oz d'une membrane rectangulaire de faible épaisseur. Dans la limite des faibles amplitudes de vibration, ils sont solutions de l'équation d'Helmholtz stationnaire :

$$\partial_x^2 u_k + \partial_y^2 u_k + k^2 u_k = 0 \quad (1)$$

On considère que la membrane rectangulaire est fixée sur ses quatre cotés.

1 Partie Mathématique

La membrane de longueurs L_x et L_y selon les axes principaux est discrétisée en différences finies avec une grille comportant N_x noeuds selon Ox et N_y noeuds selon Oy. On note Δx et Δy les pas de la grille.

1. Donner les expressions de Δx et Δy .
2. On numérote de manière unique les noeuds dans $\llbracket 0, N_x.N_y \rrbracket$. Montrer qu'un choix de numérotation possible est donné par la relation $\text{ind}(ix, iy) = iy * N_x + ix$ où $(ix, iy) \in \llbracket 0, N_x - 1 \rrbracket \times \llbracket 0, N_y - 1 \rrbracket$. Comment parcourt-on la grille? Quelles sont les coordonnées réelles (x, y) d'un noeud de coordonnées entières (ix, iy) en plaçant **l'origine des coordonnées en bas et à gauche**.
3. Donner l'expression discrète en différences finies, de l'opérateur laplacien appliqué à $u(x, y)$ en un point **intérieur** (x, y) de la grille rectangulaire, forme généralisé de celui unidimensionnel vu en cours.

Indication : faire un développement de Taylor de $u(x, y)$ à partir des premiers voisins. On utilisera le développement limité suivant, pour une fonction à deux variables :

$$u(x + dx, y + dy) = u(x, y) + dx \partial_x u + dy \partial_y u + \frac{1}{2}(dx \partial_x + dy \partial_y)^2 u + \vartheta(dx, dy)^3 \quad (2)$$

où (dx, dy) sont des déplacements infinitésimaux. Les dérivées partielles étant évaluées en (x, y) .

Réponse :

$$\Delta u = \frac{u(x + dx, y) + u(x - dx, y) - 2u(x, y)}{dx^2} + \frac{u(x, y + dy) + u(x, y - dy) - 2u(x, y)}{dy^2}$$

4. Montrer que l'on peut écrire l'équation (1) après discrétisation, sous la forme matricielle :

$$\sum_j K_{i,j} u_j = k^2 u_i = \lambda_k u_i \quad (3)$$

Préciser la forme générale de la matrice K sans se soucier des bords.

5. On tient maintenant compte du déplacement nul des noeuds du bord. Dans le cas d'une grille 4×4 de pas de maille Δx et Δy , donner précisément le remplissage de K avec l'expression des termes non-nuls. Dans la suite, on verra comment éliminer les degrés de liberté liés aux noeuds de dirichlet. On admet que pour tout noeud i du bord, on laisse vide la ligne i dans la matrice K . Utiliser le patron de la matrice vide fournie en annexe.

2 Modes propres avec conditions de dirichlet

On présente ici une méthode générale pour calculer les modes propres en tenant compte des conditions de vibration nulle sur le bord .

En notant N_d le nombre de noeuds où la condition de dirichlet est imposée, le nombre de degrés de liberté se réduit alors à $N_{dof} = N - N_d$ et correspond ici au nombre de noeuds pouvant vibrer.

2.1 Algorithme

L'algorithme consiste à former une liste l_d où apparaissent de manière unique les numéros des noeuds fixes du bord.

On forme ensuite une matrice de projection P permettant de passer de l'espace des solutions à N degrés de liberté (étude précédente) à celui restreint à N_{dof} degrés de liberté. Les dimensions de P sont $N_{dof} \times N$ (nombre de lignes \times nombre de colonnes).

On initialise la matrice P avec la matrice identité $\text{Id}(N)$ puis de supprimer toutes les lignes correspondant aux noeuds de dirichlet. Cette technique est bien adaptée à Matlab : si ld est la liste des noeuds de dirichlet sans doublon alors

$P = \text{Id}(N)$; $P(ld, :) = []$;

$$P = \begin{pmatrix} \cancel{1} & \cancel{0} & \cancel{0} & \cancel{0} \\ 0 & 1 & 0 & 0 \\ \cancel{0} & \cancel{0} & \cancel{1} & \cancel{0} \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

FIGURE 1 – exemple de matrice P avec $ld = [1, 3]$

On construit ensuite K_p qui est la matrice projetée de K dans l'espace solution à N_{dof} degrés de liberté : $K_p = PKP^t$.

On résoud l'équation aux valeurs propres : $PKP^t u_p = \lambda_p u_p$.

On obtient alors les valeurs propres λ_p associées aux vecteurs propres u_p .

On reconstruit la solution dans l'espace à N noeuds en appliquant $u = P^t u_p$. Elle vérifie automatiquement, $u(i) = 0$ pour tout noeud i de dirichlet.

3 Mise en application

On vous fournit un embryon de programme `numpy_modes_membrane.py` à compléter, utilisant une structure de données `fdm` pour stocker tous les paramètres de la simulation.

1. Compléter la fonction `_dirichlet` permettant de construire la liste `ld` des noeuds du bord où sont appliquées les conditions de dirichlet. On rappelle que l'instruction `ld.append(e)` permet d'insérer l'élément `e` dans la liste `ld`. Utiliser sous Numpy, la fonction `ld=numpy.unique(ld)` pour éliminer tout doublon dans la liste `ld`.
2. Compléter la fonction `_build_K` permettant de remplir la matrice K pour une grille de taille N_x . On rappelle que toute ligne n de K correspondant à un noeud de dirichlet n'est pas remplie.
Pour tester l'appartenance d'un noeud n dans la liste `ld`, on utilisera l'expression booléenne `n in ld`.
3. Compléter la fonction `solve` permettant de calculer la n^{eme} plus petite valeur propre en module ainsi que le mode propre associé. On utilisera la fonction Numpy `eig`.
4. Pour un système rectangulaire de taille $L_x = 2m$ et $L_y = 1m$, déterminer les 4 premiers modes de basse énergie et comparer à la solution analytique de l'équation d'Helmholtz (1). *Indication* : chercher des solutions de la forme $\sin(k_x x) \cdot \sin(k_y y)$. Reporter les valeurs propres et les contours des modes associés sur votre copie.
5. Faire de même pour $L_x = 1m$ et $L_y = 1m$, expliquer pourquoi les modes d'ordre 2 et 3 semblent différents de ceux calculés analytiquement. *Indication* : regarder la dégénérescence des valeurs propres.

4 Annexe - Nom Prénom

	1.	2.	3.	4.	5.	6.	7.	8.	9.	10	11	12	13	14	15	16
1																
2																
3																
4																
5																
6																
7																
8																
9																
10																
11																
12																
13																
14																
15																
16																

	1.	2.	3.	4.	5.	6.	7.	8.	9.	10	11	12	13	14	15	16
1																
2																
3																
4																
5																
6																
7																
8																
9																
10																
11																
12																
13																
14																
15																
16																