Dans un premier temps, on s'interesse à calculer une approximation de la solution de

$$\begin{cases} -u'' = f \text{ dans }]a,b[, \\ u(a) = \alpha, \\ u(b) = \beta. \end{cases} \tag{1}$$

Exercice 1 : Résolution analytique

Résoudre explicitement l'équation (1) dans les cas particuliers suivants :

$$a = 0,\ b = \pi, \alpha = 0, \beta = 0,\ f(x) = \sin(x),$$
$$a = 0,\ b = 1, \alpha = 0, \beta = 0, f(x) = (x+1)(2-x).$$

Exercice 2 : Résolution par différences finies

On se propose d'utiliser la méthode des différences finies d'ordre 2 pour approcher la solution de (1). Pour un entier N donné, on introduit la suite $(x_i)_{i=0,N+1}$ définie par $x_i = a+ih$ où $h = (b-a)/(N+1)$. On notera U_i la valeur approchée de $u(x_i)$.

a) En chacun des points x_i, $1 \leq i \leq N$ l'équation (1) est approchée par

$$-\frac{1}{h^2}(U_{i+1} + U_{i-1} - 2U_i) = f(x_i) = f_i,$$

et on a les conditions aux limites :

$$U_0 = \alpha \text{ et } U_{N+1} = \beta.$$

Ecrire les N équations d'inconnues U_i, $1 \leq i \leq N$ sous la forme d'un système linéaire $AU = F$ où $U = (U_1, \cdots, U_N)$.

b) Ecrire un script Matlab qui résout le problème des différences finies :

- Définir les données physiques a, b, α, β et f. On pourra choisir les données de l'exercice 1.
- Définir les données numériques : N et h.
- Construire le maillage, i.e. la suite $(x_i)_{0\leq i\leq N+1}$.
- Construire la matrice A. On utilisera la commande `spdiags`.
- Construire le second membre F.
- Résoudre le système et tracer sur une même figure la solution exacte et la solution approchée obtenue.

Exercice 3 : Vérification du code

a) Pour vérifier dans un premier temps la validité du résultat, on tracera sur un même graphique la solution approchée et la solution exacte.

b) Pour plusieurs valeurs de N calculer la norme de l'erreur entre la solution exacte et la solution approchée :

$$\epsilon_N = \max_{1\leq i\leq N} |u(x_i) - U_i|.$$

Tracer l'erreur en fonction de h et retrouver numériquement l'ordre du schéma.

Exercice 4 : Conditions aux limites de type Neumann

On s'intéresse à présent au problème :

$$\begin{cases} -u'' = f \text{ dans }]a, b[, \\ u'(a) = g_a, \\ u(b) = \beta. \end{cases} \tag{2}$$

a) Calculer p, le polynôme d'interpolation qui passe par les points $(0, u_0)$ et (h, u_1). Approcher la dérivée $u'(0)$ par $p'(0)$. Quelle formule reconnait-on ?

b) Implémenter cette discrétisation et calculer l'ordre du schéma global. Quelle remarque fait-on ?

c) Proposer une amélioration.

Exercice 5 : DF pour l'équation de Laplace en 2D

On considère à présent :

$$\begin{cases} -\triangle u = f \text{ dans }]0, 1[\times]0, 1[, \\ u(x, 0) = g_1(x), x \in]0, 1[, \\ u(x, 1) = g_2(x), x \in]0, 1[, \\ u(0, y) = g_3(y), y \in]0, 1[, \\ u(1, y) = g_4(y), y \in]0, 1[. \end{cases} \tag{3}$$

a) Considérer un maillage du carré unité de 5×5 mailles et écrire les équations venant de la discrétisation par différences finies du problème (3). Ecrire le système matriciel correspondant.

b) Généraliser à un maillage de taille quelconque.

c) Programmer la méthode des différences finies pour résoudre (3). On utilisera les fonctions :

- `meshgrid` pour construire le maillage du plan.
- `kron` pour construire une matrice définie par bloc.
- `fliplr`, `flipud` pour 'retourner' les matrices.
- `reshape` pour réorganiser les matrices.
- `surf` pour visualiser la solution.

d) Vérifier votre programme : tracer l'erreur en fonction du pas de maillage.