

TRAVAUX PRATIQUES - RATRAPAGE DU 25 MARS 2019 (3H00)

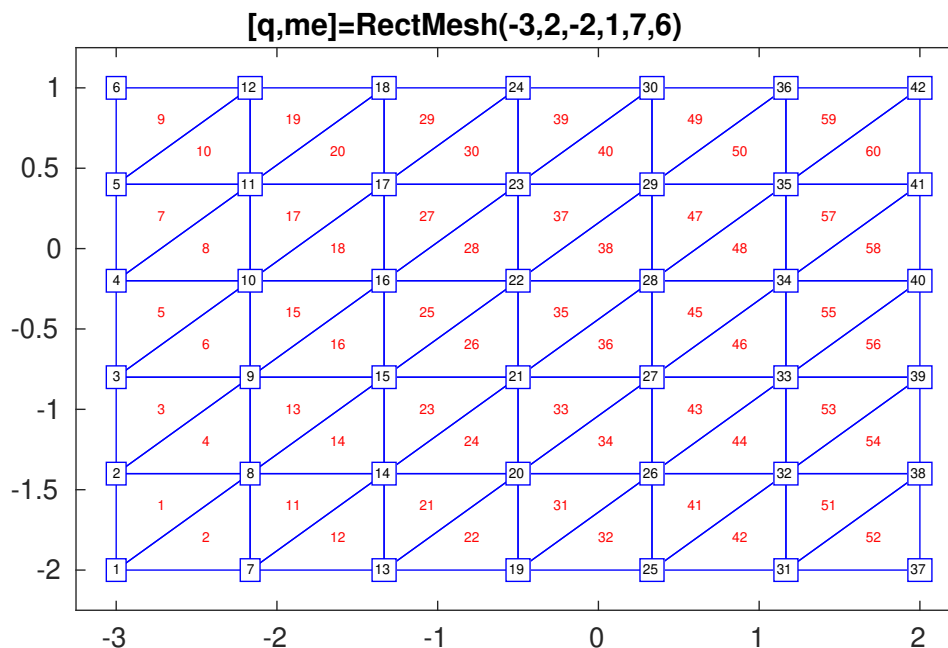
Sans documents, sans calculatrice, sans portable, ...  
Le barème est donné à titre indicatif

**EXERCICE 1 (3 POINTS)**

Pour stocker les informations (minimales) relatives à un maillage, on utilise les tableaux  $\mathbf{q}$  et  $\mathbf{me}$  respectivement tableau des sommets/points et tableau de connectivité :

| nom           | type   | dimension         | descriptif   |
|---------------|--------|-------------------|--|
| $n_q$         | entier | 1                 | nombre total de noeuds (sommets) du maillage   |
| $n_{me}$      | entier | 1                 | nombre de triangles  |
| $\mathbf{q}$  | réels  | $2 \times n_q$    | $\mathbf{q}(\text{il}, i)$ est la $\text{il}$ -ème coordonnée du $i$ -ème sommet, $\text{il} \in \{1, 2\}$ et $i \in \{1, \dots, n_q\}$ . Le $i$ -ème sommet sera aussi noté $\mathbf{q}^i = (q_x^i, q_y^i)$ avec $q_x^i = \mathbf{q}(1, i)$ et $q_y^i = \mathbf{q}(2, i)$   |
| $\mathbf{me}$ | entier | $3 \times n_{me}$ | $\mathbf{me}(\text{jl}, k)$ indice de stockage, dans le tableau $\mathbf{q}$ , du $\text{jl}$ -ème sommet du triangle d'indice $k$ , $\text{jl} \in \{1, 2, 3\}$ et $k \in \{1, \dots, n_{me}\}$ . Pour tout triangle la numérotation des points est dans le <b>sens direct</b> .<br>$\mathbf{q}(:, \mathbf{me}(1, k))$ est le 1er sommet du $k$ -ème triangle, $\mathbf{q}(:, \mathbf{me}(2, k))$ est le 2ème sommet, ... |

**Q. 1** Ecrire une fonction  $[\mathbf{q}, \mathbf{me}] = \text{RectMesh}(a, b, c, d, N_x, N_y)$  retournant les tableaux de points et de connectivité associé au rectangle  $[a, b] \times [c, d]$  avec  $N_x$  points suivant  $x$  et  $N_y$  points suivant  $y$  comme décrit sur la figure ci-dessous :



**Attention**, les numérotations sont celles imposées par la figure et elles diffèrent de celles utilisées en TP et de celles proposées dans le dernier examen. Les nombres encadrés sont les indices/numéros des sommets. Les nombres dans les triangles sont les indices/numéros des triangles.

**EXERCICE 2 (7 POINTS)**

On rappelle tout d'abord quelques définitions.

**Definition 1 (problème de Cauchy)** Soit  $\mathbf{f}$  l'application continue définie par

$$\begin{aligned} \mathbf{f} : [t^0, t^0 + T] \times \mathbb{R}^d &\longrightarrow \mathbb{R}^d \\ (t, \mathbf{y}) &\longmapsto \mathbf{f}(t, \mathbf{y}) \end{aligned}$$

avec  $T \in ]0, +\infty[$ . Un **problème de Cauchy** revient à chercher une fonction  $\mathbf{y} : [t^0, t^0 + T] \longrightarrow \mathbb{R}^d$  continue et dérivable, telle que

$$\mathbf{y}'(t) = \mathbf{f}(t, \mathbf{y}(t)), \quad \forall t \in [t^0, t^0 + T] \tag{1}$$

$$\mathbf{y}(t^0) = \mathbf{y}^{[0]} \in \mathbb{R}^d. \tag{2}$$

Différents schémas sont rappelés permettant de résoudre numériquement un problème de Cauchy. On note  $t^n$ ,  $n \in \llbracket 0, N \rrbracket$ , une discrétisation régulière de  $[t^0, t^0 + T]$ ,  $\mathbf{y}^{[n]} \approx \mathbf{y}(t^n)$  et  $\mathbf{f}^{[n]} = \mathbf{f}(t^n, \mathbf{y}^{[n]})$

- Le schéma de Runge-Kutta d'ordre 4

$$\begin{aligned} \mathbf{k}_1^{[n]} &= \mathbf{f}(t^n, \mathbf{y}^{[n]}) \\ \mathbf{k}_2^{[n]} &= \mathbf{f}\left(t^n + \frac{h}{2}, \mathbf{y}^{[n]} + \frac{h}{2}\mathbf{k}_1^{[n]}\right) \\ \mathbf{k}_3^{[n]} &= \mathbf{f}\left(t^n + \frac{h}{2}, \mathbf{y}^{[n]} + \frac{h}{2}\mathbf{k}_2^{[n]}\right) \\ \mathbf{k}_4^{[n]} &= \mathbf{f}\left(t^n + h, \mathbf{y}^{[n]} + h\mathbf{k}_3^{[n]}\right) \\ \mathbf{y}^{[n+1]} &= \mathbf{y}^{[n]} + \frac{h}{6}(\mathbf{k}_1^{[n]} + 2\mathbf{k}_2^{[n]} + 2\mathbf{k}_3^{[n]} + \mathbf{k}_4^{[n]}). \end{aligned} \tag{3}$$

- Trois schémas d'Adams-Bashforth :

$$\mathbf{y}^{[n+1]} = \mathbf{y}^{[n]} + \frac{h}{2} \left( 3\mathbf{f}^{[n]} - \mathbf{f}^{[n-1]} \right). \tag{4}$$

$$\mathbf{y}^{[n+1]} = \mathbf{y}^{[n]} + \frac{h}{12} \left( 23\mathbf{f}^{[n]} - 16\mathbf{f}^{[n-1]} + 5\mathbf{f}^{[n-2]} \right). \tag{5}$$

$$\mathbf{y}^{[n+1]} = \mathbf{y}^{[n]} + \frac{h}{24} \left( 55\mathbf{f}^{[n]} - 59\mathbf{f}^{[n-1]} + 37\mathbf{f}^{[n-2]} - 9\mathbf{f}^{[n-3]} \right). \tag{6}$$

Ces schémas sont **explicites** et leur ordre correspond au nombre de pas.

- Trois schémas d'Adams-Moulton :

$$\mathbf{y}^{[n+1]} = \mathbf{y}^{[n]} + \frac{h}{2} \left( \mathbf{f}^{[n+1]} + \mathbf{f}^{[n]} \right). \tag{7}$$

$$\mathbf{y}^{[n+1]} = \mathbf{y}^{[n]} + \frac{h}{12} \left( 5\mathbf{f}^{[n+1]} + 8\mathbf{f}^{[n]} - \mathbf{f}^{[n-1]} \right). \tag{8}$$

$$\mathbf{y}^{[n+1]} = \mathbf{y}^{[n]} + \frac{h}{24} \left( 9\mathbf{f}^{[n+1]} + 19\mathbf{f}^{[n]} - 5\mathbf{f}^{[n-1]} + \mathbf{f}^{[n-2]} \right). \tag{9}$$

Ces schémas sont **implicites** et leur ordre correspond au nombre de pas plus un.

**Q. 1 (Matlab)** Ecrire la fonction **Matlab** `[t, Y]=redRK4(f,a,b,y0,N)` permettant de résoudre numériquement le problème de Cauchy (1)-(2) par le schéma de Runge et Kutta d'ordre 4. Les paramètres d'entrée sont les données du problème de Cauchy avec  $\mathbf{f} = \mathbf{f}$ ,  $\mathbf{a} = t^0$ ,  $\mathbf{b} = t^0 + T$ ,  $\mathbf{y0} = \mathbf{y}^{[0]} \in \mathbb{R}^d$  et  $\mathbf{N}$  étant le nombre de pas de discrétisation.

**Application :** Considérons le système mécanique de trois masses  $m_1$ ,  $m_2$  et  $m_3$  attachées entre elles horizontalement par des ressorts de raideur  $k_1$ ,  $k_2$ ,  $k_3$  et  $k_4$ . Les positions au cours du temps des masses par rapport à leurs positions d'équilibre sont données par les fonctions  $x_1$ ,  $x_2$  et  $x_3$ .

Le système d'équations de mouvement du système s'écrit sous la forme :

$$\begin{cases} m_1 \ddot{x}_1(t) + (k_1 + k_2)x_1(t) - k_2x_2(t) &= 0 & (10a) \\ m_2 \ddot{x}_2(t) + (k_2 + k_3)x_2(t) - k_2x_1(t) - k_3x_3(t) &= 0 & (10b) \\ m_3 \ddot{x}_3(t) + (k_3 + k_4)x_3(t) - k_3x_2(t) &= 0 & (10c) \end{cases}$$

On veut résoudre ce système d'E.D.O. avec pour données initiales  $x_1(0) = 1$ ,  $\dot{x}_1(0) = 0$ ,  $x_2(0) = -1$ ,  $\dot{x}_2(0) = 1/2$ ,  $x_3(0) = 1/3$  et  $\dot{x}_3(0) = -1/2$ . Le temps final  $T$  sera égal à 20.

On va utiliser la méthode de Runge-Kutta d'ordre 4 pour l'étude numérique des différents types de mouvements possibles, suivant les conditions de l'expérience (choix des valeurs de  $k_1, \dots, k_4$  et de  $m_1, m_2, m_3$ ) et les conditions initiales imposées. On souhaite ensuite tracer les trois courbes discrètes correspondants aux positions des trois masses au cours du temps dans une figure et leurs vitesses dans une autre.

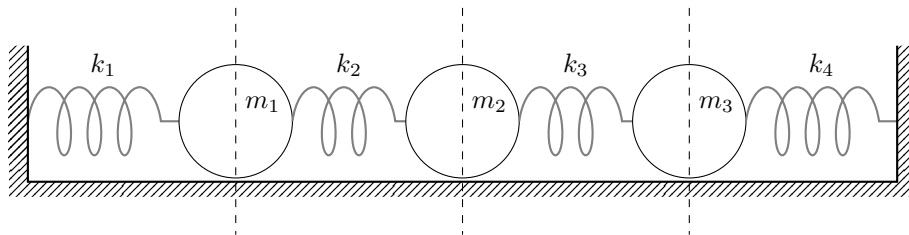


FIGURE 1 – Positions d'équilibre

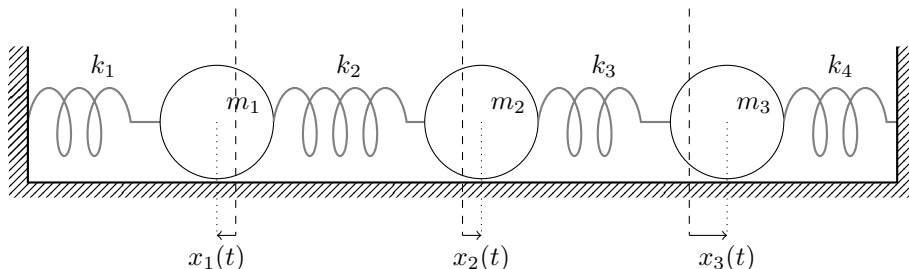


FIGURE 2 – En mouvement

- Q. 2**
1. Ecrire de manière détaillée le problème de Cauchy associé à ce système mécanique.
  2. Quelles sont les données du problème de Cauchy obtenu ? (avec leur type détaillé : entier, réel, complexe, vecteur, matrice, fonction, ...)
  3. Quelles sont les inconnues du problème de Cauchy obtenu ? (avec leur type détaillé : entier, réel, complexe, vecteur, matrice, fonction, ...)

**Q. 3 (Matlab)** Ecrire un programme **Matlab** permettant de résoudre numériquement ce système mécanique par la méthode de Runge-Kutta d'ordre 4, et de représenter dans une figure les trois courbes discrètes correspondants aux positions des trois masses au cours du temps et dans une autre figure leurs vitesses.

Un **schéma prédicteur-correcteur** procède à chaque itération en deux temps : il fournit une valeur approchée de la solution au  $n^{\text{ième}}$  pas en utilisant un schéma explicite (prédiction), puis il calcule la valeur correspondante de  $\mathbf{f}$  et enfin, il substitue cette valeur dans un schéma implicite (correction).

$\mathbf{y}^{[0]}$  donné  
 pour  $n$  variant de 0 à  $N - 1$  faire  
 Calculer  $\bar{\mathbf{y}}$  une valeur approchée de  $\mathbf{y}^{(t^{n+1})}$  par un schéma explicite à partir de  $\mathbf{y}^{[n]}$ ,  $\mathbf{y}^{[n-1]}$ , ...  
 Evaluer  $\mathbf{f} = \mathbf{f}(t^{n+1}, \bar{\mathbf{y}})$   
 Calculer  $\mathbf{y}^{[n+1]}$  à l'aide d'un schéma implicite en remplaçant  $\mathbf{f}^{[n+1]}$  par  $\bar{\mathbf{f}}$   
 finpour

On suppose écrite la fonction `[t, Y]=redRK4(f,a,b,yo,N)` correspondant au schéma de Runge et Kutta d'ordre 4 précédent. Les paramètres d'entrée  $\mathbf{f}$ ,  $\mathbf{a}$ ,  $\mathbf{b}$ ,  $\mathbf{yo}$  correspondent respectivement aux  $\mathbf{f}$ ,  $t^0$ ,  $t^0 + T$ ,  $\mathbf{y}^{[0]}$  du problème de Cauchy (1)-(2). Enfin,  $\mathbf{Y}$  est le tableau contenant les  $\mathbf{y}^{[n]}$ ,  $n \in \{0, \dots, N\}$  et  $\mathbf{t}$  est le tableau contenant les  $N+1$  réels  $t^n$ ,  $n \in \{0, \dots, N\}$

- Q. 4 (Matlab)**
1. Ecrire la fonction `[t, Y]=redPC4(f,a,b,yo,N)` correspondant à un schéma **prédicteur-correcteur** utilisant les schémas d'Adams-Bashforth et d'Adams-Moulton d'ordre 4. Ses paramètres d'entrée et de sortie sont identiques à ceux de la fonction `redRK4`.
  2. Ecrire un programme permettant de représenter graphiquement l'ordre du schéma de prédiction-corréction précédent.

**EXERCICE 3 (7.75 POINTS)**

On souhaite résoudre numériquement l'E.D.P. suivante

$$\frac{\partial u}{\partial t}(t, x) - \kappa \frac{\partial^2 u}{\partial x^2}(t, x) + c(x)u(t, x) = f(t, x), \quad \forall (t, x) \in ]t_0; t_0 + T] \times ]a; b[, \quad (1)$$

$$u(t_0, x) = u_0(x), \quad \forall x \in [a; b], \quad (2)$$

$$u(t, a) = u_a(t), \quad \forall t \in [t_0; t_0 + T], \quad (3)$$

$$\frac{\partial u}{\partial x}(t, b) + 2u(t, b) = v_b(t), \quad \forall t \in ]t_0; t_0 + T]. \quad (4)$$

avec  $\kappa > 0$ ,  $t_0 \in \mathbb{R}$ ,  $T > 0$ ,  $(a, b) \in \mathbb{R}^2$ ,  $a < b$ ,  $c : [a, b] \rightarrow \mathbb{R}^+$ .

On note  $t^n$ ,  $n \in \llbracket 0, N_t \rrbracket$  et  $x_i$ ,  $i \in \llbracket 0, N_x \rrbracket$  les discrétisations régulières des intervalles  $[t_0; t_0 + T]$  et  $[a; b]$  avec  $N_t$  pas de discrétisation en temps et  $N_x$  pas de discrétisation en espace. On souhaite résoudre l'E.D.P. à l'aide des schémas numériques

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} - \kappa \frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}}{\Delta x^2} + c_i u_i = f_i^{n+1}. \quad (5)$$

$$u_{N_x-2}^{n+1} - 4u_{N_x-1}^{n+1} + (3 + 4\Delta x)u_{N_x}^{n+1} = 2\Delta x v_b(t^{n+1}). \quad (6)$$

- Q. 1**
1. Expliquer précisément comment le schéma (5) a été obtenu à partir de (1) et expliciter les valeurs  $u_i^{n+1}$ ,  $f_i^{n+1}$ ,  $c_i$ ,  $\Delta t$  et  $\Delta x$ .
  2. Expliquer précisément comment le schéma (6) a été obtenu à partir de (4).
  3. Donner une discrétisation (détaillée) du problème (1) à (4) en utilisant les schémas (5) et (6).

On note  $\mathbf{U}^n$  les vecteurs de dimension  $N_x + 1$ , de composantes  $\mathbf{U}_i^n = u_{i-1}^n$ ,  $\forall i \in \llbracket 1, N_x + 1 \rrbracket$ .

- Q. 2**
1. Comment initialiser le vecteur  $\mathbf{U}^0$  ?
  2. En supposant le vecteur  $\mathbf{U}^n$  déjà calculé, montrer que le vecteur  $\mathbf{U}^{n+1}$  est solution du système linéaire

$$\mathbb{A}\mathbf{U}^{n+1} = \mathbf{b}^n \quad (7)$$

en explicitant la matrice  $\mathbb{A}$  et le vecteur  $\mathbf{b}^n$  (préciser les dimensions).

- Q. 3 (Matlab)**
1. Ecrire la fonction `ASSEMBLEMAT1D` retournant la matrice **creuse**  $\mathbb{M} \in \mathcal{M}_d(\mathbb{R})$  définie par

$$\mathbb{M} = \begin{pmatrix} \nu_1 & \nu_2 & \nu_3 & 0 & \dots & \dots & 0 \\ \beta_1 & \alpha_1 & \beta_1 & 0 & \dots & \dots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & \dots & 0 & \beta_{d-2} & \alpha_{d-2} & \beta_{d-2} \\ 0 & \dots & \dots & 0 & \mu_1 & \mu_2 & \mu_3 \end{pmatrix} \quad (8)$$

où  $\alpha \in \mathbb{R}^{d-2}$ ,  $\beta \in \mathbb{R}^{d-2}$ ,  $\mu \in \mathbb{R}^3$  et  $\nu \in \mathbb{R}^3$  sont donnés. Cette fonction ne devra pas utiliser de boucles `for` ou `while`.

2. On se donne  $a = -2$ ,  $b = 2$ ,  $T = 10$ ,  $f(t, x) = x^2 \cos(t)$ ,  $u_0(x) = 10$ ,  $c(x) = 2 + \sin(x)$ ,  $\kappa = 1$ ,  $v_b(t) = \sin(t)$ ,

$$u_a(t) = \begin{cases} 10 + 50t, & \text{si } t \leq 2, \\ 110, & \text{si } t \in [2; 8[, \\ 110 - 50(t - 8), & \text{si } t \in [8; 10]. \end{cases}$$

Ecrire un programme Matlab complet permettant de résoudre le problème (1) à (4) en utilisant les schémas (5) et (6).

## EXERCICE 4 (3.0 POINTS)

Soient  $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$  et  $\mathbb{B} \in \mathcal{M}_{p,q}(\mathbb{R})$ . Le produit tensoriel de Kronecker de  $\mathbb{A}$  par  $\mathbb{B}$ , noté  $\mathbb{A} \otimes \mathbb{B}$ , est la matrice de  $\mathcal{M}_{mp,nq}(\mathbb{R})$  définie avec des blocs de dimension  $p \times q$  par

$$\mathbb{A} \otimes \mathbb{B} = \begin{pmatrix} \mathbb{A}_{1,1}\mathbb{B} & \cdots & \mathbb{A}_{1,n}\mathbb{B} \\ \vdots & \ddots & \vdots \\ \mathbb{A}_{m,1}\mathbb{B} & \cdots & \mathbb{A}_{m,n}\mathbb{B} \end{pmatrix} \quad (1)$$

Le produit de Kronecker est bilinéaire et associatif : Si les dimensions des matrices  $\mathbb{A}$ ,  $\mathbb{B}$  et  $\mathbb{C}$  sont compatibles on a  $\forall \lambda \in \mathbb{K}$

$$\begin{aligned} \mathbb{A} \otimes (\mathbb{B} + \lambda \cdot \mathbb{C}) &= (\mathbb{A} \otimes \mathbb{B}) + \lambda(\mathbb{A} \otimes \mathbb{C}) \\ (\mathbb{A} + \lambda \cdot \mathbb{B}) \otimes \mathbb{C} &= (\mathbb{A} \otimes \mathbb{C}) + \lambda(\mathbb{B} \otimes \mathbb{C}) \\ \mathbb{A} \otimes (\mathbb{B} \otimes \mathbb{C}) &= (\mathbb{A} \otimes \mathbb{B}) \otimes \mathbb{C} \end{aligned}$$

Par contre, il n'est pas commutatif.

Nous allons écrire une fonction permettant d'effectuer ce produit. Mais auparavant, quelques petits rappels (ou non) des possibilités offertes par le langage Matlab.

Il est facile sous Matlab/Octave de modifier une matrice (creuse ou non). Par exemple,

- `A(I,J)=M`

Les tableaux d'indices `I` et `J` sont de dimension respective  $n$  et  $m$ . La matrice `M` est de dimension  $n$ -par- $m$ . La matrice `A` est modifiée de telle sorte que

$$A(I(i),J(j))=M(i,j), \quad \forall i \in \llbracket 1, n \rrbracket, \quad \forall j \in \llbracket 1, m \rrbracket$$

- `A(I,J)=A(I,J)+M`

Même chose en sommant.

- `A(i,:)=a`

Ici `a` est un scalaire. Tous les éléments de la ligne `i` de la matrice `A` valent `a`.

- `A(:,j)=a`

Ici `a` est un scalaire. Tous les éléments de la colonne `j` de la matrice `A` valent `a`.

- ...

Sous Matlab/Octave, il est possible de récupérer tous les éléments non nuls d'une matrice creuse `A` ainsi que leurs indices de ligne et de colonne à l'aide de la commande

$$[I,J,K]=\text{find}(A);$$

Ici les trois tableaux `I`, `J` et `K` ont même longueur et on a `A(I(k),J(k)) == K(k)` pour tout `k` inférieur ou égal à `length(K)`.

**Q. 1 (Matlab)** *Ecrire la fonction `spMatKron` permettant à partir deux matrices creuses carrées  $\mathbb{A} \in \mathcal{M}_n(\mathbb{R})$  et  $\mathbb{B} \in \mathcal{M}_m(\mathbb{R})$  de retourner la matrice creuse  $\mathbb{C} = \mathbb{A} \otimes \mathbb{B}$ . Pour celà, on initialisera la matrice retournée `C` à l'aide de la commande `C=sparse(n*m,n*m)` puis on utilisera uniquement une boucle sur les éléments non-nuls de la matrice `A` pour «remplir» la matrice `C`.*

**EXERCICE 5 (7.25 POINTS)**

Soient  $\Omega = ]a, b[ \times ]c, d[ \subset \mathbb{R}^2$  et  $\Gamma = \partial\Omega$  la frontière du domaine  $\Omega$ . On note  $\Gamma_N, \Gamma_S, \Gamma_O$  et  $\Gamma_E$  respectivement les frontières nord, sud, ouest et est. on a

$$\Gamma = \Gamma_N \cup \Gamma_S \cup \Gamma_O \cup \Gamma_E.$$

On note  $(x_i)_{i=0}^{N_x}$  et  $(y_j)_{j=0}^{N_y}$  les discrétisations régulières, respectivement, des intervalles  $[a, b]$  et  $[c, d]$  définies par

$$x_i = a + ih_x, \quad \forall i \in \llbracket 0, N_x \rrbracket \quad \text{et} \quad y_j = c + jh_y, \quad \forall j \in \llbracket 0, N_y \rrbracket \quad (1)$$

avec  $h_x = (b - a)/N_x$  et  $h_y = (d - c)/N_y$ . On note aussi

$$n_x = N_x + 1, \quad n_y = N_y + 1 \quad \text{et} \quad N = n_x \times n_y \quad (2)$$

Soient  $f : \Omega \rightarrow \mathbb{R}$  et  $g : \Gamma \rightarrow \mathbb{R}$  deux fonctions données. On veut résoudre le problème suivant

$$-\Delta u = f, \quad \text{dans } \Omega \quad (3)$$

$$u = g, \quad \text{sur } \Gamma \quad (4)$$

en utilisant la discrétisation d'ordre 2 suivante :

$$-\frac{U_{i+1,j} - 2U_{i,j} + U_{i-1,j}}{h_x^2} - \frac{U_{i,j+1} - 2U_{i,j} + U_{i,j-1}}{h_y^2} = f(x_i, y_j), \quad \forall (i, j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket, \quad (5)$$

$$U_{0,j} = g(a, y_j), \quad \forall j \in \llbracket 0, N_y \rrbracket, \quad (6)$$

$$U_{N_x,j} = g(b, y_j), \quad \forall j \in \llbracket 0, N_y \rrbracket, \quad (7)$$

$$U_{i,0} = g(x_i, c), \quad \forall i \in \llbracket 0, N_x \rrbracket, \quad (8)$$

$$U_{i,N_y} = g(x_i, d), \quad \forall i \in \llbracket 0, N_x \rrbracket. \quad (9)$$

avec  $U_{i,j} \approx u(x_i, y_j)$ .

Pour tout  $j \in \llbracket 0, N_y \rrbracket$ , on note  $U_{:,j}$  le vecteur de  $\mathbb{R}^{n_x}$  défini par

$$U_{:,j} = \begin{pmatrix} U_{0,j} \\ \vdots \\ U_{N_x,j} \end{pmatrix}.$$

On note  $\mathbf{V} \in \mathbb{R}^N$  le vecteur bloc

$$\mathbf{V} = \begin{pmatrix} U_{:,0} \\ \hline U_{:,1} \\ \hline \vdots \\ \hline U_{:,N_y} \end{pmatrix}$$

**Q. 1** Expliquez la bijection  $\mathcal{F} : \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket \rightarrow \llbracket 1, N \rrbracket$  telle que

$$\forall (i, j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket, \quad V_k = U_{i,j}, \quad \text{avec } k = \mathcal{F}(i, j).$$

Dans le cas de la numérotation en  $(i, j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket$  on parlera de **numérotation 2D** et pour la numérotation en  $k \in \llbracket 1, N \rrbracket$  on parlera de **numérotation globale**.

**Q. 2 (Matlab)** Ecrire la fonction `k=bijF(i,j,nx)` correspondant à la bijection  $\mathcal{F}$  (**numérotation 2D** vers **numérotation globale**).

Chacune des équations du problème discret (5)-(9) correspond à une discrétisation en un point  $(x_i, y_j)$ . Nous choisissons d'écrire ces équations en utilisant la même numérotation que lors de la construction du vecteur  $\mathbf{V}$  : l'équation écrite au point  $(x_i, y_j)$  sera écrite en ligne  $k = \mathcal{F}(i, j)$  du système.

**Q. 3** Etablir que le problème discret (5)-(9) peut s'écrire sous la forme du système linéaire bloc

$$\begin{pmatrix} \mathbb{E} & \mathbb{O} & \cdots & \cdots & \cdots & \mathbb{O} & \mathbb{O} \\ \mathbb{M} & \mathbb{D} & \mathbb{M} & \mathbb{O} & \cdots & \mathbb{O} & \mathbb{O} \\ \mathbb{O} & \mathbb{M} & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \mathbb{O} & \ddots & \ddots & \ddots & \mathbb{O} & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \mathbb{M} & \mathbb{O} \\ \mathbb{O} & \mathbb{O} & \cdots & \mathbb{O} & \mathbb{M} & \mathbb{D} & \mathbb{M} \\ \mathbb{O} & \mathbb{O} & \cdots & \cdots & \cdots & \mathbb{O} & \mathbb{E} \end{pmatrix} \mathbf{V} = \begin{pmatrix} B_{:,0} \\ \hline B_{:,1} \\ \hline \vdots \\ \hline B_{:,N_y} \end{pmatrix} \quad (10)$$

où chaque bloc de la matrice est une matrice  $(N_x + 1)$  par  $(N_x + 1)$ . La matrice  $\mathbb{O} \in \mathcal{M}_{N_x+1}(\mathbb{R})$  est la matrice nulle. Les matrices creuses  $\mathbb{D}$ ,  $\mathbb{M}$  et  $\mathbb{E}$  ainsi que les vecteurs  $B_{:,j} \in \mathbb{R}^{N_x+1}$ , pour tout  $j \in \llbracket 0, N_y \rrbracket$ , devront être donnés explicitement.

On note  $\mathbb{I}_n$  la matrice identité de  $\mathcal{M}_n(\mathbb{R})$  et  $\mathbb{J}_n$  la matrice de  $\mathcal{M}_n(\mathbb{R})$  définie par

$$\mathbb{J}_n = \begin{pmatrix} 0 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 1 & 0 \\ 0 & \cdots & \cdots & 0 & 0 \end{pmatrix} \in \mathcal{M}_n(\mathbb{R}).$$

Nous allons maintenant générer/assembler la matrice du système (10) **sans tenir compte des conditions aux limites** : on note  $-\mathbb{A}_{xy}$  la matrice ainsi obtenue.

C'est une matrice bloc de  $\mathcal{M}_N(\mathbb{R})$  avec  $n_y$  lignes bloc composées de blocs carrés de dimension  $n_x$  qui s'écrit sous la forme :

$$\mathbb{A}_{xy} = \begin{pmatrix} \mathbb{O} & \mathbb{O} & \cdots & \cdots & \cdots & \mathbb{O} & \mathbb{O} \\ \mathbb{O} & \mathbb{A}_x & \mathbb{O} & \mathbb{O} & \cdots & \mathbb{O} & \mathbb{O} \\ \mathbb{O} & \mathbb{O} & \mathbb{A}_x & \ddots & \ddots & \vdots & \vdots \\ \vdots & \mathbb{O} & \ddots & \ddots & \ddots & \mathbb{O} & \vdots \\ \vdots & \vdots & \ddots & \ddots & \mathbb{A}_x & \mathbb{O} & \mathbb{O} \\ \mathbb{O} & \mathbb{O} & \cdots & \mathbb{O} & \mathbb{O} & \mathbb{A}_x & \mathbb{O} \\ \mathbb{O} & \mathbb{O} & \cdots & \cdots & \cdots & \mathbb{O} & \mathbb{O} \end{pmatrix} + \begin{pmatrix} \mathbb{O} & \mathbb{O} & \cdots & \cdots & \cdots & \mathbb{O} & \mathbb{O} \\ \mathbb{S}_y & \mathbb{T}_y & \mathbb{S}_y & \mathbb{O} & \cdots & \mathbb{O} & \mathbb{O} \\ \mathbb{O} & \mathbb{S}_y & \mathbb{T}_y & \ddots & \ddots & \vdots & \vdots \\ \vdots & \mathbb{O} & \ddots & \ddots & \ddots & \mathbb{O} & \vdots \\ \vdots & \vdots & \ddots & \ddots & \mathbb{T}_y & \mathbb{S}_y & \mathbb{O} \\ \mathbb{O} & \mathbb{O} & \cdots & \mathbb{O} & \mathbb{S}_y & \mathbb{T}_y & \mathbb{S}_y \\ \mathbb{O} & \mathbb{O} & \cdots & \cdots & \cdots & \mathbb{O} & \mathbb{O} \end{pmatrix} \quad (11)$$

où

$$\mathbb{S}_y = \frac{1}{h_y^2} \mathbb{J}_{n_x}, \quad \mathbb{T}_y = -\frac{2}{h_y^2} \mathbb{J}_{n_x} \quad \text{et} \quad \mathbb{A}_x = \frac{1}{h_x^2} \mathbb{L}$$

avec

$$\mathbb{L} = \begin{pmatrix} 0 & 0 & 0 & \cdots & \cdots & 0 \\ 1 & -2 & 1 & \ddots & & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & 1 & -2 & 1 \\ 0 & \cdots & \cdots & 0 & 0 & 0 \end{pmatrix} \in \mathcal{M}_{n_x}(\mathbb{R})$$

On peut noter que les matrices  $\mathbb{A}_x$ ,  $\mathbb{T}_y$  et  $\mathbb{S}_y$  sont des matrices de  $\mathcal{M}_{n_x}(\mathbb{R})$ .

En notant  $\mathbb{A}_y \in \mathcal{M}_{n_y}(\mathbb{R})$  la matrice définie par  $\mathbb{A}_y = \frac{1}{h_y^2} \mathbb{L}$ , on déduit de (11)

$$\mathbb{A}_{xy} = \mathbb{I}_{n_y} \otimes \mathbb{A}_x + \mathbb{A}_y \otimes \mathbb{I}_{n_x}. \quad (12)$$

- Q. 4 (Matlab)**
1. Sans utiliser de boucles, écrire la fonction `Lap1DAssembling` retournant la matrice creuse  $\mathbb{L}$ .
  2. Écrire la fonction `Lap2DAssembling` retournant la matrice bloc creuse  $\mathbb{A}_{xy}$  en utilisant (12).
  3. Proposer un programme permettant de tester/valider la matrice ainsi obtenue en utilisant le laplacien d'une fonction de  $\mathbb{R}^2$  à valeurs dans  $\mathbb{R}$ .