

Différences finies 2D avec Matlab/Octave

Prémises au TP 4

Sup'Galilée, Ingénieurs MACS, 2ème année

François Cuvelier

Laboratoire d'Analyse Géométrie et Applications
Institut Galilée
Université Paris XIII.

20/11/2025

Prérequis : avoir étudié le pdf sur les matrices creuses.

1 Matrices et tableaux

- Stockage
- Numérotation 1D
- Sous-matrices

2 Représentations graphiques

3 Différences finies 2D

- Schéma différences finies
 - Système linéaire
 - Numérotations
 - Ecriture bloc
- Assemblage
 - Méthode bloc
 - Méthode bijection

Soit $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$ et $N = mn$.

$$\mathbb{A} = \left(\begin{array}{c|c|c} \mathbb{A}_{:,1} & \dots & \mathbb{A}_{:,n} \end{array} \right) \text{ avec } \mathbb{A}_{:,j} \in \mathbb{R}^n, j^{\text{ème}} \text{ colonne de } \mathbb{A}.$$

↪ stockage de tous tableaux/matrices pleines de manière contigüe en mémoire.

$$\mathbf{a} = \begin{pmatrix} \mathbb{A}_{:,1} \\ \text{---} \\ \vdots \\ \text{---} \\ \mathbb{A}_{:,m} \end{pmatrix} \in \mathbb{R}^N$$

- Numérotation 2D (classique), *2D-indices*, en $(i, j) \in \llbracket 1, m \rrbracket \times \llbracket 1, n \rrbracket$,
- Numérotation 1D, *linear indices*, en $k \in \llbracket 1, N \rrbracket$.

- ① `m=5;n=4;A=rand(m,n); a=A(:);` ↪ tableau 2D `A` (*m-by-n*) vers tableau 1D `a` (*(mn)-by-1*)
- ② `m=5;n=4;b=rand(m*n,1); B=reshape(b,[m,n]);` ↪ tableau 1D `b` vers tableau 2D `B`

1 Matrices et tableaux

- Stockage
- **Numérotation 1D**
- Sous-matrices

2 Représentations graphiques

3 Différences finies 2D

- Schéma différences finies
 - Système linéaire
 - Numérotations
 - Ecriture bloc
- Assemblage
 - Méthode bloc
 - Méthode bijection

On note $\mathcal{M} : \llbracket 1, m \rrbracket \times \llbracket 1, n \rrbracket \longrightarrow \llbracket 1, N \rrbracket$ la bijection de la numérotation 2D à la numérotation 1D.

$$K = \mathcal{M}(I, J) \quad \hookrightarrow \text{K=sub2ind}([m,n],I,J)$$

La fonction réciproque est

$$(I, J) = \mathcal{M}^{-1}(K) \quad \hookrightarrow [I, J] = \text{ind2sub}([m,n],K)$$

Ici I , J et K sont des tableaux de **même dimensions**.

Soient $A \in \mathcal{M}_{m,n}(\mathbb{R})$, $I \subset \llbracket 1, m \rrbracket$, $J \subset \llbracket 1, n \rrbracket$, $d = \#I = \#J$.

$$\mathbf{b} \in \mathbb{R}^d \text{ t.q. } \mathbf{b}(k) = A(I(k), J(k)), \quad \forall k \in \llbracket 1, d \rrbracket.$$

$\text{K=sub2ind}(\text{size}(A),I,J)$; $\mathbf{b}=A(K)$;

Que fait la commande $A(K)=0$; ?

```
% echo on
m=4;n=3;a=1:m*n;
A=reshape(a,m,n)
A=A'
% Comment recuperer A(1,3), A(2,1) et A(3,2) ?
I=[1,2,3];J=[3,1,2]; % A(1,3)=A(I(1),J(1)), ...
K=sub2ind([n,m],I,J); % ou K=sub2ind(size(A),I,J);
b=A(K) % => b(1)=A(1,3), b(2)=A(2,1) et b(3)=A(3,2)
A(K)=0 % => A(1,3)=0, A(2,1)=0 et A(3,2)=0
```

Soit B une matrice n -by- n existante. On veut la modifier pour avoir $B(i,i)=-1$ pour i impair.

Comment faire sans boucle ? (voir exempleB.m)

Listing – exempleB.m avec boucle

```
n=4;b=1:n*n;
B=reshape(b,m,n)
for i=1:2:n, B(i,i)=-1;end
```

1 Matrices et tableaux

- Stockage
- Numérotation 1D
- Sous-matrices

2 Représentations graphiques

3 Différences finies 2D

- Schéma différences finies
 - Système linéaire
 - Numérotations
 - Ecriture bloc
- Assemblage
 - Méthode bloc
 - Méthode bijection

Soient $\mathbb{A} \in \mathcal{M}_{m,n}(\mathbb{R})$, $I \subset \llbracket 1, m \rrbracket$, $d_I = \#I$, $J \subset \llbracket 1, n \rrbracket$, $d_J = \#J$.

On note $\mathbb{M} \in \mathcal{M}_{d_I, d_J}(\mathbb{R})$:

$$\mathbb{M}_{i,j} = \mathbb{A}(I(i), J(j)), \quad \forall (i,j) \in \llbracket 1, d_I \rrbracket \times \llbracket 1, d_J \rrbracket \quad \hookrightarrow \quad \mathbf{M} = \mathbf{A}(I, J);$$

A tableau 2D m-by-n,

- $\mathbf{A}(I, J) = \mathbf{M}$; ou $\mathbf{A}(I, J) = \mathbf{A}(I, J) + \mathbf{M}$; $\hookrightarrow \mathbf{M} \in ?$
- $\mathbf{A}(I, :) = \mathbf{M}$; ou $\mathbf{A}(I, :) = \mathbf{A}(I, :) + \mathbf{M}$; $\hookrightarrow \mathbf{M} \in ?$
- $\mathbf{A}(:, J) = \mathbf{M}$; ou $\mathbf{A}(:, J) = \mathbf{A}(:, J) + \mathbf{M}$; $\hookrightarrow \mathbf{M} \in ?$

Listing – exempleC.m

```
1 m=4;n=3;a=1:m*n;  
2 A=reshape(a,m,n)  
3 I=[1,2,4];J=[2,3];  
4 M=A(I,J)  
5 A(I,J)=0  
6 A(I,J)=A(I,J)-M  
7 A(I,J)=A(I,J)+1  
8 A(I,J)=A(I,J)+[1,2]  
9 A(I,J)=A(I,J)+[1;2;3]
```

Exercice

Soit $\mathbb{A} \in \mathcal{M}_N(\mathbb{R})$ avec $N = d \times n$ une matrice tridiagonale blocs

$$\mathbb{A} = \begin{pmatrix} \mathbb{D} & \mathbb{F} & 0 & \dots & 0 & 0 \\ \mathbb{E} & \mathbb{D} & \mathbb{F} & 0 & \dots & 0 & 0 \\ 0 & \mathbb{E} & \ddots & \ddots & \ddots & \ddots & \ddots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & 0 & \ddots & \ddots & \ddots & \mathbb{F} & 0 \\ 0 & 0 & \dots & 0 & \mathbb{E} & \mathbb{D} & \mathbb{F} \\ 0 & 0 & \dots & 0 & 0 & \mathbb{E} & \mathbb{D} \end{pmatrix}, \text{ avec } \mathbb{D}, \mathbb{E} \text{ et } \mathbb{F} \text{ dans } \mathcal{M}_n(\mathbb{R})$$

Q. 1 Matrice pleine

On prend $n=2; d=7; N=d*n$; avec $E=\text{eye}(n); D=2*E; F=3*E$; Comment calculer $\mathbb{A}$? (voir `exemple03.m`)

Q. 2 Matrice creuse

On prend $n=2; d=7; N=d*n$; avec $E=\text{speye}(n); D=2*E; F=3*E$; Comment calculer $\mathbb{A}$? (voir `exemple03s.m`)

Q. 3

On se donne les 3 matrices creuses $\mathbb{D}$, $\mathbb{E}$ et $\mathbb{F}$ dans $\mathcal{M}_n(\mathbb{R})$

- ① Ecrire la fonction `Assemble` retournant la matrice creuse $\mathbb{A} \in \mathcal{M}_N(\mathbb{R})$. (voir `Assemble.m`)
- ② Ecrire la fonction `AssembleSpVec` retournant la matrice creuse $\mathbb{A} \in \mathcal{M}_N(\mathbb{R})$ sans utiliser de boucle. (voir `AssembleSpVec.m`)

```
>> exo01.bench_Assemble
n= 100, d= 20, full: 0.004 (s) - sparse: 0.005 (s) - ratio 0.85
n= 100, d= 40, full: 0.009 (s) - sparse: 0.003 (s) - ratio 3.00
n= 100, d= 60, full: 0.014 (s) - sparse: 0.009 (s) - ratio 1.57
n= 100, d= 80, full: 0.016 (s) - sparse: 0.009 (s) - ratio 1.91
n= 200, d= 20, full: 0.011 (s) - sparse: 0.004 (s) - ratio 2.63
n= 200, d= 40, full: 0.022 (s) - sparse: 0.006 (s) - ratio 3.78
n= 200, d= 60, full: 0.035 (s) - sparse: 0.011 (s) - ratio 3.09
n= 200, d= 80, full: 0.046 (s) - sparse: 0.017 (s) - ratio 2.78
n= 300, d= 20, full: 0.022 (s) - sparse: 0.010 (s) - ratio 2.27
n= 300, d= 40, full: 0.045 (s) - sparse: 0.009 (s) - ratio 4.92
n= 300, d= 60, full: 0.069 (s) - sparse: 0.018 (s) - ratio 3.75
n= 300, d= 80, full: 0.092 (s) - sparse: 0.028 (s) - ratio 3.34
n= 400, d= 20, full: 0.035 (s) - sparse: 0.017 (s) - ratio 2.07
n= 400, d= 40, full: 0.074 (s) - sparse: 0.012 (s) - ratio 6.06
n= 400, d= 60, full: 0.112 (s) - sparse: 0.024 (s) - ratio 4.59
n= 400, d= 80, full: 0.151 (s) - sparse: 0.039 (s) - ratio 3.82
>> % avec Matlab
```

Matlab : **Assemble** , full et sparse

```
>> exo01.bench_Assemble02
n= 1000, d= 200, sparse: 0.526 (s) - sparse vec.: 0.042 (s) - ratio 12.57
n= 1000, d= 400, sparse: 2.059 (s) - sparse vec.: 0.071 (s) - ratio 29.12
n= 1000, d= 600, sparse: 4.783 (s) - sparse vec.: 0.105 (s) - ratio 45.65
n= 2000, d= 200, sparse: 1.028 (s) - sparse vec.: 0.066 (s) - ratio 15.60
n= 2000, d= 400, sparse: 4.277 (s) - sparse vec.: 0.182 (s) - ratio 23.53
n= 2000, d= 600, sparse: 9.660 (s) - sparse vec.: 0.268 (s) - ratio 36.03
n= 3000, d= 200, sparse: 1.625 (s) - sparse vec.: 0.086 (s) - ratio 18.80
n= 3000, d= 400, sparse: 6.392 (s) - sparse vec.: 0.200 (s) - ratio 31.93
n= 3000, d= 600, sparse: 14.108 (s) - sparse vec.: 0.365 (s) - ratio 38.61
>> % Avec Matlab
```

Matlab : **Assemble** sparse et **AssembleSpVec**

```
octave:11> exo01.bench_Assemble
n= 100, d= 20, full: 0.006 (s) - sparse: 0.007 (s) - ratio 0.86
n= 100, d= 40, full: 0.055 (s) - sparse: 0.007 (s) - ratio 7.48
n= 100, d= 60, full: 0.118 (s) - sparse: 0.019 (s) - ratio 6.23
n= 100, d= 80, full: 0.205 (s) - sparse: 0.036 (s) - ratio 5.76
n= 200, d= 20, full: 0.053 (s) - sparse: 0.027 (s) - ratio 1.98
n= 200, d= 40, full: 0.204 (s) - sparse: 0.018 (s) - ratio 11.52
n= 200, d= 60, full: 0.456 (s) - sparse: 0.048 (s) - ratio 9.50
n= 200, d= 80, full: 0.803 (s) - sparse: 0.098 (s) - ratio 8.17
n= 300, d= 20, full: 0.118 (s) - sparse: 0.108 (s) - ratio 1.10
n= 300, d= 40, full: 0.467 (s) - sparse: 0.032 (s) - ratio 14.46
n= 300, d= 60, full: 1.015 (s) - sparse: 0.101 (s) - ratio 10.05
n= 300, d= 80, full: 1.815 (s) - sparse: 0.186 (s) - ratio 9.76
n= 400, d= 20, full: 0.211 (s) - sparse: 0.218 (s) - ratio 0.97
n= 400, d= 40, full: 0.823 (s) - sparse: 0.043 (s) - ratio 19.00
n= 400, d= 60, full: 1.827 (s) - sparse: 0.147 (s) - ratio 12.42
n= 400, d= 80, full: 3.449 (s) - sparse: 0.299 (s) - ratio 11.54
```

Octave : **Assemble** , full et sparse

```
octave:10> exo01.bench_Assemble02
n= 1000, d= 200, sparse: 1.610 (s) - sparse vec.: 0.033 (s) - ratio 49.51
n= 1000, d= 400, sparse: 7.064 (s) - sparse vec.: 0.086 (s) - ratio 82.00
n= 1000, d= 600, sparse: 16.001 (s) - sparse vec.: 0.101 (s) - ratio 158.64
n= 2000, d= 200, sparse: 3.511 (s) - sparse vec.: 0.054 (s) - ratio 65.38
n= 2000, d= 400, sparse: 14.533 (s) - sparse vec.: 0.150 (s) - ratio 97.07
n= 2000, d= 600, sparse: 32.395 (s) - sparse vec.: 0.218 (s) - ratio 148.64
n= 3000, d= 200, sparse: 5.388 (s) - sparse vec.: 0.101 (s) - ratio 53.58
n= 3000, d= 400, sparse: 21.596 (s) - sparse vec.: 0.204 (s) - ratio 105.87
n= 3000, d= 600, sparse: 48.261 (s) - sparse vec.: 0.305 (s) - ratio 158.42
octave:11>
```

Octave : **Assemble** sparse et **AssembleSpVec**

- 1 Matrices et tableaux
 - Stockage
 - Numérotation 1D
 - Sous-matrices
- 2 Représentations graphiques
- 3 Différences finies 2D
 - Schéma différences finies
 - Système linéaire
 - Numérotations
 - Ecriture bloc
 - Assemblage
 - Méthode bloc
 - Méthode bijection

Soit $f(x, y) = \cos(x) + \sin(y)$. Nous souhaitons représenter cette fonction sur $[a, b] \times [c, d]$

```
f=@(x,y) cos(x+y).*sin(x-y);
a=pi/2;b=pi/2;nx=51;
c=pi/3;d=7*pi/3;ny=113;
```

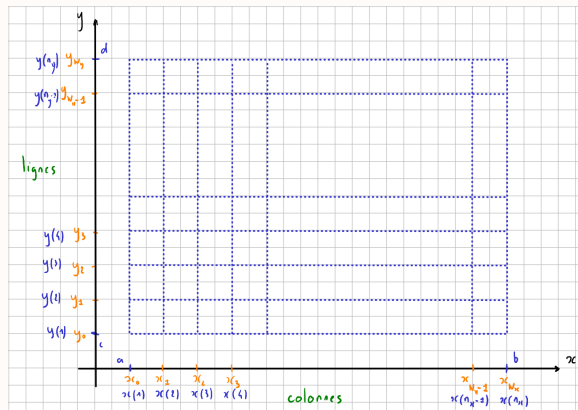
```
x=linspace(a,b,nx); % 1-by-nx
y=linspace(c,d,ny); % 1-by-ny
```

```
[X,Y]=meshgrid(x,y);
Z=f(X,Y); % Z,X et Y : ny-by-nx
```

```
figure(1)
surf(X,Y,Z)
xlabel('x');ylabel('y')
```

```
Z2=f(x,y'); % Z2 : ny-by-nx
figure(2)
surf(x,y,Z2)
xlabel('x');ylabel('y')
```

voir `exemple04.m` and co



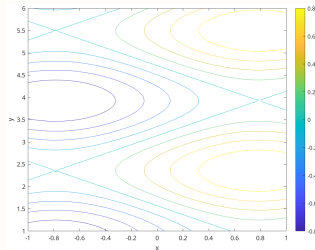
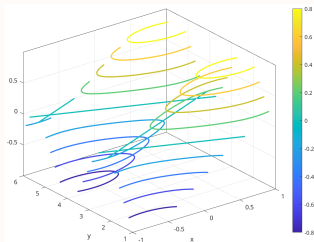
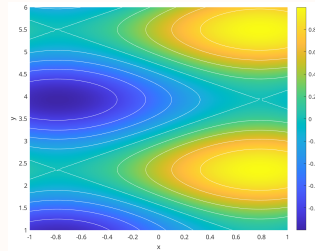
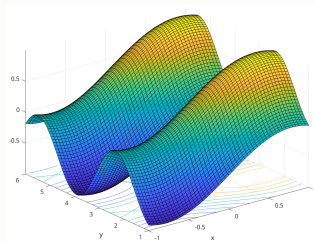
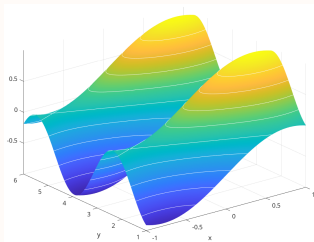


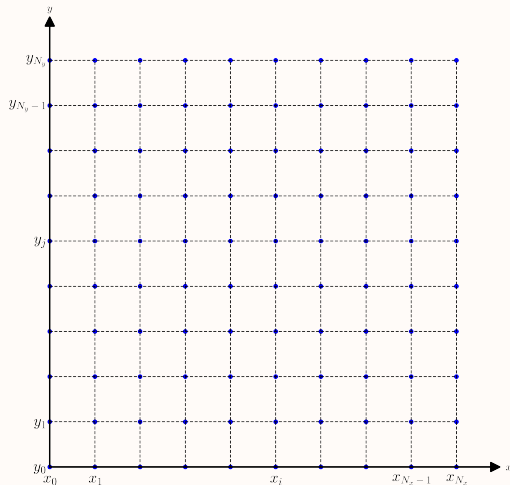
Figure – Utilisation successive des fonctions `surf+contour3`, `surfc`, `pcolor+contour`, `contour3` et `contour`

- 1 Matrices et tableaux
 - Stockage
 - Numérotation 1D
 - Sous-matrices
- 2 Représentations graphiques
- 3 Différences finies 2D
 - Schéma différences finies
 - Système linéaire
 - Numérotations
 - Ecriture bloc
 - Assemblage
 - Méthode bloc
 - Méthode bijection

$\Omega =]a, b[\times]c, d[\subset \mathbb{R}^2$ et $\Gamma = \Gamma_N \cup \Gamma_S \cup \Gamma_O \cup \Gamma_E = \partial\Omega$

$$-\Delta u = f \text{ dans } \Omega \quad (1)$$

$$\forall k \in \{N, S, O, E\}, \quad u = g_k, \text{ sur } \Gamma_k. \quad (2)$$



- (1) sur les points intérieurs $\Rightarrow$
 $\forall (i, j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket$

$$-\left(\frac{\partial^2 u}{\partial x^2}(x_i, y_j) + \frac{\partial^2 u}{\partial y^2}(x_i, y_j) \right) = f(x_i, y_j) \quad (3)$$

- (2) sur les points extérieurs $\Rightarrow$

$$u(a, y_j) = g_O(a, y_j), \quad \forall j \in \llbracket 0, N_y \rrbracket, \quad (4)$$

$$u(b, y_j) = g_E(b, y_j), \quad \forall j \in \llbracket 0, N_y \rrbracket, \quad (5)$$

$$u(x_i, c) = g_S(x_i, c), \quad \forall i \in \llbracket 0, N_x \rrbracket, \quad (6)$$

$$u(x_i, d) = g_N(x_i, d), \quad \forall i \in \llbracket 0, N_x \rrbracket. \quad (7)$$

Conditions de compatibilité :

$$g_O(a, c) = g_S(a, c), \quad g_S(b, c) = g_E(b, c)$$

$$g_E(b, d) = g_N(b, d), \quad g_N(a, d) = g_O(a, d).$$

Equation écrite en un point intérieur

On a

$$\frac{\partial^2 u}{\partial x^2}(x_i, y_j) = \frac{u(x_{i+1}, y_j) - 2u(x_i, y_j) + u(x_{i-1}, y_j))}{h_x^2} + \mathcal{O}(h_x^2), \quad \frac{\partial^2 u}{\partial y^2}(x_i, y_j) = \frac{u(x_i, y_{j+1}) - 2u(x_i, y_j) + u(x_i, y_{j-1}))}{h_y^2} + \mathcal{O}(h_y^2)$$

et (3) se réécrit au point intérieur (x_i, y_j) avec $(i, j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket$,

$$-\frac{u(x_{i+1}, y_j) - 2u(x_i, y_j) + u(x_{i-1}, y_j))}{h_x^2} - \frac{u(x_i, y_{j+1}) - 2u(x_i, y_j) + u(x_i, y_{j-1}))}{h_y^2} + \mathcal{O}(h_x^2) + \mathcal{O}(h_y^2) = f(x_i, y_j).$$

En "oubliant" les termes en $\mathcal{O}(h_x^2)$ et $\mathcal{O}(h_y^2)$ on obtient des équations approchées.

On note alors $U_{i,j}$ (en espérant $U_{i,j} \approx u(x_i, y_j)$!) les solutions de

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

En posant $\beta_x = \frac{-1}{h_x^2}$, $\beta_y = \frac{-1}{h_y^2}$, $\mu = \frac{2}{h_x^2} + \frac{2}{h_y^2}$.

$(E_{i,j})$ $\Leftrightarrow$ équation portée par le point (x_i, y_j)

$$\beta_x(U_{i+1,j} + U_{i-1,j}) + \beta_y(U_{i,j+1} + U_{i,j-1}) + \mu U_{i,j} = f_{i,j}. \quad (E_{i,j})$$

$$\begin{aligned}
\beta_x(U_{i+1,j} + U_{i-1,j}) + \beta_y(U_{i,j+1} + U_{i,j-1}) + \mu U_{i,j} &= f_{i,j}, & \forall (i,j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket, & (E_{i,j}) \\
U_{0,j} &= g_O(a, y_j), & \forall j \in \llbracket 0, N_y \rrbracket, & (E_{0,j}) \\
U_{N_x,j} &= g_E(b, y_j), & \forall j \in \llbracket 0, N_y \rrbracket, & (E_{N_x,j}) \\
U_{i,0} &= g_S(x_i, c), & \forall i \in \llbracket 0, N_x \rrbracket, & (E_{i,0}) \\
U_{i,N_y} &= g_N(x_i, d), & \forall i \in \llbracket 0, N_x \rrbracket. & (E_{i,N_y})
\end{aligned}$$

avec $\beta_x = \frac{-1}{h_x^2}$, $\beta_y = \frac{-1}{h_y^2}$, $\mu = \frac{2}{h_x^2} + \frac{2}{h_y^2}$ et en espérant $U_{i,j} \approx u(x_i, y_j)$!

$(E_{i,j})$  équation portée par le point (x_i, y_j)

Q. 4

Déterminer

- ① les conditions de compatibilité ?
- ② le nombre d'équations et le nombre d'inconnues ?

$$\begin{aligned}
 \beta_x(U_{i+1,j} + U_{i-1,j}) + \beta_y(U_{i,j+1} + U_{i,j-1}) + \mu U_{i,j} &= f_{i,j}, & \forall (i,j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket, & (E_{i,j}) \\
 U_{0,j} &= g_O(a, y_j), & \forall j \in \llbracket 0, N_y \rrbracket, & (E_{0,j}) \\
 U_{N_x,j} &= g_E(b, y_j), & \forall j \in \llbracket 0, N_y \rrbracket, & (E_{N_x,j}) \\
 U_{i,0} &= g_S(x_i, c), & \forall i \in \llbracket 0, N_x \rrbracket, & (E_{i,0}) \\
 U_{i,N_y} &= g_N(x_i, d), & \forall i \in \llbracket 0, N_x \rrbracket. & (E_{i,N_y})
 \end{aligned}$$

avec $\beta_x = \frac{-1}{h_x^2}$, $\beta_y = \frac{-1}{h_y^2}$, $\mu = \frac{2}{h_x^2} + \frac{2}{h_y^2}$ et en espérant $U_{i,j} \approx u(x_i, y_j)$!

On note $n_x = N_x + 1$, $n_y = N_y + 1$ et $N = n_x n_y$.

$\forall (i,j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket$, $(E_{i,j})$, système linéaire de N équations à N inconnues :

$$\left(\forall (i,j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket, (E_{i,j}) \right) \Leftrightarrow \mathbb{A} \mathbf{V} = \mathbf{b}, \text{ avec } \mathbb{A} \in \mathcal{M}_N(\mathbb{R}), \mathbf{V} \in \mathbb{R}^N, \mathbf{b} \in \mathbb{R}^N. \quad (8)$$

Les inconnues $(U_{i,j})$ doivent être décrites sous la forme d'un vecteur $\mathbf{V} \in \mathbb{R}^N$!

Numérotations

Les inconnues $(U_{i,j}) \approx u(x_i, y_j)$, sont en numérotation dites **mathématique** ou **2D** du TP.

Ecriture en numérotation dites **vectoriel** ou **global** du TP : le choix n'est pas unique !

On choisit

$$\mathbf{V} = \begin{pmatrix} \frac{U_{:,0}}{U_{:,1}} \\ \vdots \\ U_{:,N_y} \end{pmatrix} \quad \text{où} \quad \mathbf{U}_{:,j} = \begin{pmatrix} U_{0,j} \\ \vdots \\ U_{N_x,j} \end{pmatrix} \in \mathbb{R}^{n_x}, \quad \forall j \in \llbracket 0, N_y \rrbracket.$$

Il existe une bijection $\mathcal{F} : \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket \longrightarrow \llbracket 1, N \rrbracket$ telle que

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

$k \in \llbracket 1, N \rrbracket$ correspond à la numérotation dites **globale** dans le TP.

Q. 5

Déterminer la bijection $\mathcal{F} : \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket \longrightarrow \llbracket 1, N \rrbracket$ et sa réciproque telle que

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

$$\mathbf{V} = \begin{pmatrix} \mathbf{U}_{:,0} \\ \mathbf{U}_{:,1} \\ \vdots \\ \mathbf{U}_{:,j} \\ \vdots \\ \mathbf{U}_{:,N_y} \end{pmatrix} = \begin{pmatrix} U_{0,0} \\ \vdots \\ U_{N_x,0} \\ U_{0,1} \\ \vdots \\ U_{N_x,1} \\ \vdots \\ U_{0,j} \\ \vdots \\ U_{N_x,j} \\ \vdots \\ U_{0,N_y} \\ \vdots \\ U_{N_x,N_y} \end{pmatrix} \quad \text{avec } n_x = N_x + 1$$

Indices de blocs (à gauche) : $n_x, n_x, \dots, n_x, \dots, n_x, (j+1)n_x, \dots, N_y n_x + 1, \dots, (N_y + 1)n_x$

Indices de lignes (à gauche) : $1, \dots, n_x, n_x + 1, \dots, 2n_x, \dots, jn_x + 1, \dots, (j+1)n_x, \dots, N_y n_x + 1, \dots, (N_y + 1)n_x$

- L'indice global du dernier élément du bloc $\mathbf{U}_{:,j}$: $(j+1)n_x$ avec $n_x = N_x + 1$

$$\mathcal{F} : \begin{matrix} \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket \\ (i, j) \end{matrix} \longrightarrow \begin{matrix} \llbracket 1, N \rrbracket \\ jn_x + i + 1 \end{matrix} \quad (9)$$

- Pour déterminer $\mathcal{F}^{-1}$: soit $k \in \llbracket 1, N \rrbracket$, on a

$$k = \mathcal{F}(i, j) \Leftrightarrow k - 1 = jn_x + i$$

Division Euclidienne de $k - 1$ par n_x : j quotient, i reste.

$$i = (k - 1) \bmod n_x, \quad \text{et } j = E\left(\frac{k - 1}{n_x}\right)$$

$$\mathcal{F}^{-1} : \begin{matrix} \llbracket 0, N \rrbracket \\ k \end{matrix} \longrightarrow \begin{matrix} \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket \\ \left((k - 1) \bmod n_x, E\left(\frac{k - 1}{n_x}\right) \right) \end{matrix} \quad (10)$$

```
function k=bijF(i,j,nx)
% i in [0,nx[, j in [0,ny[
% k in [1,nx*ny]
k=j*nx+i+1;
end
```

```
function [i,j]=bijRecF(k,nx)
% k in [1,nx*ny] avec k-1=j*nx+i
% division euclidienne de k-1 par nx
% reste: i in [0,nx[,
% quotient: j in [0,ny[
i=rem(k-1,nx); % ou mod(k-1,nx)
j=floor((k-1)/nx); % ou (k-1-i)/nx
end
```

Q. 6

Que font les programmes suivants

Listing – Fichier bijection00.m

```
clear all;close all
Nx=4;Ny=5;l=0:Nx;J=0:Ny;
E=0;
for j=0:Ny
    for i=0:Nx
        k=bijF(i,j,Nx+1);
        [ii,jj]=bijRecF(k,Nx+1);
        E=E+abs(ii-i)+abs(jj-j);
    end
end
fprintf('Error: %g\n',E)
```

Listing – Fichier bijection01.m

```
clear all;close all
Nx=4;Ny=5;
l=0:Nx;J=0:Ny;
k=bijF(l',J,Nx+1);
k=k(:)';
li=1:Nx-1;Ji=1:Ny-1;
ki=reshape(bijF(li',Ji,Nx+1),1,[]);
```

Listing – Fichier bijection03.m

```
clear all;close all
Nx=4;Ny=5;l=0:Nx;J=0:Ny;
k=1:(Nx+1)*(Ny+1);
[i,j]=bijRecF(k,Nx+1);
kk=bijF(i,j,Nx+1);
E=sum(abs(k-kk));
fprintf('Error: %g\n',E)
```

$$\mathbb{A}\mathbf{V} = \mathbf{b}, \quad \text{avec } \mathbb{A} \in \mathcal{M}_N(\mathbb{R}), \quad \mathbf{V} \in \mathbb{R}^N, \mathbf{b} \in \mathbb{R}^N \quad (8)$$

Ecriture $n_y \times n_y$ blocs carrés de $\mathcal{M}_{n_x}(\mathbb{R})$:

$$\mathbb{A}\mathbf{V} = \begin{matrix} & \begin{matrix} n_x & & n_x & & n_x \end{matrix} \\ \begin{matrix} n_x \\ \\ \\ n_x \end{matrix} & \begin{pmatrix} \mathbb{A}_{0,0} & \dots & \mathbb{A}_{0,r} & \dots & \mathbb{A}_{0,N_y} \\ \vdots & \ddots & \vdots & & \vdots \\ \mathbb{A}_{r,0} & \dots & \mathbb{A}_{r,r} & \dots & \mathbb{A}_{r,N_y} \\ \vdots & & \vdots & \ddots & \vdots \\ \mathbb{A}_{N_y,0} & \dots & \mathbb{A}_{N_y,r} & \dots & \mathbb{A}_{N_y,N_y} \end{pmatrix} \end{matrix} \begin{pmatrix} \mathbf{U}_{:,0} \\ \vdots \\ \mathbf{U}_{:,r} \\ \vdots \\ \mathbf{U}_{:,N_y} \end{pmatrix} = \begin{pmatrix} \mathbf{B}_0 \\ \vdots \\ \mathbf{B}_r \\ \vdots \\ \mathbf{B}_{N_y} \end{pmatrix} = \mathbf{B} \quad (11)$$

Choix écriture $(E_{i,j})$ dans (11)  **ligne $k = \mathcal{F}(i,j)$**

Produit matriciel bloc avec les notations proposées :

$$\forall r \in \llbracket 0, N_y \rrbracket, \quad \mathbf{B}_r = \sum_{s=0}^{N_y} \mathbb{A}_{r,s} \mathbf{U}_{:,s}.$$

$$\mathbb{A}\mathbf{V} = \begin{matrix} & \begin{matrix} n_x & & n_x & & n_x \end{matrix} \\ \begin{matrix} n_x \\ n_x \\ n_x \end{matrix} & \begin{pmatrix} \mathbb{A}_{\mathbf{0},\mathbf{0}} & \dots & \mathbb{A}_{\mathbf{0},r} & \dots & \mathbb{A}_{\mathbf{0},N_y} \\ \vdots & \ddots & \vdots & & \vdots \\ \mathbb{A}_{r,\mathbf{0}} & \dots & \mathbb{A}_{r,r} & \dots & \mathbb{A}_{r,N_y} \\ \vdots & & \vdots & \ddots & \vdots \\ \mathbb{A}_{N_y,\mathbf{0}} & \dots & \mathbb{A}_{N_y,r} & \dots & \mathbb{A}_{N_y,N_y} \end{pmatrix} \end{matrix} \begin{pmatrix} \mathbf{U}_{:,0} \\ \vdots \\ \mathbf{U}_{:,r} \\ \vdots \\ \mathbf{U}_{:,N_y} \end{pmatrix} = \begin{pmatrix} \mathbf{B}_0 \\ \vdots \\ \mathbf{B}_r \\ \vdots \\ \mathbf{B}_{N_y} \end{pmatrix} = \mathbf{B} \quad (11)$$

$$\begin{aligned}
\beta_x(U_{i+\mathbf{1},j} + U_{i-\mathbf{1},j}) + \beta_y(U_{i,j+\mathbf{1}} + U_{i,j-\mathbf{1}}) + \mu U_{i,j} &= f_{i,j}, & \forall (i,j) \in \llbracket 0, N_x \llbracket \times \llbracket 0, N_y \llbracket, & (E_{i,j}) \\
U_{\mathbf{0},j} &= g_O(x_{\mathbf{0}}, y_j), & \forall j \in \llbracket 0, N_y \llbracket, & (E_{\mathbf{0},j}) \\
U_{N_x,j} &= g_E(x_{N_x}, y_j), & \forall j \in \llbracket 0, N_y \llbracket, & (E_{N_x,j}) \\
U_{i,\mathbf{0}} &= g_S(x_i, y_{\mathbf{0}}), & \forall i \in \llbracket 0, N_x \rrbracket, & (E_{i,\mathbf{0}}) \\
U_{i,N_y} &= g_N(x_i, y_{N_y}), & \forall i \in \llbracket 0, N_x \rrbracket. & (E_{i,N_y})
\end{aligned}$$

avec $\beta_x = \frac{-1}{h_x^2}$, $\beta_y = \frac{-1}{h_y^2}$, $\mu = \frac{2}{h_x^2} + \frac{2}{h_y^2}$

Choix écriture $(E_{i,j})$ dans (11)  **ligne $k = \mathcal{F}(i,j) = jn_x + i + 1$**

Q. 7

- ① *Expliciter la première et la dernière ligne bloc de (11) (i.e. ligne bloc $r = 0$ et $r = N_y$).*
- ② *Expliciter la ligne bloc r de (11) pour $r \in \llbracket 0, N_y - 1 \rrbracket$.*

$\mathbb{A}\mathbf{V} = \mathbf{B}$ s'écrit donc sous forme bloc

$$\begin{pmatrix} \mathbb{I} & 0 & \dots & \dots & 0 & 0 \\ \mathbb{M} & \mathbb{D} & \mathbb{M} & 0 & \dots & 0 & 0 \\ 0 & \mathbb{M} & \ddots & \ddots & \ddots & \ddots & \ddots \\ \vdots & 0 & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & 0 & \dots & 0 & \mathbb{M} & \mathbb{D} & \mathbb{M} \\ 0 & 0 & \dots & \dots & 0 & 0 & \mathbb{I} \end{pmatrix} \mathbf{V} = \begin{pmatrix} \mathbf{B}_0 \\ \mathbf{B}_1 \\ \vdots \\ \mathbf{B}_{N_y-1} \\ \mathbf{B}_{N_y} \end{pmatrix} \quad (12)$$

avec $\beta_x = \frac{-1}{h_x^2}$, $\beta_y = \frac{-1}{h_y^2}$, $\mu = \frac{2}{h_x^2} + \frac{2}{h_y^2}$,

$$\mathbb{D} = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 & 0 \\ \beta_x & \mu & \beta_x & 0 & \dots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & \dots & 0 & \beta_x & \mu & \beta_x \\ 0 & \dots & \dots & 0 & 0 & 1 \end{pmatrix} \in \mathcal{M}_{n_x}(\mathbb{R}),$$

$$\mathbb{M} = \begin{pmatrix} 0 & 0 & \dots & \dots & 0 & 0 \\ 0 & \beta_y & 0 & \dots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & \dots & 0 & \beta_y & 0 & 0 \\ 0 & \dots & \dots & 0 & 0 & 0 \end{pmatrix} \in \mathcal{M}_{n_x}(\mathbb{R}),$$

$$\mathbf{B}_0 = \begin{pmatrix} g_S(x_0, y_0) \\ \vdots \\ g_S(x_{N_x}, y_0) \end{pmatrix} \in \mathbb{R}^{n_x}, \quad \mathbf{B}_{N_y} = \begin{pmatrix} g_N(x_0, y_{N_y}) \\ \vdots \\ g_N(x_{N_x}, y_{N_y}) \end{pmatrix} \in \mathbb{R}^{n_x},$$

$$\forall r \in \llbracket 0, N_y \rrbracket, \quad \mathbf{B}_r = \begin{pmatrix} \frac{g_O(x_0, y_r)}{f(x_1, y_r)} \\ \vdots \\ \frac{f(x_{N_x-1}, y_r)}{g_E(x_{N_x}, y_r)} \end{pmatrix} \in \mathbb{R}^{n_x}.$$

- 1 Matrices et tableaux
 - Stockage
 - Numérotation 1D
 - Sous-matrices
- 2 Représentations graphiques
- 3 Différences finies 2D
 - Schéma différences finies
 - Système linéaire
 - Numérotations
 - Ecriture bloc
 - Assemblage
 - Méthode bloc
 - Méthode bijection

Méthode bloc

$\mathbb{A}\mathbf{V} = \mathbf{B}$ s'écrit donc sous forme bloc

$$\begin{pmatrix} \text{I} & 0 & \dots & 0 & 0 \\ \text{M} & \text{D} & \text{M} & 0 & 0 \\ 0 & \text{M} & \ddots & \ddots & 0 \\ \vdots & 0 & \ddots & \ddots & \text{M} \\ 0 & 0 & \dots & 0 & \text{D} \\ 0 & 0 & \dots & 0 & \text{I} \end{pmatrix} \mathbf{V} = \begin{pmatrix} \mathbf{B}_0 \\ \mathbf{B}_1 \\ \vdots \\ \mathbf{B}_{N_y-1} \\ \mathbf{B}_{N_y} \end{pmatrix} \quad (13)$$

avec $\beta_x = \frac{-1}{h_x^2}$, $\beta_y = \frac{-1}{h_y^2}$, $\mu = \frac{2}{h_x^2} + \frac{2}{h_y^2}$,

$$\mathbb{D} = \begin{pmatrix} 1 & 0 & \dots & 0 & 0 \\ \beta_x & \mu & \beta_x & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & \beta_x & \mu \\ 0 & \dots & 0 & 0 & 1 \end{pmatrix} \in \mathcal{M}_{n_x}(\mathbb{R}),$$

$$\mathbb{M} = \begin{pmatrix} 0 & 0 & \dots & 0 & 0 \\ 0 & \beta_y & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & \beta_y & 0 \\ 0 & \dots & 0 & 0 & 0 \end{pmatrix} \in \mathcal{M}_{n_x}(\mathbb{R}),$$

$$\mathbf{B}_0 = \begin{pmatrix} g_S(x_0, y_0) \\ \vdots \\ g_S(x_{N_x}, y_0) \end{pmatrix} \in \mathbb{R}^{n_x}, \quad \mathbf{B}_{N_y} = \begin{pmatrix} g_N(x_0, y_{N_y}) \\ \vdots \\ g_N(x_{N_x}, y_{N_y}) \end{pmatrix} \in \mathbb{R}^{n_x},$$

$$\forall r \in \llbracket 0, N_y \rrbracket, \quad \mathbf{B}_r = \begin{pmatrix} \frac{g_O(x_0, y_r)}{f(x_1, y_r)} \\ \vdots \\ \frac{f(x_{N_x}-1, y_r)}{g_E(x_{N_x}, y_r)} \end{pmatrix} \in \mathbb{R}^{n_x}.$$

Q. 8

Proposer un algorithme permettant de créer la matrice $\mathbb{A}$ et le vecteur $\mathbf{B}$ en utilisant la structure bloc.

- **Avantages**

- ▶ algorithmes assez simples,
- ▶ vectorisation possible,
- ▶ ...

- **Inconvénients**

- ▶ il faut impérativement avoir écrit le système bloc en détail,
- ▶ passage en 3D,

Méthode bijection

$$\mathbb{A}\mathbf{V} = \mathbf{B} \quad \text{avec } \mathbf{V}_{\mathcal{F}(i,j)} = U_{i,j}, \quad \forall (i,j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket \quad (11)$$

$$\beta_x(U_{i+1,j} + U_{i-1,j}) + \beta_y(U_{i,j+1} + U_{i,j-1}) + \mu U_{i,j} = f(x_i, y_j), \quad \forall (i,j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket, \quad (E_{i,j})$$

+C.L.

avec $\beta_x = \frac{-1}{h_x^2}$, $\beta_y = \frac{-1}{h_y^2}$, $\mu = \frac{2}{h_x^2} + \frac{2}{h_y^2}$

Choix écriture $(E_{i,j})$ dans (11)  ligne $k = \mathcal{F}(i,j) = jn_x + i + 1$

Q. 9

Proposer un algorithme permettant de créer la matrice $\mathbb{A} \in \mathcal{M}_N(\mathbb{R})$ et le second membre $\mathbf{B} \in \mathbb{R}^N$ sans tenir compte des conditions aux limites (i.e. les lignes correspondantes sont laissées à zéros)

Soit $(i,j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket$ et $k = \mathcal{F}(i,j)$. La ligne k du système est donc

$$\beta_x(V_{\mathcal{F}(i+1,j)} + V_{\mathcal{F}(i-1,j)}) + \beta_y(V_{\mathcal{F}(i,j+1)} + V_{\mathcal{F}(i,j-1)}) + \mu V_{\mathcal{F}(i,j)} = f(x_i, y_j)$$

$$\beta_x(V_{k+1} + V_{k-1}) + \beta_y(V_{k+n_x} + V_{k-n_x}) + \mu V_k = f(x_i, y_j)$$

$$(\mathbb{A}\mathbf{V} = \mathbf{B})_k \Leftrightarrow \sum_{r=1}^N A_{k,r} V_r = B_k \Leftrightarrow \begin{cases} \forall r \in \llbracket 1, N \rrbracket \setminus \{k-n_x, k-1, k, k+1, k+n_x\}, & A_{k,r} = 0, \\ A_{k,k+1} V_{k+1} + A_{k,k-1} V_{k-1} + A_{k,k+n_x} V_{k+n_x} + A_{k,k-n_x} V_{k-n_x} + \mu A_{k,k} V_k = B_k \end{cases}$$

$$A_{k,k-1} = A_{k,k+1} = \beta_x, \quad A_{k,k-n_x} = A_{k,k+n_x} = \beta_y, \quad A_{k,k} = \mu, \quad \text{sinon } A_{k,r} = 0$$

$$B_k = f(x_i, y_j), \quad \text{où } k = \mathcal{F}(i,j), \quad \text{et } (i,j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket$$

$$\mathbb{A}\mathbf{V} = \mathbf{B} \quad \text{avec } \mathbf{V}_{\mathcal{F}(i,j)} = U_{i,j}, \quad \forall (i,j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket \quad (11)$$

$$\begin{aligned} \beta_x(U_{i+1,j} + U_{i-1,j}) + \beta_y(U_{i,j+1} + U_{i,j-1}) + \mu U_{i,j} &= f(x_i, y_j), & \forall (i,j) \in \llbracket 0, N_x \rrbracket \times \llbracket 0, N_y \rrbracket, & (E_{i,j}) \\ U_{0,j} &= g_O(x_0, y_j), & \forall j \in \llbracket 0, N_y \rrbracket, & (E_{0,j}) \\ U_{N_x,j} &= g_E(x_{N_x}, y_j), & \forall j \in \llbracket 0, N_y \rrbracket, & (E_{N_x,j}) \\ U_{i,0} &= g_S(x_i, y_0), & \forall i \in \llbracket 0, N_x \rrbracket, & (E_{i,0}) \\ U_{i,N_y} &= g_N(x_i, y_{N_y}), & \forall i \in \llbracket 0, N_x \rrbracket. & (E_{i,N_y}) \end{aligned}$$

avec $\beta_x = \frac{-1}{h_x^2}$, $\beta_y = \frac{-1}{h_y^2}$, $\mu = \frac{2}{h_x^2} + \frac{2}{h_y^2}$

Choix écriture $(E_{i,j})$ dans (11)  ligne $k = \mathcal{F}(i,j) = jn_x + i + 1$

Q. 10

Proposer un algorithme permettant de créer la matrice $\mathbb{A} \in \mathcal{M}_N(\mathbb{R})$ et le second membre $\mathbf{B} \in \mathbb{R}^N$ en ne tenant compte que des conditions aux limites (i.e. les lignes correspondant aux points intérieurs sont laissées à zéros)

- **Avantages**

- ▶ pas nécessaire d'avoir l'écriture bloc,
- ▶ algorithmes simples,
- ▶ vectorisation possible,
- ▶ passage en 3D,
- ▶ ...

- **Inconvénients**

- ▶ *cache* la structure de la matrice
- ▶ ...