### Analyse Numérique I\*

Sup'Galilée, Ingénieurs MACS, 1ère année / L3 MIM

#### François Cuvelier

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

2025/10/24

- [1] F. Cuvelier, Analyse numérique I, résolution de systèmes linéaires, méthodes directes, résumé. fichier pdf, https://www.math.univ-paris13.fr/~cuvelier/docs/Enseignements/MACS1/AnaNumI/25-26/resume\_RSLdirecte.pdf.
- [2] F. Cuvelier, Analyse numérique élémentaire (version du 29 sep. 2025), Polycopié (téléchargement), 2025.

# Chapitre IV

Résolution de systèmes linéaires

### Plan

- Conditionnement
- Méthodes directes
  - Matrices diagonales
  - Matrices triangulaires inférieures

- Matrices triangulaires supérieures
- Ecriture algébrique
- Résultats théorique
- Utilisation pratique
- Résultats théoriques
- Résolution d'un système linéaire
- Algorithme : Factorisation positive de Cholesky
- La tranformation de Householder

Soient  $\mathbb{A} \in \mathcal{M}_n(\mathbb{K})$  une matrice inversible et  $\boldsymbol{b} \in \mathbb{K}^n$ .

Résoudre

$$Ax = b$$

Le calcul de la matrice inverse  $\mathbb{A}^{-1}$  revient à résoudre n systèmes linéaires.



Nour résoudre un système linéaire, on ne calcule pas la matrice inverse associée.

**Méthodes directes:** On cherche M inversible tel que MA facilement inversible

$$\mathbb{A}\mathbf{x} = \mathbf{b} \iff \mathbb{M}\mathbb{A}\mathbf{x} = \mathbb{M}\mathbf{b}.$$

**Méthodes itératives :** On cherche  $\mathbb{B}$  et  $\boldsymbol{c}$ ,

$$\mathbf{x}^{[k+1]} = \mathbb{B}\mathbf{x}^{[k]} + \mathbf{c}, \quad k \geqslant 0, \ \mathbf{x}^{[0]} \text{ donné}$$

en espérant  $\lim_{k\to+\infty} \mathbf{x}^{[k]} = \mathbf{x}$ .



### Conditionnement

Soient  $\mathbb{A} \in \mathcal{M}_n(\mathbb{K})$  inversible et  $\boldsymbol{b} \in \mathbb{K}^n$ .

$$Ax = b$$

De petites erreurs sur les données engendrent-elles de petites erreurs sur la solution?

Conditionnement

# Exemple de R.S. Wilson

Soient

$$\mathbb{A} = \begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix}, \quad \Delta \mathbb{A} = \begin{pmatrix} 0 & 0 & \frac{1}{10} & \frac{1}{5} \\ \frac{2}{25} & \frac{1}{25} & 0 & 0 \\ 0 & -\frac{1}{50} & -\frac{11}{100} & 0 \\ -\frac{1}{100} & -\frac{1}{100} & 0 & -\frac{1}{50} \end{pmatrix}$$

et 
$$\textbf{\textit{b}}^{t}=\left(32,\,23,\,33,\,31\right),\, \left(\pmb{\Delta \textit{b}}\right)^{t}=\left(\frac{1}{100},\,-\frac{1}{100},\,\frac{1}{100},\,-\frac{1}{100}\right)$$
 . Des calculs exacts donnent

◆□▶◆□▶◆□▶◆□▶ □ り<00</p>

Conditionnement

Soient  $\mathbb{A} \in \mathcal{M}_n(\mathbb{K})$  inversible et  $\boldsymbol{b} \in \mathbb{K}^n$ .

$$Ax = b$$

De petites erreurs sur les données engendrent-elles de petites erreurs sur la solution?

Non, pas forcément!

Le système linéaire prédédent est mal conditionné.

On dit qu'un système linéaire est **bien conditionné** ou qu'il a un **bon conditionnement** si de petites perturbations des données n'entrainent qu'une variation *raisonnable* de la solution.

Est-il possible de "mesurer" le conditionnement d'une matrice?

7 / 68

Conditionnement 2025/10/24



### **Definition**

Soit  $\|.\|$  une norme matricielle subordonnée, le conditionnement d'une matrice régulière  $\mathbb{A}$ , associé à cette norme, est le nombre

$$\operatorname{cond}(\mathbb{A}) = \|\mathbb{A}\| \|\mathbb{A}^{-1}\|.$$

Nous noterons  $\operatorname{cond}_p(\mathbb{A}) = \|\mathbb{A}\|_p \|\mathbb{A}^{-1}\|_p$ .

Conditionnement

8 / 68



### Proposition:

Soit A une matrice régulière. On a les propriétés suivantes

- $\forall \alpha \in \mathbb{K}^*, \operatorname{cond}(\alpha \mathbb{A}) = \operatorname{cond}(\mathbb{A}).$
- $\operatorname{cond}_{p}(\mathbb{A}) \geqslant 1, \ \forall p \in [1, +\infty].$

Conditionnement

Théorème : 3



Soit A une matrice inversible. Soient x et  $x + \Delta x$  les solutions respectives de

$$Ax = b$$
 et  $A(x + \Delta x) = b + \Delta b$ .

Supposons  $b \neq 0$ , alors l'inégalité

$$\frac{\|\boldsymbol{\Delta}\boldsymbol{x}\|}{\|\boldsymbol{x}\|}\leqslant \operatorname{cond}(\mathbb{A})\frac{\|\boldsymbol{\Delta}\boldsymbol{b}\|}{\|\boldsymbol{b}\|}$$

est satisfaite, et c'est la meilleure possible : pour une matrice A donnée, on peut trouver des vecteurs  $\mathbf{b} \neq \mathbf{0}$  et  $\Delta \mathbf{b} \neq \mathbf{0}$  tels qu'elle devienne une égalité.

10 / 68

Conditionnement 2025/10/24 Théorème : 48

Soient  $\mathbb{A}$  et  $\mathbb{A} + \Delta \mathbb{A}$  deux matrices inversibles. Soient  $\mathbf{x}$  et  $\mathbf{x} + \Delta \mathbf{x}$  les solutions respectives de

$$\mathbb{A}\mathbf{x} = \mathbf{b}$$
 et  $(\mathbb{A} + \Delta\mathbb{A})(\mathbf{x} + \Delta\mathbf{x}) = \mathbf{b}$ .

Supposons  $b \neq 0$ , alors on a

$$\frac{\|\Delta x\|}{\|x + \Delta x\|} \leqslant \operatorname{cond}(\mathbb{A}) \frac{\|\Delta \mathbb{A}\|}{\|\mathbb{A}\|}.$$

#### Remarque 1.1

Une matrice est donc bien conditionnée si son conditionnement est proche de 1.

11 / 68

Conditionnement 2025/10/24

### Plan

- Conditionnement
- Méthodes directes
  - Matrices particulières
    - Matrices diagonale
    - Matrices triangulaires inférieures
    - Matrices triangulaires supérieures
  - Exercices et résultats préliminaires
  - Méthode de Gauss-Jordan
    - Ecriture algébrique

- Factorisation LU
  - Résultats théoriques
  - Utilisation pratique
- Factorisation LDL\*
- Factorisation de Cholesky
  - Résultats théoriques
  - Résolution d'un système linéaire
  - Algorithme : Factorisation positive de Cholesky
- Factorisation QR
  - La tranformation de Householder

12 / 68

Méthodes directes 2025/10/24

# Système diagonal

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{K})$  diagonale inversible et  $\boldsymbol{b} \in \mathbb{K}^n$ .

$$x_i = b_i/A_{i,i}, \quad \forall i \in \llbracket 1, n \rrbracket. \tag{1}$$

Algorithme Fonction RSLMatDiag permettant de résoudre le système linéaire à matrice diagonale inversible

$$Ax = b$$
.

**Données :** A : matrice diagonale de  $\mathcal{M}_n(\mathbb{R})$  inversible.

**b** : vecteur de  $\mathbb{R}^n$ .

**Résultat :** x : vecteur de  $\mathbb{R}^n$ .

- 1: Fonction  $x \leftarrow \text{RSLMatDiag}(A, b)$
- 2: Pour  $i \leftarrow 1$  à n faire
- 3:  $x(i) \leftarrow b(i)/A(i,i)$
- 4: Fin Pour
- 5: Fin Fonction



13 / 68

Méthodes directes Matrices particulières 2025/10/24

# Système triangulaire inférieur

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{K})$  triangulaire inférieure inversible  $(A_{i,j} = 0 \text{ si } i < j)$ 

$$\mathbb{A}\mathbf{x} = \mathbf{b} \iff \begin{pmatrix} A_{1,1} & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ A_{n,1} & \dots & \dots & A_{n,n} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ \vdots \\ \vdots \\ b_n \end{pmatrix}$$

A inversible ←⇒

Méthodes directes 2025/10/24 14 / 68

# Système triangulaire inférieur

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{K})$  triangulaire inférieure inversible  $(A_{i,j} = 0 \text{ si } i < j)$ 

$$\mathbb{A}\mathbf{x} = \mathbf{b} \iff \begin{pmatrix} A_{1,1} & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ A_{n,1} & \dots & \dots & A_{n,n} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}$$

 $\mathbb{A}$  inversible  $\iff A_{i,i} \neq 0, \forall i \in [1, n]$ 

Méthodes directes 2025/10/24 14 / 68

# Système triangulaire inférieur

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{K})$  triangulaire inférieure inversible  $(A_{i,j} = 0 \text{ si } i < j)$ 

$$\mathbb{A}\mathbf{x} = \mathbf{b} \iff \begin{pmatrix} A_{1,1} & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ A_{n,1} & \dots & \dots & A_{n,n} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ \vdots \\ \vdots \\ b_n \end{pmatrix}$$

 $\mathbb{A}$  inversible  $\iff A_{i,i} \neq 0, \forall i \in [1, n]$ 

Soit 
$$i \in [1, n]$$
,  $\left( (\mathbb{A}\mathbf{x})_i = b_i, \iff \sum_{i=1}^n A_{i,j} x_j = b_i \right)$ .

$$b_{i} = \sum_{j=1}^{i-1} A_{i,j} x_{j} + A_{i,i} x_{i} + \sum_{j=i+1}^{n} \underbrace{A_{i,j}}_{=0} x_{j} = \sum_{j=1}^{i-1} A_{i,j} x_{j} + A_{i,i} x_{i}$$

$$x_i = \frac{1}{A_{i,i}} \left( b_i - \sum_{i=1}^{i-1} A_{i,j} x_j \right), \ \forall i \in \llbracket 1, n \rrbracket.$$

(2)

Méthodes directes 2025/10/24 14 / 68

$$x_i = \frac{1}{A_{i,i}} \left( b_i - \sum_{i=1}^{i-1} A_{i,j} x_j \right), \ \forall i \in [1, n].$$

### Algorithme 2 $\mathcal{R}_0$

Résoudre Ax = b en calculant successivement  $x_1, x_2, \ldots, x_n$ .

Algorithme 2  $\mathcal{R}_1$ 

1: Pour 
$$i \leftarrow 1$$
 à  $n$  faire  
2:  $x_i \leftarrow \frac{1}{A_{i,i}} \left( b_i - \sum_{j=1}^{i-1} A_{i,j} x_j \right)$   
3: Fin Pour

15 / 68

Méthodes directes Matrices particulières 2025/10/24

$$x_i = \frac{1}{A_{i,i}} \left( b_i - \sum_{j=1}^{i-1} A_{i,j} x_j \right), \ \forall i \in [1, n].$$

### Algorithme 2 $\mathcal{R}_1$

1: Pour  $i \leftarrow 1$  à n faire

$$\left[x_i \leftarrow \frac{1}{A_{i,i}} \left(b_i - \sum_{j=1}^{i-1} A_{i,j} x_j\right)\right]$$

- 3: Fin Pour

# Algorithme 2 $\mathbb{R}_2$

1. Pour  $i \leftarrow 1$  à n faire

2: 
$$S \leftarrow \sum_{j=1}^{i-1} A_{i,j} x_j$$
  
3:  $x_i \leftarrow (b_i - S)/A_{i,i}$ 

4: Fin Pour

15 / 68

Méthodes directes Matrices particulières 2025/10/24

$$x_i = \frac{1}{A_{i,i}} \left( b_i - \sum_{j=1}^{i-1} A_{i,j} x_j \right), \ \forall i \in [1, n].$$

### Algorithme 2 $\mathcal{R}_2$

1: Pour  $i \leftarrow 1$  à n faire

$$S \leftarrow \sum_{j=1}^{i-1} A_{i,j} x_j$$

- 3:  $x_i \leftarrow (b_i S)/A_{i,i}$
- 4. Fin Pour

# Algorithme 2 $\mathbb{R}_3$

1: Pour  $i \leftarrow 1$  à n faire

2: 
$$S \leftarrow 0$$
  
Pour  $j \leftarrow 1$  à  $i - 1$  faire  $S \leftarrow S + A(i,j) * x(j)$ 

Fin Pour

6: 
$$x_i \leftarrow (b_i - S)/A_{i,i}$$

7: Fin Pour

15 / 68

Méthodes directes Matrices particulières 2025/10/24

$$x_i = \frac{1}{A_{i,i}} \left( b_i - \sum_{j=1}^{i-1} A_{i,j} x_j \right), \ \forall i \in [1, n].$$

**Algorithme** Fonction RSLTriInf permettant de résoudre le système linéaire triangulaire inférieur inversible  $\mathbb{A}\mathbf{x} = \mathbf{b}$ .

**Données :**  $\mathbb{A}$  : matrice triangulaire de  $\mathcal{M}_n(\mathbb{K})$  inférieure inversible.

**b** : vecteur de  $\mathbb{K}^n$ .

**Résultat :** x : vecteur de  $\mathbb{K}^n$ .

- 1: Fonction  $x \leftarrow \text{RSLTriInf}(A, b)$
- 2: Pour  $i \leftarrow 1$  à n faire
- 3: *S* ← 0
  - Pour  $j \leftarrow 1$  à i-1 faire
- 5:  $S \leftarrow S + A(i,j) * x(j)$
- 6: Fin Pour
- 7:  $x(i) \leftarrow (b(i) S)/A(i, i)$
- 8: Fin Pour
- 9: Fin Fonction

15 / 68

Méthodes directes Matrices particulières 2025/10/24

# Système triangulaire supérieur

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{K})$  triangulaire supérieure inversible  $(A_{i,j} = 0 \text{ si } i > j)$ 

$$\mathbb{A}\mathbf{x} = \mathbf{b} \iff \begin{pmatrix} A_{1,1} & \dots & \dots & A_{1,n} \\ 0 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & A_{n,n} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}$$

Exercice 1 Barrayail à faire

Ecrire la fonction RSLTriSup permettant de résoudre le système triangulaire supérieur  $\mathbb{A} \mathbf{x} = \mathbf{b}$ .

Méthodes directes 2025/10/24 16 / 68

### Plan

- Conditionnement
- Méthodes directes
  - Matrices particulières
    - Matrices diagonales
    - Matrices triangulaires inférieures
    - Matrices triangulaires supérieures
  - Exercices et résultats préliminaires
  - Méthode de Gauss-Jordan
    - Ecriture algébrique

- Factorisation LU
  - Résultats théoriques
  - Utilisation pratique
- Factorisation LDL\*
- Factorisation de Cholesky
  - Résultats théoriques
  - Résolution d'un système linéaire
  - Algorithme : Factorisation positive de Cholesky
- Factorisation QR
  - La tranformation de Householder

# Exercice 2 : Travail à faire Matrice de permutation

Soit  $(i,j) \in [1,n]^2$ ,  $i \neq j$ , on note  $\mathbb{P}_n^{[i,j]} \in \mathcal{M}_n(\mathbb{R})$  la matrice identitée dont on a permuté les lignes i et j.

Représenter cette matrice et la définir proprement.

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$ . On note  $A_{r,:}$  le r-ème vecteur ligne de  $\mathbb{A}$  et  $A_{:,s}$  le s-ème vecteur colonne de  $\mathbb{A}$ .

Q. 2

- ① Déterminer les lignes de la matrice  $\mathbb{D} = \mathbb{P}_n^{[i,j]}\mathbb{A}$  en fonction des vecteurs lignes de  $\mathbb{A}$ .
- 2 Déterminer les colonnes de la matrice  $\mathbb{E} = \mathbb{AP}_n^{[i,j]}$  en fonction des vecteurs colonnes de  $\mathbb{A}$ .

Q. 3

- Calculer le déterminant de  $\mathbb{P}_n^{[i,j]}$ .
- 2 Déterminer l'inverse de  $\mathbb{P}_n^{[i,j]}$

#### Exercice 3: Matrice d'élimination

Soit  $\mathbf{v} \in \mathbb{C}^n$  avec  $v_1 \neq 0$ . On note  $\mathbb{E}^{[\mathbf{v}]} \in \mathcal{M}_n(\mathbb{C})$  la matrice triangulaire inférieure à diagonale unité définie par

$$\mathbb{E}^{[\mathbf{v}]} = \begin{pmatrix} \frac{1}{-v_2/v_1} & 0 & \dots & 0 \\ \vdots & 0 & \dots & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ -v_n/v_1 & 0 & \dots & 0 & 1 \end{pmatrix}$$
 (1)

Q. 1

- 2 Déterminer l'inverse de  $\mathbb{E}^{[v]}$ .

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  avec  $A_{1,1} \neq 0$ . On note  $A_{i,j}$  le j-ème vecteur colonne de  $\mathbb{A}$  et  $A_{i,i}$  son i-ème vecteur ligne. On pose  $A_1 = A_{i,1}$ .

Q. 2

- ① Soit  $\mathbf{v} \in \mathbb{C}^n$  avec  $v_1 \neq 0$ . Calculer  $\tilde{\mathbb{A}} = \mathbb{E}^{[\mathbf{v}]} \mathbb{A}$  en fonction des vecteurs lignes de  $\mathbb{A}$ .
- $\blacksquare$  En déduire que la première colonne de  $\mathbb{E}^{[A_1]}\mathbb{A}$  est le vecteur  $(A_{1,1},0,\ldots,0)^t$  i.e.

$$\mathbb{E}^{[\mathbf{A}_1]} \mathbb{A} \mathbf{e}_1 = A_{1,1} \mathbf{e}_1 \tag{2}$$

où  $\mathbf{e}_1$  est le premier vecteur de la base canonique de  $\mathbb{C}^n$ .



#### Lemme : matrice de permutation

Soit  $(i,j) \in [1,n]^2$ . On note  $\mathbb{P}_n^{[i,j]} \in \mathcal{M}_n(\mathbb{R})$  la matrice identitée dont on a permuté les lignes i et j. Alors la matrice  $\mathbb{P}_n^{[i,j]}$  est symétrique et orthogonale. Pour toute matrice  $\mathbb{A} \in \mathcal{M}_n(\mathbb{K})$ ,

- **1** la matrice  $\mathbb{P}_n^{[i,j]}\mathbb{A}$  est matrice  $\mathbb{A}$  dont on a permuté les **lignes** i et j,
- 2 la matrice  $\mathbb{AP}_n^{[i,j]}$  est matrice  $\mathbb{A}$  dont on a permuté les **colonnes** i et j,



#### Lemme: matrice d'élimination

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  avec  $A_{1,1} \neq 0$ . Il existe une matrice  $\mathbb{E} \in \mathcal{M}_n(\mathbb{C})$  triangulaire inférieure à diagonale unité telle que

$$\mathbb{E}\mathbb{A}\boldsymbol{e}_1 = A_{1,1}\boldsymbol{e}_1 \tag{3}$$

où  $e_1$  est le premier vecteur de la base canonique de  $\mathbb{C}^n$ .



### Théorème : décomposition de Schur



Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$ . Il existe une matrice unitaire  $\mathbb{U}$  et une matrice triangulaire supérieure  $\mathbb{T}$  telles que

$$\mathbb{A} = \mathbb{U}\mathbb{T}\mathbb{U}^* \tag{4}$$

### Plan

- Conditionnement
- Méthodes directes
  - Matrices particulières
    - Matrices diagonales
    - Matrices triangulaires inférieures
    - Matrices triangulaires supérieures
  - Exercices et résultats préliminaires
  - Méthode de Gauss-Jordan
    - Ecriture algébrique

- Factorisation LU
  - Résultats théoriques
  - Utilisation pratique
- Factorisation LDL\*
- Factorisation de Cholesky
  - Résultats théoriques
  - Résolution d'un système linéaire
  - Algorithme : Factorisation positive de Cholesky
- Factorisation QR
  - La tranformation de Householder

2025/10/24

22 / 68

Méthodes directes Méthode de Gauss-Jordan

# Algorithme de Gauss-Jordan

$$Ax = b \iff Ux = f$$

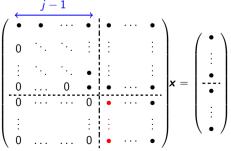
où U matrice triangulaire supérieure.

Opérations élémentaires sur les matrices :

- $\mathcal{L}_i \leftrightarrow \mathcal{L}_j$  permutation lignes i et j
- $\mathcal{L}_i \leftarrow \mathcal{L}_i + \alpha \mathcal{L}_i$  combinaison linéaire

A l'aide d'opérations élémentaires, on va transformer successivement en n-1 étapes le système.

 Etape j: on va s'arranger pour annuler les termes sous-diagonaux de la colonne j de la matrice sans modifier les j - 1 premières colonnes.



$$\begin{pmatrix}
\bullet & \bullet & \cdots & \bullet \\
0 & \ddots & \ddots & \vdots & \vdots & \vdots \\
\vdots & \ddots & \ddots & \bullet & \vdots & \vdots & \vdots \\
0 & \cdots & 0 & \bullet & \cdots & \bullet \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
0 & \cdots & 0 & \bullet & \cdots & \bullet
\end{pmatrix}$$

$$x = \begin{pmatrix}
\bullet \\
\vdots \\
\bullet \\
\vdots \\
\bullet
\end{pmatrix}$$

#### **Algorithme** Algorithme de Gauss-Jordan formel pour la résolution de $\mathbb{A}\mathbf{x} = \mathbf{b}$

- 1: Pour  $j \leftarrow 1$  à n-1 faire
- 2: Rechercher l'indice k de la ligne du pivot (sur la colonne  $j, k \in [j, n]$ )
- 3: Permuter les lignes  $j(\mathcal{L}_i)$  et  $k(\mathcal{L}_k)$  du système si besoin.
- 4: Pour  $i \leftarrow j + 1$  à *n* faire
- 5: Eliminer en effectuant  $\mathcal{L}_i \leftarrow \mathcal{L}_i \frac{A_{i,j}}{A_{i+1}} \mathcal{L}_j$
- 6: Fin Pour
- 7: Fin Pour
- 8: Résoudre le système triangulaire supérieur par la méthode de la remontée.

#### **Algorithme** Algorithme de Gauss-Jordan avec fonctions pour la résolution de $\mathbb{A}\mathbf{x} = \mathbf{b}$

```
Données :
                   \mathbb{A}: matrice de \mathcal{M}_n(\mathbb{K}) inversible.
                         : vecteur de \mathbb{K}^n.
                    b
Résultat: \mathbf{x}: vecteur de \mathbb{R}^n.
  1: Fonction x \leftarrow \text{RSLGauss}(A, b)
         Pour j \leftarrow 1 à n-1 faire
             k \leftarrow \text{ChercheIndPivot}(A, j)

⇒ à écrire

             Si k \neq i alors
 4:
                 [\mathbb{A}, \mathbf{b}] \leftarrow \overline{\text{PermLignesSys}}(\mathbb{A}, \mathbf{b}, j, k)

⇒ à écrire

             Fin Si
 6.
             Pour i \leftarrow j + 1 à n faire
                 [\mathbb{A}, \boldsymbol{b}] \leftarrow \overline{\text{CombLignesSys}} (\mathbb{A}, \boldsymbol{b}, j, i, -A(i, j)/A(j, j))
 8:

⇒ à écrire

             Fin Pour
 Q.
         Fin Pour
10.
11:
         \mathbf{x} \leftarrow \text{RSLTriSup}(\mathbb{A}, \mathbf{b})
                                                                                                                                                            12: Fin Fonction
```

25 / 68

#### Algorithme Recherche d'un pivot pour l'algorithme de Gauss-Jordan.

```
Données : \mathbb{A} : matrice de \mathcal{M}_n(\mathbb{K}).

    entier, 1 ≤ j ≤ n.

Résultat: k: dans [i, n], indice ligne pivot
 1: Fonction k \leftarrow \text{ChercheIndPivot}(\mathbb{A}, j)
        k \leftarrow j, pivot \leftarrow |\mathbb{A}(j,j)|
        Pour i \leftarrow i + 1 à n faire
           Si |\mathbb{A}(i,i)| > \text{pivot alors}
              k \leftarrow i
 5.
              pivot \leftarrow |\mathbb{A}(i, i)|
           Fin Si
        Fin Pour
 9. Fin Fonction
```

#### Algorithme Permutte deux lignes d'une matrice et d'un vecteur.

```
Données : A

 : matrice de M<sub>n</sub>(K).

                  : vecteur de K<sup>n</sup>.
              i, k: entiers, 1 \leq j, k \leq n.
Résultat : A et b modifiés.
```

```
1: Fonction [\mathbb{A}, \boldsymbol{b}] \leftarrow \text{PermLignesSys}(\mathbb{A}, \boldsymbol{b}, j, k)
          Pour l \leftarrow 1 à n faire
        t \leftarrow \mathbb{A}(i, l)
           \mathbb{A}(i, I) \leftarrow \mathbb{A}(k, I)
        A(k, l) \leftarrow t
         Fin Pour
          t \leftarrow \boldsymbol{b}(i), \, \boldsymbol{b}(i) \leftarrow \boldsymbol{b}(k), \, \boldsymbol{b}(k) \leftarrow t
8. Fin Fonction
```

```
Algorithme Combinaison linéaire \mathcal{L}_i \leftarrow \mathcal{L}_i + \alpha \mathcal{L}_i appliqué à une matrice et à un
vecteur.
```

```
Données : A
```

```
    matrice de M<sub>*</sub>(K)

         : vecteur de K<sup>n</sup>.
        : entiers. 1 ≤ i, i ≤ n.
alpha : scalaire de K
```

Résultat : A et b modifiés

```
1: Fonction [\mathbb{A}, \mathbf{b}] \leftarrow \text{CombLignesSys}(\mathbb{A}, \mathbf{b}, i, i, \alpha)
2: Pour k \leftarrow 1 à n faire
              \mathbb{A}(i, k) \leftarrow \mathbb{A}(i, k) + \alpha * \mathbb{A}(i, k)
         Fin Pour
         \boldsymbol{b}(i) \leftarrow \boldsymbol{b}(i) + \alpha \boldsymbol{b}(i)
6. Fin Fonction
```

26 / 68

### Exercice 4 Méthode de Gauss, écriture algébrique



Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  inversible.

- Montrer qu'il existe une matrice  $\mathbb{G} \in \mathcal{M}_n(\mathbb{C})$  telle que  $|\det(\mathbb{G})| = 1$  et  $\mathbb{G}\mathbb{A}\boldsymbol{e}_1 = \alpha\boldsymbol{e}_1$  avec  $\alpha \neq 0$  et **e**<sub>1</sub> premier vecteur de la base canonique de  $C^n$ .
- Q. 2 1 Montrer par récurrence sur l'ordre des matrices que pour toute matrice  $\mathbb{A}_n \in \mathcal{M}_n(\mathbb{C})$ inversible, il existe une matrice  $\mathbb{S}_n \in \mathcal{M}_n(\mathbb{C})$  telle que  $|\det \mathbb{S}_n| = 1$  et  $\mathbb{S}_n \mathbb{A}_n = \mathbb{U}_n$  avec  $\mathbb{U}_n$  matrice triangulaire supérieure inversible.
  - 2 Soit  $\mathbf{b} \in \mathbb{C}^n$ . En supposant connue la décompostion précédente  $\mathbb{S}_n \mathbb{A}_n = \mathbb{U}_n$ , expliquer comment résoudre le système  $\mathbb{A}_n \mathbf{x} = \mathbf{b}$ .
- Que peut-on dire si A est non inversible?

**Indication:** utiliser le lemme 3.1.2 de [2]/lemme 2.1 de [1] (matrice de permutation) et le lemme 3.1.1 de [2]/lemme 2.2 de [1] (matrice d'élimination)

2025/10/24

On a donc démontré la proposition suivante

#### Proposition 2.1

Soit  $\mathbb A$  une matrice carrée, inversible ou non. Il existe (au moins) une matrice inversible  $\mathbb G$  telle que  $\mathbb G\mathbb A$  soit triangulaire supérieure.

28 / 68

### Plan

- Conditionnement
- Méthodes directes
  - Matrices particulières
    - Matrices diagonales
    - Matrices triangulaires inférieures
    - Matrices triangulaires supérieures
  - Exercices et résultats préliminaires
  - Méthode de Gauss-Jordan
    - Ecriture algébrique

- ullet Factorisation  $\mathbb{L}\mathbb{U}$ 
  - Résultats théoriques
  - Utilisation pratique
- Factorisation LDL\*
- Factorisation de Cholesky
  - Résultats théoriques
  - Résolution d'un système linéaire
  - Algorithme : Factorisation positive de Cholesky
- Factorisation QR
  - La tranformation de Householder

Méthodes directes Factorisation LU 2025/10/24 29 / 68

# Exercice 5 Vers la factorisation LU

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  une matrice dont les sous-matrices principales d'ordre i, notées  $\Delta_i$ ,  $i \in [1, n]$ . Montrer par récurrence sur l'ordre n de la matrice  $\mathbb{A}$  qu'il existe une matrice  $\mathbb{E} \in \mathcal{M}_n(\mathbb{C})$ , triangulaire inférieure à diagonale unité telle que la matrice  $\mathbb{U}$  définie par

$$\mathbb{U}=\mathbb{E}\mathbb{A}$$

soit triangulaire supérieure avec  $U_{i,i} = \det \Delta_i / (U_{1,1} \times \cdots \times U_{i-1,i-1}), \ \forall i \in [1, n].$ 

Méthodes directes Factorisation LU 2025/10/24 30 / 68

#### Théorème : Factorisation LU



Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  une matrice dont les sous-matrices principales sont inversibles alors il existe

- une unique matrice  $\mathbb{L} \in \mathcal{M}_n(\mathbb{C})$  triangulaire inférieure (lower triangular en anglais) à diagonale unité,
- une unique matrice  $\mathbb{U} \in \mathcal{M}_n(\mathbb{C})$  triangulaire supérieure (upper triangular en anglais) inversible

telles ques

$$\mathbb{A} = \mathbb{LU}$$
.

# preuve : Travail à faire

- Existence : exercice précédent  $\mathbb{U}=\mathbb{E}\mathbb{A}$  et  $\mathbb{L}=\mathbb{E}^{-1}$
- Unicité :  $\mathbb{A} = \mathbb{L}_1 \mathbb{U}_1 = \mathbb{L}_2 \mathbb{U}_2 \dots$

Méthodes directes Factorisation LU 2025/10/24 31 / 68

## Exercice Barrayail à faire

Montrer que si  $\mathbb A$  inversible admet une factorisation  $\mathbb L \mathbb U$  alors toutes ses sous-matrices principales sont inversibles.



### Corollaire 2.1:



Si  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  est une matrice hermitienne définie positive alors elle admet une unique factorisation  $\mathbb{L}\mathbb{U}$ .

**preuve :** Travail à faire Montrer que si A hermitienne définie positive alors toutes ses sous-matrices principales sont hermitienne définies positives. Puis conclure.

### Remarque 2.1

Si la matrice  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  est inversible mais que ses sous-matrices principales ne sont pas toutes inversibles, il est possible par des permutations préalables de lignes de la matrice de se ramener à une matrice telle que ses sous-matrices principales soient inversibles.

### Théorème 2.1 : Factorisation LU avec permutations



Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  une matrice inversible. Il existe une matrice  $\mathbb{P}$ , produit de matrices de permutation, une matrice  $\mathbb{L} \in \mathcal{M}_n(\mathbb{C})$  triangulaire inférieure à diagonale unité et une matrice  $\mathbb{U} \in \mathcal{M}_n(\mathbb{C})$  triangulaire supérieure inversible telles ques

$$\mathbb{P}\mathbb{A} = \mathbb{L}\mathbb{U}. \tag{5}$$

## Utilisation pratique de la factorisation $\mathbb{L}\mathbb{U}$

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{K})$  admettant une factorisation  $\mathbb{L}\mathbb{U}$ 

Trouver 
$$\mathbf{x} \in \mathbb{K}^n$$
 tel que 
$$\mathbb{A}\mathbf{x} = \mathbf{b} \iff \mathbb{L} \mathbb{U}\mathbf{x} = \mathbf{b}$$
 (6)

est équivalent à

34 / 68

# Utilisation pratique de la factorisation LU

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{K})$  admettant une factorisation  $\mathbb{L}\mathbb{U}$ 

Trouver 
$$\mathbf{x} \in \mathbb{K}^n$$
 tel que

$$\mathbb{A}\mathbf{x} = \mathbf{b} \iff \mathbb{L}\mathbb{U}\mathbf{x} = \mathbf{b} \tag{6}$$

### est équivalent à

Trouver  $\mathbf{x} \in \mathbb{K}^n$  solution de

$$\mathbb{U} x = y$$

avec  $\mathbf{y} \in \mathbb{K}^n$  solution de

$$\mathbf{L}\mathbf{y}=\mathbf{b}.\tag{8}$$

34 / 68

## Algorithme de résolution de systèmes linéaire par LU

**Algorithme** Fonction RSLFactLU permettant de résoudre, par une factorisation  $\mathbb{L}\mathbb{U}$ , le système linéaire  $\mathbb{A}\mathbf{x} = \mathbf{b}$  où  $\mathbb{A}$  est une matrice de  $\mathcal{M}_n(\mathbb{K})$ , dont toutes les sous-matrices principales sont inversibles, et  $\mathbf{b} \in \mathbb{K}^n$ .

```
, Données : \mathbb{A} : matrice de \mathcal{M}_n(\mathbb{K}) dont les sous-matrices
```

principales sont inversibles;

**b** : vecteur de  $\mathbb{K}^n$ .

**Résultat :** x : vecteur de  $\mathbb{K}^n$ .

```
1: Fonction x \leftarrow \text{RSLFactLU}(A, b)
```

2: 
$$[\mathbb{L}, \mathbb{U}] \leftarrow \text{FactLU}(\mathbb{A})$$

⊳ Factorisation LU

3: 
$$\mathbf{y} \leftarrow \text{RSLTriInf}(\mathbb{L}, \mathbf{b})$$

ightharpoonup Résolution du système  $\mathbb{L} {m y} = {m b}$ 

```
4: \mathbf{x} \leftarrow \text{RSLTriSup}(\mathbb{U}, \mathbf{y})
```

ightharpoonup Résolution du système  $\mathbb{U} {m x} = {m y}$ 

5: Fin Fonction

Il nous faut donc écrire la fonction FactLU

2025/10/24

$$\begin{pmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,n} \\ A_{2,1} & A_{2,2} & \dots & A_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n,1} & A_{n,2} & \dots & A_{n,n} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ L_{2,1} & 1 & 0 & \dots & 0 \\ L_{3,1} & L_{3,2} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ L_{n,1} & L_{n,2} & \dots & L_{n,n-1} & 1 \end{pmatrix} \begin{pmatrix} U_{1,1} & U_{1,2} & \dots & \dots & U_{1,n} \\ 0 & U_{2,2} & \dots & \dots & U_{2,n} \\ 0 & 0 & U_{3,3} & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & U_{n,n} \end{pmatrix}$$

On connait △, on cherche L et U

36 / 68

$$\begin{pmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,n} \\ A_{2,1} & A_{2,2} & \dots & A_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n,1} & A_{n,2} & \dots & A_{n,n} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ L_{2,1} & 1 & 0 & \dots & 0 \\ L_{3,1} & L_{3,2} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ L_{n,1} & L_{n,2} & \dots & L_{n,n-1} & 1 \end{pmatrix} \begin{pmatrix} U_{1,1} & U_{1,2} & \dots & \dots & U_{1,n} \\ 0 & U_{2,2} & \dots & \dots & U_{2,n} \\ 0 & 0 & U_{3,3} & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & U_{n,n} \end{pmatrix}$$

- **Etape** 1 :
  - lacktriangle On connaît la **première ligne** de  $\mathbb{L} \Longrightarrow$  on peut calculer la **première ligne** de  $\mathbb{U}$

36 / 68

$$\begin{pmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,n} \\ A_{2,1} & A_{2,2} & \dots & A_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n,1} & A_{n,2} & \dots & A_{n,n} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ L_{2,1} & 1 & 0 & \dots & 0 \\ L_{3,1} & L_{3,2} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ L_{n,1} & L_{n,2} & \dots & L_{n,n-1} & 1 \end{pmatrix} \begin{pmatrix} U_{1,1} & U_{1,2} & \dots & \dots & U_{1,n} \\ 0 & U_{2,2} & \dots & \dots & U_{2,n} \\ 0 & 0 & U_{3,3} & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & U_{n,n} \end{pmatrix}$$

#### • **Etape** 1 :

- lacktriangle On connaît la **première ligne** de  $\mathbb{L} \Longrightarrow$  on peut calculer la **première ligne** de  $\mathbb{U}$
- lacktriangle On connait la **première colonne** de  $\mathbb U \Longrightarrow$  on peut calculer la **première colonne** de  $\mathbb L$

36 / 68

$$\begin{pmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,n} \\ A_{2,1} & A_{2,2} & \dots & A_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n,1} & A_{n,2} & \dots & A_{n,n} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ L_{2,1} & 1 & 0 & \dots & 0 \\ L_{3,1} & L_{3,2} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ L_{n,1} & L_{n,2} & \dots & L_{n,n-1} & 1 \end{pmatrix} \begin{pmatrix} U_{1,1} & U_{1,2} & \dots & \dots & U_{1,n} \\ 0 & U_{2,2} & \dots & \dots & U_{2,n} \\ 0 & 0 & U_{3,3} & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & U_{n,n} \end{pmatrix}$$

#### • **Etape** 1 :

- lacktriangle On connaît la **première ligne** de  $\mathbb{L} \Longrightarrow$  on peut calculer la **première ligne** de  $\mathbb{U}$
- lacktriangle On connait la **première colonne** de  $\mathbb U \Longrightarrow$  on peut calculer la **première colonne** de  $\mathbb L$

### • **Etape** 2 :

▶ On connait la deuxième ligne de  $\mathbb{L}$   $\Longrightarrow$  on peut calculer la deuxième ligne de  $\mathbb{U}$  car on connait la première ligne de  $\mathbb{U}$ 

$$\begin{pmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,n} \\ A_{2,1} & A_{2,2} & \dots & A_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n,1} & A_{n,2} & \dots & A_{n,n} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ L_{2,1} & 1 & 0 & \dots & 0 \\ L_{3,1} & L_{3,2} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ L_{n,1} & L_{n,2} & \dots & L_{n,n-1} & 1 \end{pmatrix} \begin{pmatrix} U_{1,1} & U_{1,2} & \dots & \dots & U_{1,n} \\ 0 & U_{2,2} & \dots & \dots & U_{2,n} \\ 0 & 0 & U_{3,3} & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & U_{n,n} \end{pmatrix}$$

#### Etape 1 :

- lacktriangle On connaît la **première ligne** de  $\mathbb{L} \Longrightarrow$  on peut calculer la **première ligne** de  $\mathbb{U}$
- lacktriangle On connaît la **première colonne** de  $\mathbb U \Longrightarrow$  on peut calculer la **première colonne** de  $\mathbb L$

### • **Etape** 2 :

- On connait la deuxième ligne de  $\mathbb{L} \Longrightarrow$  on peut calculer la deuxième ligne de  $\mathbb{U}$  car on connait la première ligne de  $\mathbb{U}$
- ▶ On connait la deuxième colonne de  $\mathbb{U} \Longrightarrow$  on peut calculer la deuxième colonne de  $\mathbb{L}$  car on connait la première colonne de  $\mathbb{L}$

$$\begin{pmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,n} \\ A_{2,1} & A_{2,2} & \dots & A_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n,1} & A_{n,2} & \dots & A_{n,n} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ L_{2,1} & 1 & 0 & \dots & 0 \\ L_{3,1} & L_{3,2} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ L_{n,1} & L_{n,2} & \dots & L_{n,n-1} & 1 \end{pmatrix} \begin{pmatrix} U_{1,1} & U_{1,2} & \dots & \dots & U_{1,n} \\ 0 & U_{2,2} & \dots & \dots & U_{2,n} \\ 0 & 0 & U_{3,3} & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & U_{n,n} \end{pmatrix}$$

#### Etape 1 :

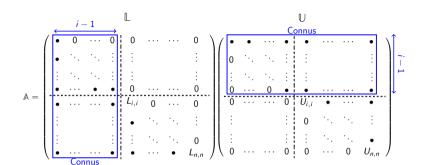
- lacktriangle On connaît la **première ligne** de  $\mathbb{L} \Longrightarrow$  on peut calculer la **première ligne** de  $\mathbb{U}$
- lacktriangle On connaît la **première colonne** de  $\mathbb U \Longrightarrow$  on peut calculer la **première colonne** de  $\mathbb L$

### • **Etape** 2 :

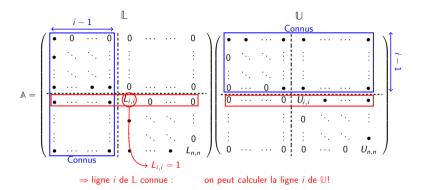
- ▶ On connait la deuxième ligne de  $\mathbb{L}$   $\Longrightarrow$  on peut calculer la deuxième ligne de  $\mathbb{U}$  car on connait la première ligne de  $\mathbb{U}$
- ightharpoonup On connait la **deuxième colonne** de  $\mathbb U$   $\Longrightarrow$  on peut calculer la **deuxième colonne** de  $\mathbb L$  car on connait la première colonne de  $\mathbb L$

• ...

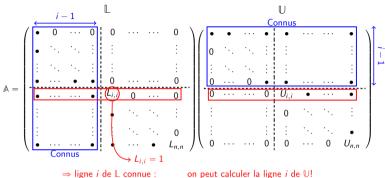
36 / 68



 Méthodes directes
 Factorisation LU
 2025/10/24
 37 / 68



37 / 68



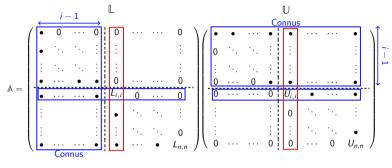
On cherche  $\bigcup_{i,i} \forall j \in [i, n]$ .

$$A_{i,j} \stackrel{\text{def}}{=} \sum_{k=1}^{n} L_{i,k} U_{k,j} = \sum_{k=1}^{i-1} \overbrace{L_{i,k} U_{k,j}}^{\text{connus}} + \overbrace{L_{i,i}}^{-1} U_{i,j} + \sum_{k=i+1}^{n} \overbrace{L_{i,k}}^{=0} U_{k,j}$$

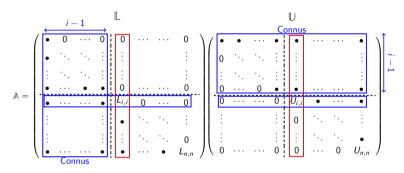
$$U_{i,j} = A_{i,j} - \sum_{k=1}^{i-1} L_{i,k} U_{k,j}, \quad \forall j \in [\![i,n]\!].$$

200

Méthodes directes Factorisation [ ] 37 / 68 2025/10/24



 $\Rightarrow$  colonne *i* de  $\mathbb{U}$  connue : on peut calculer la colonne *i* de  $\mathbb{L}!$ 



 $\Rightarrow$  colonne *i* de  $\mathbb{U}$  connue :

on peut calculer la colonne i de  $\mathbb{L}!$ 

On cherche  $L_{j,i} \ \forall j \in [i+1, n], (L_{i,i} = 1)$ 

$$A_{j,i} \stackrel{\text{def}}{=} \sum_{k=1}^{n} L_{j,k} U_{k,i} = \sum_{k=1}^{i-1} \underbrace{L_{j,k} U_{k,i} + L_{j,i}}_{\text{$L_{j,k}$}} \underbrace{U_{i,i}}_{\text{$U_{i,j}$}} + \sum_{k=i+1}^{n} L_{j,k} \underbrace{U_{k,i}}_{\text{$U_{k,i}$}}$$
$$L_{j,i} = \left(A_{j,i} - \sum_{k=1}^{i-1} L_{j,k} U_{k,i}\right) / U_{i,i}, \quad \forall j \in [\![i+1,n]\!].$$

### En résumé, à l'étape i:

• Calcul de la ligne i de  $\mathbb U$ 

$$U_{i,j} = 0, \quad \forall j \in \llbracket 1, i-1 
rbracket,$$

$$U_{i,j} = A_{i,j} - \sum_{k=1}^{i-1} L_{i,k} U_{k,j}, \quad \forall j \in \llbracket i, n 
rbracket.$$

• Calcul de la colonne i de l

$$L_{j,i} = 0, \quad \forall j \in [1, i-1],$$

$$L_{i,i} = 1,$$

$$L_{j,i} = \left(A_{j,i} - \sum_{k=1}^{i-1} L_{j,k} U_{k,i}\right) / U_{i,i}, \quad \forall j \in [i+1, n].$$

# Algorithme 9 $\boxed{\mathcal{R}_0}$

1. Calculer les matrices L et U

Algorithme 9  $\overline{\mathcal{R}_1}$ 

- 1: Pour  $i \leftarrow 1$  à n faire
- 2: Calculer la ligne i de  $\mathbb{U}$ .
- 3: Calculer la colonne i de  $\mathbb{L}$ .
- 4: Fin Pour

990

### Algorithme 9 $R_1$

- Pour i ← 1 à n faire
- 2: Calculer la ligne i de U.
- $_{3:}$  Calculer la colonne i de  $\mathbb{L}$ .
- 4: Fin Pour

En résumé, à l'étape i:

• Calcul de la ligne i de  $\mathbb U$ 

$$U_{i,j} = 0, \quad \forall j \in [1, i-1],$$
  
 $U_{i,j} = A_{i,j} - \sum_{k=1}^{i-1} L_{i,k} U_{k,j}, \quad \forall j \in [i, n].$ 

• Calcul de la colonne i de  $\mathbb L$ 

$$\begin{array}{ll} L_{j,i} & = 0, & \forall j \in \llbracket 1, i - 1 \rrbracket, \\ L_{i,i} & = 1, \\ L_{j,i} & = \left(A_{j,i} - \sum_{i=1}^{i-1} L_{j,k} U_{k,i}\right) / U_{i,i}, & \forall j \in \llbracket i + 1, n \rrbracket. \end{array}$$

### Algorithme 9 $\boxed{\mathcal{R}_2}$

- 1: Pour  $i \leftarrow 1$  à n faire
- Pour  $j \leftarrow 1$  à i 1 faire  $U(i,j) \leftarrow 0$ Fin Pour
- Pour  $j \leftarrow i$  à n faire
- 6:  $U_{i,j} \leftarrow A_{i,j} \sum_{i=1}^{i-1} L_{i,k} U_{k,j}$ 
  - Fin Pour
- 8: **Pour**  $j \leftarrow 1$  à i 1 **faire** 9:  $L_{i,i} \leftarrow 0$
- 10: Fin Pour
- 11:  $L_{i,i} \leftarrow 1$ 12: Pour  $j \leftarrow i + 1$  à n faire
- 13:  $L_{j,i} \leftarrow \frac{1}{U_{i,i}} \left( A_{j,i} \sum_{k=1}^{i-1} L_{j,k} U_{k,i} \right)$
- 14: Fin Pour
- 15: Fin Pour

Algorithme 9 R2 Algorithme 9  $\mathbb{R}_3$ 1: Pour  $i \leftarrow 1$  à n faire 1: Pour  $i \leftarrow 1$  à n faire Pour  $i \leftarrow 1$  à i - 1 faire Pour  $i \leftarrow 1$  à i - 1 faire  $U(i,j) \leftarrow 0$  $U(i,j) \leftarrow 0$ Fin Pour Fin Pour Pour  $j \leftarrow i$  à n faire Pour  $j \leftarrow i$  à n faire 6: Fin Pour Pour  $j \leftarrow 1$  à i-1 faire Fin Pour  $L_{i,i} \leftarrow 0$ Fin Pour Pour  $i \leftarrow 1$  à i-1 faire 10.  $L_{i,i} \leftarrow 0$  $L_{i,i} \leftarrow 1$ 11: 11: Fin Pour Pour  $j \leftarrow i + 1$  à *n* faire  $L_{i,i} \leftarrow 1$ 12: 13: Pour  $i \leftarrow i + 1$  à n faire 13: Fin Pour 15: Fin Pour Fin Pour 17: Fin Pour

#### Algorithme 9 $\mathbb{R}_3$ Algorithme 9 $\mathbb{R}_4$ 1: Pour $i \leftarrow 1$ à n faire 1: Pour $i \leftarrow 1$ à n faire Pour $i \leftarrow 1$ à i-1 faire Pour $i \leftarrow 1$ à i-1 faire $U(i,i) \leftarrow 0$ $U(i,i) \leftarrow 0$ Fin Pour Fin Pour Pour $i \leftarrow i$ à n faire Pour $j \leftarrow i$ à n faire $S_1 \leftarrow \sum_{i=1}^{i-1} L_{i,k} U_{k,j}$ $S_1 \leftarrow 0$ Pour $k \leftarrow 1$ à i-1 faire $S_1 \leftarrow S_1 + L_{i,k} * U_{k,i}$ $U_{i,i} \leftarrow A_{i,i} - S_1$ Fin Pour Fin Pour Pour $j \leftarrow 1$ à i-1 faire $U_{i,j} \leftarrow A_{i,j} - S_1$ 10: $L_{i,i} \leftarrow 0$ 10: Fin Pour 11: Fin Pour 11: Pour $j \leftarrow 1$ à i-1 faire 12: $L_{i,i} \leftarrow 1$ $L_{i,i} \leftarrow 0$ Pour $j \leftarrow i + 1$ à n faire Fin Pour $L_{i,i} \leftarrow 1$ $S_2 \leftarrow \sum_{k=1}^{i-1} L_{j,k} U_{k,i}$ Pour $i \leftarrow i + 1$ à n faire 14: $L_{j,i} \leftarrow \frac{1}{U_{i,i}} (A_{j,i} - S_2).$ $S_2 \leftarrow 0$ 17: Pour $k \leftarrow 1$ à i-1 faire Fin Pour $S_2 \leftarrow S_2 + L_{i,k} * U_{k,i}$ 19: 17: Fin Pour Fin Pour 20: $L_{j,i} \leftarrow \frac{1}{U_{i,i}} \left( A_{j,i} - S_2 \right).$ Fin Pour 23: Fin Pour

A = 1 U

: matrice de  $\mathcal{M}_n(\mathbb{K})$  dont les sous-matrices principales sont inversibles

**Résultat :**  $\mathbb{L}$  : matrice de  $\mathcal{M}_n(\mathbb{K})$  triangulaire inférieure

```
avec L_{i,i} = 1, \forall i \in [1, n]
                   \mathbb{U}: matrice de \mathcal{M}_n(\mathbb{K}) triangulaire supérieure.
 1: Fonction [\mathbb{L}, \mathbb{U}] \leftarrow \text{FactLU}(\mathbb{A})
        \mathbb{U} \leftarrow \mathbb{O}_n
                                                                                                                                                    \triangleright \mathbb{O}_n matrice nulle n \times n
         \mathbb{L} \leftarrow \mathbb{I}_n

Arr \parallel_n matrice identitée n \times n
         Pour i \leftarrow 1 à n faire
            Pour i \leftarrow i à n faire

ightharpoonup Calcul de la ligne i de U
                S_1 \leftarrow 0
                Pour k \leftarrow 1 à i-1 faire
                   S_1 \leftarrow S_1 + L(i,k) * U(k,i)
               Fin Pour
 Q.
                U(i,j) \leftarrow A(i,j) - S_1
10.
            Fin Pour
11.
            Pour i \leftarrow i + 1 à n faire

    Calcul de la colonne i de 
    I

12:
                S_2 \leftarrow 0
13:
                Pour k \leftarrow 1 à i-1 faire
14:
                    S_2 \leftarrow S_2 + L(i,k) * U(k,i)
15:
                Fin Pour
16:
                L(i, i) \leftarrow (A_{i,i} - S_2) / U(i, i).
17:
            Fin Pour
         Fin Pour
```

20: Fin Fonction

### Plan

- Conditionnement
- Méthodes directes
  - Matrices particulières
    - Matrices diagonales
    - Matrices triangulaires inférieures
    - Matrices triangulaires supérieures
  - Exercices et résultats préliminaires
  - Méthode de Gauss-Jordan
    - Ecriture algébrique

- Factorisation LU
  - Résultats théoriques
  - Utilisation pratique
- Factorisation LDL\*
- Factorisation de Cholesky
  - Résultats théoriques
  - Résolution d'un système linéaire
  - Algorithme : Factorisation positive de Cholesky
- Factorisation QR
  - La tranformation de Householder

40 / 68

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  hermitienne inversible admettant une factorisation  $\mathbb{L}\mathbb{U}$ . On pose

$$\mathbb{D} = \operatorname{diag} \mathbb{U} \text{ et } \mathbb{R} = \mathbb{D}^{-1} \mathbb{U}.$$

R est alors triangulaire supérieure à diagonale unité. On a alors

$$A = LU = LDD^{-1}U = LDR.$$

$$A$$
 hermitienne  $A^* = A \implies A = \mathbb{R}^*(\mathbb{D}^*\mathbb{L}^*) = \mathbb{L}(\mathbb{D}\mathbb{R})$ 

Par unicité de la factorisation  $\mathbb{L}\mathbb{U}$  :

$$\mathbb{R}^* = \mathbb{L} \text{ et } \mathbb{D}^* \mathbb{L}^* = \mathbb{D} \mathbb{R} \implies \mathbb{R}^* = \mathbb{L} \text{ et } \mathbb{D}^* = \mathbb{D}$$

41 / 68

#### Théorème 2.2 : Factorisation | D| \*



Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  une matrice hermitienne inversible admettant une factorisation  $\mathbb{L}\mathbb{U}$ . Alors  $\mathbb{A}$ s'écrit sous la forme

$$\mathbb{A} = \mathbb{LDL}^* \tag{9}$$

où  $\mathbb{D} = \operatorname{diag} \mathbb{U}$  est une matrice à coefficients réels.



### Corollaire 2.2 : Exercice 6,





Une matrice  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  admet une factorisation  $\mathbb{LDL}^*$  avec  $\mathbb{L} \in \mathcal{M}_n(\mathbb{C})$  matrice triangulaire inférieure à diagonale unité et  $\mathbb{D} \in \mathcal{M}_n(\mathbb{R})$  matrice diagonale à coefficients diagonaux strictement positifs si et seulement si la matrice A est hermitienne définie positive.

### Plan

- Conditionnement
- Méthodes directes
  - Matrices particulières
    - Matrices diagonales
    - Matrices triangulaires inférieures
    - Matrices triangulaires supérieures
  - Exercices et résultats préliminaires
  - Méthode de Gauss-Jordan
    - Ecriture algébrique

- Factorisation LU
  - Résultats théoriques
  - Utilisation pratique
- Factorisation LDL\*
- Factorisation de Cholesky
  - Résultats théoriques
  - Résolution d'un système linéaire
  - Algorithme : Factorisation positive de Cholesky
- Factorisation QR
  - La tranformation de Householder



### **Definition 2.1**

Une factorisation régulière de Cholesky d'une matrice  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  est une factorisation  $\mathbb{A} =$  $\mathbb{BB}^*$  où  $\mathbb{B}$  est une matrice triangulaire inférieure inversible.

Si les coefficients diagonaux de B sont positifs, on parle alors d'une factorisation positive de Cholesky.

## Théorème 2.3 : Factorisation de Cholesky, Exercice 7





La matrice  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  admet une factorisation régulière de Cholesky si et seulement si la matrice A est hermitienne définie positive. Dans ce cas, elle admet une unique factorisation positive.

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  hermitienne définie positive et  $\boldsymbol{b} \in \mathbb{C}^n$ . On note  $\mathbb{B}$  la matrice de factorisation positive de Cholesky de  $\mathbb{A}$ .

Trouver 
$$\mathbf{x} \in \mathbb{C}^n$$
 tel que 
$$\mathbb{A}\mathbf{x} = \mathbf{b} \ (\iff \mathbb{BB}^*\mathbf{x} = \mathbf{b}) \tag{10}$$

est équivalent à

45 / 68

Méthodes directes Factorisation de Cholesky 2025/10/24

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  hermitienne définie positive et  $\boldsymbol{b} \in \mathbb{C}^n$ . On note  $\mathbb{B}$  la matrice de factorisation positive de Cholesky de  $\mathbb{A}$ .

Trouver 
$$\mathbf{x} \in \mathbb{C}^n$$
 tel que 
$$\mathbb{A}\mathbf{x} = \mathbf{b} \ (\iff \mathbb{BB}^*\mathbf{x} = \mathbf{b}) \tag{10}$$

### est équivalent à

Trouver  $\mathbf{x} \in \mathbb{C}^n$  solution de  $\mathbb{B}^*\mathbf{x} = \mathbf{y} \tag{11}$  avec  $\mathbf{y} \in \mathbb{C}^n$  solution de  $\mathbb{B}\mathbf{y} = \mathbf{b}. \tag{12}$ 

45 / 68

Méthodes directes Factorisation de Cholesky 2025/10/24

## Algorithme de résolution de systèmes linéaire par Cholesky

**Algorithme** Fonction RSLCholesky permettant de résoudre, par une factorisation de Cholesky positive, le système linéaire

$$Ax = b$$

où  $\mathbb{A}$  une matrice hermitienne de  $\mathcal{M}_n(\mathbb{C})$  définie positive et  $\boldsymbol{b} \in \mathbb{C}^n$ .

**Données :**  $\mathbb{A}$  : matrice de  $\mathcal{M}_n(\mathbb{C})$  hermitienne définie positive,

**b** : vecteur de  $\mathbb{C}^n$ .

**Résultat** :  $\mathbf{x}$  : vecteur de  $\mathbb{C}^n$ .

- 1: Fonction  $x \leftarrow \text{RSLCholesky}(A, b)$
- 2:  $\mathbb{B} \leftarrow \text{Cholesky}(\mathbb{A})$
- 3:  $\mathbf{y} \leftarrow \text{RSLTriInf}(\mathbb{B}, \mathbf{b})$
- 4:  $\mathbb{U} \leftarrow \text{MatAdjointe}(\mathbb{B})$
- 5:  $\mathbf{x} \leftarrow \text{RSLTriSup}(\mathbb{U}, \mathbf{y})$
- 6: Fin Fonction

- ightharpoonup Factorisation positive de Cholesky
- ightharpoonup Résolution du système  $\mathbb{B} oldsymbol{y} = oldsymbol{b}$
- ightharpoonup Calcul de la matrice adjointe de  $\mathbb B$ 
  - ightharpoonup Résolution du système  $\mathbb{B}^* \mathbf{x} = \mathbf{y}$

Il nous faut donc écrire la fonction Cholesky

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  hermitienne définie positive. il existe une unique matrice  $\mathbb{B} \in \mathcal{M}_n(\mathbb{C})$  triangulaire inférieure avec  $B_{i,j} \in \mathbb{R}^{+*}, \ \forall i \in [1,n]$ , telle que

$$A = BB^*$$

c'est à dire

$$\begin{pmatrix} A_{1,1} & \dots & A_{1,n} \\ \vdots & \ddots & \dots \\ A_{n,1} & \dots & A_{n,n} \end{pmatrix} = \begin{pmatrix} B_{1,1} & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ B_{n,1} & \dots & \dots & B_{n,n} \end{pmatrix} \begin{pmatrix} B_{1,1} & \dots & \dots & B_{n,1} \\ 0 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & \overline{B_{n,n}} \end{pmatrix}.$$

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  hermitienne définie positive. il existe une unique matrice  $\mathbb{B} \in \mathcal{M}_n(\mathbb{C})$  triangulaire inférieure avec  $B_{i,i} \in \mathbb{R}^{+*}, \ \forall i \in [1, n], \ \text{telle que}$ 

$$A = BB^*$$

c'est à dire

$$\begin{pmatrix} \mathbf{A}_{1,1} & \dots & \mathbf{A}_{1,n} \\ \vdots & \ddots & \dots \\ \mathbf{A}_{n,1} & \dots & \mathbf{A}_{n,n} \end{pmatrix} = \begin{pmatrix} \mathbf{B}_{1,1} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \mathbf{0} \\ \mathbf{B}_{n,1} & \dots & \dots & \mathbf{B}_{n,n} \end{pmatrix} \begin{pmatrix} \mathbf{B}_{1,1} & \dots & \dots & \mathbf{B}_{n,1} \\ \mathbf{0} & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{B}_{n,n} \end{pmatrix}.$$

Calcul de B<sub>1,1</sub> (la 1ère ligne de B est donc déterminée)
 ⇒ calcul 1ère colonne de B.

47 / 68

Méthodes directes Factorisation de Cholesky 2025/10/24

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  hermitienne définie positive. il existe une unique matrice  $\mathbb{B} \in \mathcal{M}_n(\mathbb{C})$  triangulaire inférieure avec  $B_{i,j} \in \mathbb{R}^{+*}, \ \forall i \in [1,n]$ , telle que

$$A = BB^*$$

c'est à dire

$$\begin{pmatrix} A_{1,1} & \dots & A_{1,n} \\ \vdots & \ddots & \ddots \\ A_{n,1} & \dots & A_{n,n} \end{pmatrix} = \begin{pmatrix} B_{1,1} & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ B_{n,1} & \dots & \dots & B_{n,n} \end{pmatrix} \begin{pmatrix} \overline{B_{1,1}} & \dots & \dots & \overline{B_{n,1}} \\ 0 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & \overline{B_{n,n}} \end{pmatrix}.$$

- Calcul de B<sub>1,1</sub> (la 1ère ligne de B est donc déterminée)
   ⇒ calcul 1ère colonne de B.
- Puis calcul de  $B_{2,2}$  (la 2ème ligne de  $\mathbb B$  est donc déterminée)  $\Longrightarrow$  calcul 2ème colonne de  $\mathbb B$ .
- Etc...

Soit  $i \in [1, n]$ . On suppose connues les i - 1 premières colonnes de  $\mathbb{B}$ .

Peut-on calculer la colonne i de  $\mathbb{B}$ ?

$$\mathbb{A} = \mathbb{BB}^* \implies A_{i,i} = \sum_{k=1}^n B_{i,k}(\mathbb{B}^*)_{k,i} = \sum_{k=1}^n B_{i,k}\overline{B_{i,k}}$$

Or  $\mathbb{B}$  triangulaire inférieure (i.e.  $B_{i,j} = 0$  si j > i)

$$A_{i,i} = \sum_{k=1}^{i-1} |B_{i,k}|^2 + |B_{i,i}|^2$$

et donc

$$\mathbf{B}_{i,i} = \left(\mathbf{A}_{i,i} - \sum_{i=1}^{i-1} |\mathbf{B}_{i,j}|^2\right)^{1/2}.$$

48 / 68

Méthodes directes Factorisation de Cholesky 2025/10/24

Il reste à déterminer  $B_{j,i}, \forall j \in [i+1, n]$ .

$$\mathbf{A}_{j,i} = \sum_{k=1}^{n} \mathbf{B}_{j,k} (\mathbb{B}^*)_{k,i} = \sum_{k=1}^{n} \mathbf{B}_{j,k} \overline{\mathbf{B}_{i,k}}, \ \forall j \in \llbracket i+1, n \rrbracket$$

Comme L est triangulaire inférieure on obtient

$$\mathbf{A}_{j,i} = \sum_{k=1}^{i} \mathbf{B}_{j,k} \overline{\mathbf{B}_{i,k}} = \sum_{k=1}^{i-1} \mathbf{B}_{j,k} \overline{\mathbf{B}_{i,k}} + \mathbf{B}_{j,i} \overline{\mathbf{B}_{i,i}}, \ \forall j \in \llbracket i+1, n \rrbracket$$

Or  $B_{i,i} > 0$  connu et les i-1 premières colonnes de  $\mathbb B$  aussi.

$$\begin{aligned} \mathbf{B}_{j,i} &=& \frac{1}{\mathbf{B}_{i,i}} \left( \mathbf{A}_{j,i} - \sum_{k=1}^{i-1} \mathbf{B}_{j,k} \overline{\mathbf{B}_{i,k}} \right), \ \forall j \in \llbracket i+1,n \rrbracket \\ \mathbf{B}_{i,i} &=& \mathbf{0}, \ \forall j \in \llbracket 1,i-1 \rrbracket. \end{aligned}$$

Méthodes directes Factorisation de Cholesky 2025/10/24 49 / 68

Algorithme 11  $\mathcal{R}_0$ 

 $_{1:}$  Calculer la matrice  ${\mathbb B}$ 

Algorithme 11  $R_1$ 

- 1: Pour  $i \leftarrow 1$  à n faire
- 2: Calculer  $B_{i,i}$ , connaissant les i-1 premières colonnes de  $\mathbb{B}$ .
- 3: Calculer la  $i^{\text{ème}}$  colonne de  $\mathbb{B}$ .
- 4: Fin Pour

## Algorithme 11 $\mathbb{R}_1$

- 1: Pour  $i \leftarrow 1$  à n faire
- Calculer  $B_{i,i}$ , connaissant les i-1 premières colonnes de  $\mathbb{B}$ .
- 2: Calculer la  $i^{\text{ème}}$  colonne de  $\mathbb{B}$ .
- 4: Fin Pour

**Etape** i: les i-1 premières colonnes de  $\mathbb B$  étant connues, on peut calculer la i-ème colonne:

$$\begin{array}{lcl} \mathbf{B}_{i,i} & = & \left(\mathbf{A}_{i,i} - \sum_{j=1}^{i-1} |\mathbf{B}_{i,j}|^2\right)^{1/2} > 0 \\ \\ \mathbf{B}_{j,i} & = & \frac{1}{\mathbf{B}_{i,i}} \left(\mathbf{A}_{j,i} - \sum_{k=1}^{i-1} \mathbf{B}_{j,k} \overline{\mathbf{B}_{i,k}}\right), \ \forall j \in [\![i+1,n]\!] \\ \\ \mathbf{B}_{i,i} & = & 0, \ \forall i \in [\![1,i-1]\!]. \end{array}$$

## Algorithme 11 $R_2$

1: Pour  $i \leftarrow 1$  à n faire

2. 
$$\left\{ B_{i,i} \leftarrow \left( A_{i,i} - \sum_{j=1}^{i-1} |B_{i,j}|^2 \right)^{1/2} \right.$$

- Pour  $j \leftarrow 1$  à i-1 faire
- 4:  $B_{j,i} \leftarrow 0$
- 5: Fin Pour
  - Pour  $j \leftarrow i + 1$  à n faire
- 7:  $\mathbf{B}_{j,i} \leftarrow \frac{1}{\mathbf{B}_{i,i}} \left( \mathbf{A}_{j,i} \sum_{k=1}^{i-1} \mathbf{B}_{j,k} \overline{\mathbf{B}_{i,k}} \right).$ 
  - Fin Pour
- 9: Fin Pour

50 / 68

Méthodes directes Factorisation de Cholesky 2025/10/24

### Algorithme 11 $\mathcal{R}_2$

1: Pour  $i \leftarrow 1$  à n faire

$$\mathbb{B}_{i,i} \leftarrow \left(\mathbf{A}_{i,i} - \sum_{j=1}^{i-1} |\mathbf{B}_{i,j}|^2\right)^{1/2}$$

- 2:
  - Pour  $j \leftarrow 1$  à i-1 faire
- $B_{i,i} \leftarrow 0$
- Fin Pour
- Pour  $j \leftarrow i + 1$  à n faire

$$\left[\mathbf{B}_{j,i} \leftarrow \frac{1}{\mathbf{B}_{i,i}} \left(\mathbf{A}_{j,i} - \sum_{k=1}^{i-1} \mathbf{B}_{j,k} \overline{\mathbf{B}_{i,k}}\right).\right]$$

- Fin Pour
- 9: Fin Pour

# Algorithme 11 $\mathcal{R}_3$

1. Pour  $i \leftarrow 1$  à n faire

$$\begin{array}{c}
2: \\
\longrightarrow \\
3: \\
B_{i,i} \leftarrow (A_{i,i} - S_1)^{1/2}
\end{array}$$

- Pour  $i \leftarrow 1$  à i-1 faire
- $B_{i,i} \leftarrow 0$ Fin Pour
- Pour  $j \leftarrow i + 1$  à n faire

$$S_2 \leftarrow \sum_{k=1}^{i-1} \mathbf{B}_{j,k} \overline{\mathbf{B}_{i,k}}$$

- Fin Pour
- 11: Fin Pour

### Algorithme 11 $\mathcal{R}_3$

1: Pour  $i \leftarrow 1$  à n faire

$$S_1 \leftarrow \sum_{j=1}^{i-1} |\mathbf{B}_{i,j}|^2$$

- 2.  $B_{i,i} \leftarrow (A_{i,i} - S_1)^{1/2}$
- Pour  $i \leftarrow 1$  à i 1 faire
- $B_{i,i} \leftarrow 0$
- Fin Pour
- Pour  $i \leftarrow i + 1$  à n faire

$$S_2 \leftarrow \sum_{k=1}^{i-1} \mathbf{B}_{j,k} \overline{\mathbf{B}_{i,k}}$$

- 8:  $B_{j,i} \leftarrow \frac{1}{B_{i,i}} \left( A_{j,i} - S_2 \right).$
- Fin Pour
- 11: Fin Pour

# Algorithme 11 $\mathcal{R}_4$

- 1: Pour  $i \leftarrow 1$  à n faire
- 2:  $S_1 \leftarrow 0$ 3: **Pour**  $j \leftarrow 1$  à i-1 faire
- $S_1 \leftarrow S_1 + |\mathbf{B}_{i,i}|^2$
- Fin Pour
- 6:  $B_{i,i} \leftarrow (A_{i,i} S_1)^{1/2}$
- 7: Pour  $i \leftarrow 1$  à i 1 faire
- 8:  $B_{i,i} \leftarrow 0$
- 9: Fin Pour
- Pour  $i \leftarrow i + 1$  à n faire

$$S_2 \leftarrow 0$$

- Pour  $k \leftarrow 1$  à i-1 faire
  - $S_2 \leftarrow S_2 + B_{i,k} \overline{B_{i,k}}$
- Fin Pour 14:

15: 
$$B_{j,i} \leftarrow \frac{1}{B_{i,i}} (A_{j,i} - S_2).$$

- Fin Pour
- 17: Fin Pour

Algorithme Fonction Cholesky permettant de calculer la matrice  $\mathbb{B}$ , dites matrice de factorisation positive de Cholesky associée à la matrice  $\mathbb{A}$ , telle que  $\mathbb{A} = \mathbb{B}\mathbb{B}^*$ .

```
Données : A : matrice de \mathcal{M}_n(\mathbb{C}) hermitienne définie positive.
Résultat :
               B : matrice de \mathcal{M}_n(\mathbb{C}) triangulaire inférieure
                           avec B(i, i) > 0, \forall i \in [1, n]
 1: Fonction B \leftarrow Cholesky(A)
        Pour i \leftarrow 1 à n faire
          S_1 \leftarrow 0
  3:
           Pour j \leftarrow 1 à i-1 faire
  5:
              S_1 \leftarrow S_1 + |B(i, i)|^2
           Fin Pour
           B(i, i) \leftarrow \operatorname{sqrt}(A(i, i) - S_1)
  7.
           Pour i \leftarrow 1 à i - 1 faire
  8:
              B(i,i) \leftarrow 0
           Fin Pour
10:
           Pour j \leftarrow i + 1 à n faire
11:
              S_2 \leftarrow 0
12:
              Pour k \leftarrow 1 à i-1 faire
13:
                 S_2 \leftarrow S_2 + \mathrm{B}(i,k) * \overline{\mathrm{B}(i,k)}
14:
              Fin Pour
15:
              B(i,i) \leftarrow (A(i,i) - S_2)/B(i,i).
16:
           Fin Pour
17:
        Fin Pour
10. Fin Fonction
```

Exercice Travail à faire

Proposer une méthode permettant de tester la fonction Cholesky.

52 / 68

Méthodes directes Factorisation de Cholesky 2025/10/24

## Plan

- Conditionnement
- Méthodes directes
  - Matrices particulières
    - Matrices diagonales
    - Matrices triangulaires inférieures
    - Matrices triangulaires supérieures
  - Exercices et résultats préliminaires
  - Méthode de Gauss-Jordan
    - Ecriture algébrique

- Factorisation LU
  - Résultats théoriques
  - Utilisation pratique
- Factorisation LDL\*
- Factorisation de Cholesky
  - Résultats théorique
  - Résolution d'un système linéaire
  - Algorithme : Factorisation positive de Cholesky
- $\circ$  Factorisation  $\mathbb{QR}$ 
  - La tranformation de Householder

53 / 68

Méthodes directes Factorisation QR 2025/10/24



#### Definition : Matrice élémentaire de Householder

Soit  $\mathbf{u} \in \mathbb{C}^n$  tel que  $\|\mathbf{u}\|_2 = 1$ . On appelle matrice élémentaire de Householder la matrice  $\mathbb{H}(\mathbf{u}) \in \mathcal{M}_n(\mathbb{C})$  définie par

$$\mathbb{H}(\mathbf{u}) = \mathbb{I} - 2\mathbf{u}\mathbf{u}^*. \tag{13}$$

## Exercice 8 🧆

Soit  $\mathbf{u} \in \mathbb{C}^n$  tel que  $\|\mathbf{u}\|_2 = 1$ . On note  $\mathbb{H} \in \mathcal{M}_n(\mathbb{C})$  la matrice définie par

$$\mathbb{H} = \mathbb{I} - 2\boldsymbol{u}\boldsymbol{u}^*.$$

Q. 1

- Montrer que ℍ est hermitienne.
- 2 Montrer que ℍ est unitaire.

Soit  $\mathbf{x} \in \mathbb{K}^n$ . On note  $\mathbf{x}_{\parallel} = \operatorname{proj}_{\mathbf{u}}(\mathbf{x}) \stackrel{\text{def}}{=} \langle \mathbf{u}, \mathbf{x} \rangle \mathbf{u}$  et  $\mathbf{x}_{\perp} = \mathbf{x} - \mathbf{x}_{\parallel}$ .

Q. 2

Montrer que

$$\mathbb{H}(\mathbf{x}_{\perp} + \mathbf{x}_{\parallel}) = \mathbf{x}_{\perp} - \mathbf{x}_{\parallel}.$$

et

$$\mathbb{H}\mathbf{x} = \mathbf{x}, \quad si \langle \mathbf{x}, \mathbf{u} \rangle = 0.$$

### On vient de démontrer:



## Propriété 2.1

Toute matrice élémentaire de Householder est hermitienne et unitaire.



## Propriété 2.2

Soient  $\mathbf{x} \in \mathbb{K}^n$  et  $\mathbf{u} \in \mathbb{K}^n$ ,  $\|\mathbf{u}\|_2 = 1$ . On note  $\mathbf{x}_{\parallel} = \mathrm{proj}_{\mathbf{u}}(\mathbf{x}) \stackrel{\mathsf{def}}{=} \langle \mathbf{u}, \mathbf{x} \rangle \mathbf{u}$  et  $\mathbf{x}_{\perp} = \mathbf{x} - \mathbf{x}_{\parallel}$ . On a alors

$$\mathbb{H}(\boldsymbol{u})(\boldsymbol{x}_{\perp} + \boldsymbol{x}_{\parallel}) = \boldsymbol{x}_{\perp} - \boldsymbol{x}_{\parallel}. \tag{14}$$

et

$$\mathbb{H}(\boldsymbol{u})\boldsymbol{x} = \boldsymbol{x}, \quad \text{si } \langle \boldsymbol{x}, \boldsymbol{u} \rangle = 0. \tag{15}$$

Méthodes directes Factorisation QR 2025/10/24 55 / 68

### Exercice 9 🦀

Soient  ${\pmb a}$  et  ${\pmb b}$  deux vecteurs non colinéaires de  ${\mathbb C}^n$  avec  $\|{\pmb b}\|_2=1$ . On va chercher  $\alpha\in{\mathbb C}$  et  ${\pmb u}\in{\mathbb C}^n$ ,  $\|{\pmb u}\|_2=1$ , vérifiant

$$\mathbb{H}(\boldsymbol{u})\boldsymbol{a} = \alpha \boldsymbol{b}. \tag{1}$$

Montrer que si  $\alpha$  et **u** vérifient (1) alors

$$|\alpha| = \|\mathbf{a}\|_2 \tag{2}$$

$$\mathbf{a} - 2\langle \mathbf{u}, \mathbf{a} \rangle \mathbf{u} = \alpha \mathbf{b}$$

on en déduit que

$$|\langle \mathbf{u}, \mathbf{a} \rangle|^2 = \frac{\langle \mathbf{a}, \mathbf{a} \rangle - \alpha \langle \mathbf{a}, \mathbf{b} \rangle}{2} \tag{4}$$

Nous allons maintenant établir une condition pour que (4) ait un sens.

- On suppose que arg  $\alpha = -\arg(\langle \mathbf{a}, \mathbf{b} \rangle) [\pi]$ 
  - 1 Montrer que  $\alpha \langle \mathbf{a}, \mathbf{b} \rangle \in \mathbb{R}$ .
  - 2 Montrer que  $\langle \mathbf{a}, \mathbf{a} \rangle \alpha \langle \mathbf{a}, \mathbf{b} \rangle \in \mathbb{R}^{*+}$ .
- Soient  $\alpha$  et  $\boldsymbol{u}$  vérifiant (1). En déduire que si  $\arg \alpha = -\arg(\langle \boldsymbol{a}, \boldsymbol{b} \rangle)$   $[\pi]$  alors  $\boldsymbol{u}$  est donné par

$$u = \frac{1}{2\langle u, a \rangle} (a - \alpha b). \tag{5}$$

 $et \|u\|_2 = 1.$ 

(3)

On vient de démontrer le résultat suivant

#### Théorème 2.4

Soient **a**, **b** deux vecteurs non nuls et non colinéaires de  $\mathbb{C}^n$  avec  $\|\mathbf{b}\|_2 = 1$ . Soit  $\alpha \in \mathbb{C}$  tel que  $|\alpha| = \|\mathbf{a}\|_2$  et  $\arg \alpha = -\arg \langle \boldsymbol{a}, \boldsymbol{b} \rangle [\pi].$  On a alors

$$\mathbb{H}\left(\frac{\mathbf{a} - \alpha \mathbf{b}}{\|\mathbf{a} - \alpha \mathbf{b}\|_{2}}\right) \mathbf{a} = \alpha \mathbf{b}.$$
 (16)



### Corollaire 2.3

Soit  $\mathbf{a} \in \mathbb{C}^n$  non nul et non colinéaire à  $\mathbf{e}_1$ , premier vecteur de la base canonique de  $\mathbb{C}^n$ . Alors, le vecteur  $\mathbf{u} \in \mathbb{C}^n$  donné par

$$oldsymbol{u} = rac{oldsymbol{a} + \left\|oldsymbol{a}
ight\|_2 e^{\imath \operatorname{arg}(oldsymbol{a_1})} oldsymbol{e}_1}{\left\|oldsymbol{a} + \left\|oldsymbol{a}
ight\|_2 e^{\imath \operatorname{arg}(oldsymbol{a_1})} oldsymbol{e}_1
ight\|_2}$$

est bien défini et on a

$$\mathbb{H}(\boldsymbol{u})\boldsymbol{a} = -\|\boldsymbol{a}\|_{2} e^{i \operatorname{arg}(\boldsymbol{a}_{1})} \boldsymbol{e}_{1}. \tag{17}$$







Méthodes directes Factorisation OR 2025/10/24 57 / 68

### Exercice 10 🦓

Soient **a** et **b** deux vecteurs de  $\mathbb{C}^n$  avec  $\|\mathbf{b}\|_2 = 1$ .

- Q. 1
- Soit  $\alpha \in \mathbb{C}$  tel que  $|\alpha| = \|\mathbf{a}\|_2$  et  $\arg(\alpha) = -\arg(\langle \mathbf{a}, \mathbf{b} \rangle) + \delta \pi$  avec  $\delta \in [0, 1]$ .
- ① On suppose que  $\mathbf{a} = \lambda \mathbf{b}$ ,  $\lambda \in \mathbb{C}^*$ , (i.e.  $\mathbf{a}$  et  $\mathbf{b}$  colinéaires). Exprimer  $\mathbf{a} \alpha \mathbf{b}$  en fonction de  $\lambda$  et  $\mathbf{b}$ .
- 2 Que peut-on dire si a est nul?
- Q. 2
  - Ecrire la fonction algorithmique Householder de paramètres  $\mathbf{a}$ ,  $\mathbf{b}$  et  $\delta \in [\![0,1]\!]$  retournant une matrice  $\mathbb{S} \in \mathcal{M}_n(\mathbb{C})$  unitaire et  $\alpha \in \mathbb{C}$  telles que
    - si  $\mathbf{a} \langle \mathbf{b}, \mathbf{a} \rangle \mathbf{b} = 0$  (i.e.  $\mathbf{a}$  nul ou colinéaire à  $\mathbf{b}$ ) alors  $\mathbb S$  est la matrice identitée et  $\alpha = 0$ ,
    - sinon  $\alpha$  est le nombre complexe défini en Q. 1 (dépendant de  $\delta$ ) et  $\mathbb S$  est la matrice élémentaire de Householder

$$\mathbb{S} = \mathbb{H}\left(\frac{\mathbf{a} - \alpha \mathbf{b}}{\left\|\mathbf{a} - \alpha \mathbf{b}\right\|_2}\right)$$

telle que  $Sa = \alpha b$ .

Des fonctions comme dot(a, b) (produit scalaire de deux vecteurs), norm(a) (norme 2 d'un vecteur), arg(z) (argument d'un nombre complexe), eve(n) (matrice identitée de  $\mathcal{M}_n(\mathbb{C})$ ),  $matprod(\mathbb{A}, \mathbb{B})$  (produit de deux matrices),  $ctranspose(\mathbb{A})$  (adjoint d'une matrice), ... pourront être utilisées

Q. 3

Proposer un programme permettant de tester cette fonction. On pourra utiliser la fonction vecrand(n) retournant un vecteur aléatoire de  $\mathbb{C}^n$ , les parties réelles et imaginaires de chacune de ses composantes étant dans ]0,1[ (loi uniforme).

Q. 4

Proposer un programme permettant de vérifier que  $\delta=1$  est le "meilleur" choix.

#### **Algorithme** fonction $[S, \alpha] \leftarrow \text{Householder}(\boldsymbol{a}, \boldsymbol{b}, \delta)$ .

Retourne une matrice  $\mathbb{S} \in \mathcal{M}_n(\mathbb{C})$  unitaire et  $\alpha \in \mathbb{C}$  telles que

- si  $\mathbf{a} \langle \mathbf{b}, \mathbf{a} \rangle \mathbf{b} = 0$  (i.e.  $\mathbf{a}$  nul ou colinéaire à  $\mathbf{b}$ ) alors  $\mathbb{S}$  est la matrice identitée et  $\alpha = 0$ ,
- sinon  $\alpha$  est le nombre complexe défini par

$$|\alpha| = \|\mathbf{a}\|_2$$
 et  $\arg(\alpha) = -\arg(\langle \mathbf{a}, \mathbf{b} \rangle) + \delta\pi$ ,

et,  $\mathbb S$  est la matrice élémentaire de Householder

$$\mathbb{S} = \mathbb{H}\left(\frac{\mathbf{a} - \alpha \mathbf{b}}{\|\mathbf{a} - \alpha \mathbf{b}\|_2}\right)$$

telle que  $Sa = \alpha b$ .

**Données :** a, b : deux vecteurs de  $\mathbb{C}^n$  non nuls et non colinéaires.

 $\delta$  : 0 ou 1, permet de déterminer  $\alpha$ .

**Résultat :**  $\mathbb{S}$  : matrice de Householder ou indentité dans  $\mathcal{M}_n(\mathbb{C})$ ,

 $\alpha$ : nombre complexe, de module  $\|\mathbf{a}\|_2$  et d'argument  $-\arg(\langle \mathbf{a}, \mathbf{b} \rangle) + \delta \pi$ .

- 1: Fonction  $[\mathbb{S}, \alpha] \leftarrow \text{Householder}(\mathbf{a}, \mathbf{b}, \delta)$
- 2: ba  $\leftarrow \det(\boldsymbol{b}, \boldsymbol{a})$
- 3: **Si** norm(a ba \* b) < 1e 15 alors
- 4:  $\mathbb{S} \leftarrow \text{eye}(n), \ \alpha \leftarrow 0$
- 5: Sinon
- 6:  $\alpha \leftarrow \text{norm}(\mathbf{a}) * \exp(i * (\delta * \pi + \arg(\text{ba})))$
- 7:  $\mathbf{u} \leftarrow \mathbf{a} \alpha * \mathbf{b}$
- 8:  $\boldsymbol{u} \leftarrow \boldsymbol{u}/\text{norm}(\boldsymbol{u})$
- 9:  $\mathbb{S} \leftarrow \text{eye}(n) 2 * \text{matprod}(\boldsymbol{u}, \text{ctranspose}(\boldsymbol{u}))$
- 10: Fin Si
- 11: Fin Fonction



 $ightharpoonup dot(b, a) : b^*a$ 

Programme algorithmique vérifiant la fonction Householder:

```
    n ← 100
    a ← vecrand(n)
    b ← vecrand(n)
    b ← b/norm(b, 2)
    [ℍ, α] ← Householder(a, b, 0)
    error ← norm(ℍ * a − α * b, 2)
```

 $\operatorname{vecrand}(n)$  retourne un vecteur aléatoire de  $\mathbb{C}^n$  (loi uniforme [0,1] sur parties réelles et imaginaires)

```
1: n \leftarrow 100
```

2: 
$$\mathbf{a} \leftarrow \operatorname{vecrand}(n)$$

3: 
$$\mathbf{b} \leftarrow \mathbf{a} + \text{tol} * \text{vecrand}(n)$$

4: 
$$\boldsymbol{b} \leftarrow \boldsymbol{b}/\text{norm}(\boldsymbol{b},2)$$

5: 
$$[\mathbb{H}_1, \alpha_1] \leftarrow \text{Householder}(\boldsymbol{a}, \boldsymbol{b}, 1)$$

6: 
$$[\mathbb{H}_0, \alpha_0] \leftarrow \text{Householder}(\boldsymbol{a}, \boldsymbol{b}, 0)$$

7: error0 
$$\leftarrow$$
 norm( $\mathbb{H}_0 * \boldsymbol{a} - \alpha_0 * \boldsymbol{b}, 2$ )/(1 + abs( $\alpha_0$ ))

8: 
$$\operatorname{error1} \leftarrow \operatorname{norm}(\mathbb{H}_1 * \boldsymbol{a} - \alpha_1 * \boldsymbol{b}, 2) / (1 + \operatorname{abs}(\alpha_1))$$

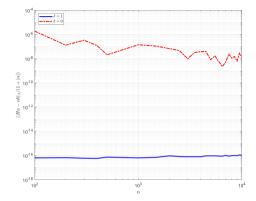


Figure: Choix de  $\delta$  dans Householder : erreur relative en norme  $L_2$  avec tol = 1e - 12



Meilleur choix:  $\delta = 1$ .

Méthodes directes Factorisation OR 61 / 68 2025/10/24

 $(\mathcal{P}_n)$ 

Soit  $\mathbb{A}_n \in \mathcal{M}_n(\mathbb{C})$  une matrice. Il existe une matrice unitaire  $\mathbb{U}_n \in \mathcal{M}_n(\mathbb{C})$  et une matrice triangulaire supérieure  $\mathbb{R}_n \in \mathcal{M}_n(\mathbb{C})$  telles que

$$\mathbb{U}_n \mathbb{A}_n = \mathbb{R}_n. \tag{1}$$

Démontrer par récurrence que  $\forall n \in \mathbb{N}, n \geq 2, (\mathcal{P}_n)$  est vraie.

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$ . Montrer au'il existe une matrice unitaire  $\mathbb{Q} \in \mathcal{M}_n(\mathbb{C})$  et une matrice triangulaire supérieure  $\mathbb{R} \in \mathcal{M}_n(\mathbb{C})$  telles que

$$A = \mathbb{QR}$$
.

 $(Q_n)$ 

Soit  $\mathbb{A}_n \in \mathcal{M}_n(\mathbb{R})$  une matrice. Il existe une matrice orthogonale  $\mathbb{U}_n \in \mathcal{M}_n(\mathbb{R})$  et une matrice triangulaire supérieure  $\mathbb{R}_n \in \mathcal{M}_n(\mathbb{R})$  telles que

$$\mathbb{U}_n \mathbb{A}_n = \mathbb{R}_n. \tag{2}$$

Q. 3

La proposition  $(Q_n)$  est-elle vérifiée pour tout  $n \ge 2$ ? Justifier.

Q. 4

Soit  $A \in \mathcal{M}_n(\mathbb{R})$ .

 $\square$  Montrer qu'il existe une matrice orthogonale  $\mathbb{Q} \in \mathcal{M}_n(\mathbb{R})$  et une matrice triangulaire supérieure  $\mathbb{R} \in \mathcal{M}_n(\mathbb{R})$  telles que

$$\mathbb{A}=\mathbb{Q}\mathbb{R}.$$

② Montrer qu'il existe une matrice orthogonale  $\mathbb{Q} \in \mathcal{M}_n(\mathbb{R})$  et une matrice triangulaire supérieure  $\mathbb{R} \in \mathcal{M}_n(\mathbb{R})$  à coefficient diagonaux positifs ou nuls telles que

$$A = \mathbb{OR}$$

 $\mathfrak{I}$  On suppose  $\mathbb{A}$  inversible. Montrer qu'il existe une unique matrice orthogonale  $\mathbb{Q} \in \mathcal{M}_n(\mathbb{R})$  et une unique matrice triangulaire supérieure  $\mathbb{R} \in \mathcal{M}_n(\mathbb{R})$  à coefficient diagonaux strictement positifs telles que

Factorisation OR

 $A = \mathbb{OR}$ .

### Théorème : Factorisation $\mathbb{QR}$



Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$  une matrice. Il existe une matrice unitaire  $\mathbb{Q} \in \mathcal{M}_n(\mathbb{C})$  et une matrice triangulaire supérieure  $\mathbb{R} \in \mathcal{M}_n(\mathbb{C})$  telles que

$$\mathbb{A} = \mathbb{QR}. \tag{18}$$

Si  $\mathbb A$  est réelle alors  $\mathbb Q$  et  $\mathbb R$  sont aussi réelles et l'on peut choisir  $\mathbb Q$  de telle sorte que les coefficients diagonaux de  $\mathbb R$  soient positifs. De plus, si  $\mathbb A$  est inversible alors la factorisation est unique.

Méthodes directes Factorisation QR 2025/10/24 63 / 68

**Objectif:** transformer  $\mathbb{A} \in \mathcal{M}_n(C)$  en une matrice triangulaire supérieure à l'aide de (n-1) matrices unitaires  $(\mathbb{H}^{[j]})_{i=1}^{n-1}$ .

$$\forall j \in [\![1,n-1]\!], \ \mathbb{A}^{[j]} \stackrel{\mathsf{def}}{=} \mathbb{H}^{[j]} \mathbb{A}^{[j-1]}, \ \ \mathsf{avec} \ \mathbb{A}^{[0]} = \mathbb{A}.$$

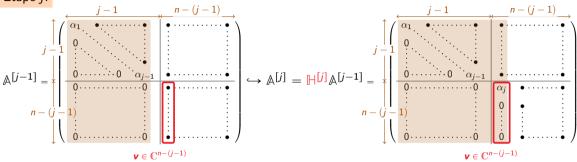
pour avoir

 $\mathbb{A}^{[n-1]} = \mathbb{H}^{[n-1]} \dots \mathbb{H}^{[1]} \mathbb{A}$  triangulaire supérieure.

### Etape 1:

$$\mathbb{A}^{[0]} = \begin{pmatrix} A_{1,1} & A_{1,2} & \cdots & A_{1,n} \\ A_{2,1} & A_{2,2} & \cdots & A_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n,1} & A_{n,2} & \cdots & A_{n,n} \end{pmatrix} \hookrightarrow \mathbb{A}^{[1]} \stackrel{\mathsf{def}}{=} \mathbb{H}_1 \mathbb{A}^{[0]} = \begin{pmatrix} \alpha_1 & \bullet & \cdots & \bullet \\ 0 & \bullet & \cdots & \bullet \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \bullet & \cdots & \bullet \end{pmatrix}.$$





Exercice 12, puis Exercice 13, pages suivantes!

65 / 68

Méthodes directes Factorisation QR 2025/10/24

Soit  $\mathbb{A} \in \mathcal{M}_{m+n}(\mathbb{C})$  la matrice bloc

$$\mathbb{A} = \frac{m}{n} \left( \begin{array}{c|c} m & n \\ \hline \mathbb{E} & \mathbb{V} \end{array} \right).$$

On note  $\mathbf{v} = \mathbb{V}_{:,1} \in \mathbb{C}^n$  le premier vecteur colonne de  $\mathbb{V}$  et on suppose que  $\mathbf{v}$  est non nul et non colinéaire à  $\mathbf{e}_1^n$  (premier vecteur de la base canonique de  $\mathbb{C}^n$ ).

Expliciter, en fonction de  $oldsymbol{v},$  le vecteur  $oldsymbol{u} \in \mathbb{C}^n, \|oldsymbol{u}\|_2 = 1,$  tel que

$$\mathbb{H}(\boldsymbol{u})\boldsymbol{v} = \alpha \boldsymbol{e}_{1}^{n}, \quad \text{avec } \mathbb{H}(\boldsymbol{u}) \stackrel{\text{def}}{=} \mathbb{I} - 2\boldsymbol{u}\boldsymbol{u}^{*}.$$

Q. 2

Soient 
$$\mathbf{x} \in \mathbb{C}^m$$
 et  $\mathbf{y} \in \mathbb{C}^n$ . On pose  $\mathbf{w} = \left(\frac{\mathbf{x}}{\mathbf{y}}\right) \in \mathbb{C}^{m+n}$ . Déterminer  $\mathbb{H}(\mathbf{w})$  en fonction de  $\mathbb{H}(\mathbf{x})$  et de  $\mathbb{H}(\mathbf{y})$ .

Q. 3

On pose 
$$\mathbf{w} = \left( \begin{array}{c} \mathbf{0}_m \\ \mathbf{u} \end{array} \right) \in \mathbb{C}^{m+n}.$$

- ① Déterminer  $\mathbb{H}(\mathbf{w})\mathbb{A}$  en fonction de  $\mathbb{H}(\mathbf{u})$ .
- Que peut-on dire de particulier sur le bloc (2.2) de  $\mathbb{H}(\mathbf{w})\mathbb{A}$ ?

#### Lemme

Soit  $\mathbb{A} \in \mathcal{M}_{m+n}(\mathbb{C})$  une matrice bloc

$$\mathbb{A} = {m \choose n} \left( \begin{array}{c|c} m & n \\ \hline \mathbb{E} & \mathbb{V} \end{array} \right).$$

On pose  $\mathbf{v} = \mathbb{V}_{:,1} \in \mathbb{C}^n$  le premier vecteur colonne de  $\mathbb{V}$ , et on suppose qu'il existe  $i \in [2, n]$  tel que  $v_i = \mathbb{V}_{i,1} \neq 0$ . Soit  $\boldsymbol{u} \in \mathbb{C}^n$  le vecteur défini par

$$\mathbf{\textit{u}} = \frac{\mathbf{\textit{v}} - \alpha \mathbf{\textit{e}}_1}{\|\mathbf{\textit{v}} - \alpha \mathbf{\textit{e}}_1^n\|} \text{ avec } \alpha = -\|\mathbf{\textit{v}}\|_2 \, \mathrm{e}^{\imath \arg(v_1)}$$

où  $e_1$  est le premier vecteur de la base canonique de  $\mathbb{C}^n$ .

En notant 
$$\mathbf{w} = \left(\begin{array}{c} \mathbf{0}_m \\ \mathbf{u} \end{array}\right) \in \mathbb{C}^{m+n}$$
, on a alors

$$\mathbb{H}(\boldsymbol{w})\mathbb{A} = \left(\begin{array}{c|c} \mathbb{U} & \mathbb{F} \\ \hline \mathbb{H}(\boldsymbol{u})\mathbb{E} & \mathbb{H}(\boldsymbol{u})\mathbb{V} \end{array}\right) \text{ avec } \mathbb{H}(\boldsymbol{u})\mathbb{V} = \begin{bmatrix} \alpha & \bullet \cdots \cdots \bullet \\ \hline 0 & \bullet \cdots \cdots \bullet \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \bullet \cdots \cdots \bullet \end{bmatrix}.$$

# Exercice 13

Soit  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$ .

- Expliquer comment construire une matrice  $\mathbb{H} \in \mathcal{M}_n(\mathbb{C})$  unitaire, produit d'au plus n-1 matrices élémentaires de Householder, et,  $\mathbb{R} \in \mathcal{M}_n(\mathbb{C})$ , triangulaire supérieure telles  $\mathbb{H} \mathbb{A} = \mathbb{R}$ .
- Ecrire une fonction FactQR permettant de calculer la factorisation  $\mathbb{QR}$  d'une matrice  $\mathbb{A} \in \mathcal{M}_n(\mathbb{C})$ .

  On pourra utiliser la fonction Householder (voir Exercice 3.1.9, page 92).
- Ecrire un programme permettant de tester cette fonction. On dispose des fonctions:
  - MatRand(m, n) retournant une matrice aléatoire de  $\mathcal{M}_{m,n}(\mathbb{C})$  chacune des parties imaginaires et réelles de ses éléments étant une variable aléatoire suivant la loi uniforme [0,1].
  - NormInf(A) retournant la norme infinie d'une matrice carrée A.

