Méthodes directes de la résolution des systèmes linéaires

Ali TAMOUSSIT

tamoussit2009@gmail.com

Table des matières

Introduction

La résolution des systèmes linéaires est considérée comme l'un des deux [1] problèmes fondamenteaux de *l'Analyse Numérique Matricielle*, et cette résolution intervient à divers domaines, en particulier lorsqu'il s'agit de modéliser puis résoudre numériquement des problèmes comme par exemple en :

- **Electrité :** recherche des intensités du courant dans un circuit électronique en utilisant les lois des noeuds et des mailles... ;
- **Chimie :** l'équilibre des équations des réactions chimiques ;
- **Algèbre linéaire :** recherche des vecteurs propres d'une matrice carrée associés à une valeur propre donnée, déterminer le noyau d'un endomorphisme en dimension finie (système homogène) ;
- **Géométrie analytique** : étudier la position relative des droites et des plans dans l'espace ;
- **Optimisation :** minimisation ou maximisation d'une fonction quadratique ;
- **Analyse numérique :** interpolation polynômiale (matrice de Vandermonde), résolution numérique des EDP (méthode des différences finies où les matrices rencontrées sont tridiagonales).

Position du problème :

Le problème consiste à trouver la solution du système d'équations

$$\begin{cases} a_{1,1}x_1 + a_{1,2}x_2 + \cdots + a_{1,n}x_n = b_1 \\ a_{2,1}x_1 + a_{2,2}x_2 + \cdots + a_{2,n}x_n = b_2 \\ \vdots \\ a_{n,1}x_1 + a_{n,2}x_2 + \cdots + a_{n,n}x_n = b_n \end{cases}$$

En notation matricielle le problème s'énonce comme suit :
On cherche à résoudre le système $Ax = b$ avec :

- A matrice à cœfficients réels de taille $n \times n$ et supposée inversible (i.e. $\det(A) \neq 0$) ;
- b vecteur de $\mathbb{R}^n$;
- x inconnue à chercher de $\mathbb{R}^n$.

Si $b = 0$ le système est dit ***homogène***, donc il admet au moins une solution $x = 0$ qui est la solution *nulle* ou *triviale* .
L'hypothèse d'inversibilité de A assure l'existence et l'unicité de la solution du système.

L'idée générale de résolution d'un système linéaire :

Pour résoudre un système d'équations linéaires on cherche à remplacer le système initial par un autre, plus simple, ayant la même solution, sachant que la résolution de système $Ax = b$ est très simple lorsque :

- A diagonale ;
- A orthogonale ;
- A tridiagonale ;
- A triangulaire supérieure ou inférieure ;
- A creuse (en général).

Classification des méthodes de résolution :

Il existe deux grandes familles de méthodes de résolution :

1. **Les méthodes directes** qui permettent d'obtenir la solution en un nombre fini d'opérations soit par triangularisation ou soit par décomposition de la matrice A.
   Les principales méthodes sont :
   - L'élimination de Gauss,
   - La décomposition LU,
   - La décomposition de Cholesky,

1. Les deux problèmes fondamentaux de *l'Analyse Numérique Matricielle* sont :
   a. Résolution d'un système linéaire,
   b. Calcul des valeurs propres et des vecteurs propres d'une matrice.

– La décomposition QR.
Ces méthodes sont utilisées pour les matrices pleines et les petits systèmes (n peu élevé).

2. **Les méthodes itératives** qui consistent à construire une suite $(x_n)_n$ qui converge vers la solution.
Les principales méthodes sont :
– Méthode de Jacobi,
– Méthode de Gauss-Seidel,
– Méthode du gradient conjugué.
Ces méthodes sont utilisées pour les matrices creuses et les grands systèmes.

Remarque : *Le choix entre les méthodes directes et les méthodes itératives dépend du type (ou forme) de la matrice du système.*

Les méthodes directes :

Les différentes méthodes directes ramènent la résolution de $Ax = b$ à la résolution d'un système dont la matrice est triangulaire.
Pour cela, on cherche une matrice inversible M telle que MA soit triangulaire ; alors :

$$Ax = b \Leftrightarrow MAx = Mb.$$

Les méthodes directes que nous allons étudier sont :
- **Elimination de Gauss :** avec pivot nul et non nul ;
- **Décomposition LU :** L triangulaire inférieure , U triangulaire supérieure ;
- **Décomposition de Cholesky :** décomposition LL^T ;
- **Décomposition QR :** par la méthode de Householder.

1 Rappels et compléments sur les matrices

On rappelle les résultats élémentaires en algèbre linéaire qui concerne les matrices.

Notation 1.1. *$\mathcal{M}_n(\mathbb{R})$ désigne l'ensemble des matrices carrées d'ordre n à cœfficients réels.*

1.1 L'inverse des matrices :

Définition 1.1. *Une matrice $A \in \mathcal{M}_n(\mathbb{R})$ est dite **inversible** s'il existe une matrice de $\mathcal{M}_n(\mathbb{R})$ notée A^{-1} telle que : $AA^{-1} = A^{-1}A = I_n$.*

Notation 1.2. *$GL_n(\mathbb{R})$ est l'ensemble des matrices carrées inversibles d'ordre n.*

Proposition 1.1. *Si $A \in GL_n(\mathbb{R})$, alors : $A^{-1} \in GL_n(\mathbb{R})$ et $(A^{-1})^{-1} = A$;*
$\forall A, B \in GL_n(\mathbb{R})$: $AB \in GL_n(\mathbb{R})$ et $(AB)^{-1} = B^{-1}A^{-1}$.

Proposition 1.2. ***(Inversibilité et valeurs propres)***
Soit $A \in \mathcal{M}_n(\mathbb{R})$.

$$A \in GL_n(\mathbb{R}) \Longleftrightarrow \textit{0 n'est pas une valeur propre de } A.$$

Théorème 1.1. *(**Critère de Hadamard**[2])*
Soit $A = (a_{i,j})_{1\leq i,j\leq n} \in \mathcal{M}_n(\mathbb{R})$ telle que : $\forall i \in \{1, 2, \cdots, n\}$: $|a_{i,i}| > \sum_{\substack{j=1\\j\neq i}}^{n} |a_{i,j}|, (\star)$

alors A est inversible.

Démonstration. *Effectuons une démonstration par l'absurde en supposant qu'il existe un vecteur colonne $x = (x_1, \cdots, x_n)^T$ non nul tel que $Ax = 0$.*
Soit k élément de $1, 2, \cdots, n$ tel que $|x_k| = \max_{1\leq i\leq n}(|x_i|)$.
Remarquons que : $|x_k| > 0$.
En effet, on a le vecteur x est non nul, alors $\exists i_0 \in \{1, 2, \cdots, n\}$ tel que $x_{i_0} \neq 0$.

2. Jacques Salomon Hadamard (1865-1963) est un mathématicien français.

Donc $|x_k| \geq |x_{i_0}| > 0$.

Etudions alors la $k^{\text{ème}}$ *équation tirée de* $Ax = 0$ *:* $\quad a_{k,k}x_k + \sum\limits_{\substack{i=1\\i\neq k}}^{n} a_{k,i}x_i = 0$

$$\Rightarrow \quad |a_{k,k}x_k| = \left|\sum_{\substack{i=1\\i\neq k}}^{n} a_{k,i}x_i\right| \quad \Rightarrow \quad |a_{k,k}|\,|x_k| \leq |x_k| \left|\sum_{\substack{i=1\\i\neq k}}^{n} a_{k,i}\right|$$

donc $|a_{k,k}| \leq \left|\sum\limits_{\substack{i=1\\i\neq k}}^{n} a_{k,i}\right|$ *ce qui contredit le fait que* A *vérifie* $(\star)$.
Ainsi A *est inversible.*

Définition 1.2. *Une telle matrice* A *qui vérifie* $(\star)$ *est dite à* ***diagonale strictement dominante****.*

Exemple 1.1. *La matrice* $A = \begin{pmatrix} -4 & 2 & 1 \\ 1 & 6 & 2 \\ 1 & -2 & 5 \end{pmatrix}$ *est une matrice à diagonale strictement dominante.*
Donc la matrice A *est inversible.*

Remarque 1.1. *Le critère de Hadamard donne une condition suffisante mais qui n'est pas nécessaire.*
Contre exemple : *La matrice* $B = \begin{pmatrix} -2 & 2 & 1 \\ 1 & 3 & 2 \\ 1 & -2 & 0 \end{pmatrix}$ *est une matrice inversible qui n'est pas à diagonale strictement dominante.*

Proposition 1.3. *Les cœfficients diagonaux d'une matrice à diagonale strictement dominante sont non nuls.*

1.2 Matrices triangulaires :

Définition 1.3. *Une matrice* $A = (a_{i,j})_{1\leqslant i,j\leqslant n} \in \mathcal{M}_n(\mathbb{R})$ *est dite* ***triangulaire inférieure***[3] *(respectivement* ***triangulaire supérieure***[4]*), si* $\forall i < j$ *(respectivement* $i > j$*),* $i, j \in \Big\{1, 2, \cdots n\Big\}$ *:* $a_{i,j} = 0$.

Remarque 1.2. *Une matrice qui est à la fois triangulaire inférieure et triangulaire supérieure est dite* ***diagonale****.*

Proposition 1.4. *Soit* $A = (a_{i,j})_{1\leqslant i,j\leqslant n} \in \mathcal{M}_n(\mathbb{R})$.

1. *Si* A *est triangulaire inférieure (respectivement supérieure) et inversible, alors* A^{-1} *est triangulaire inférieure (respectivement supérieure).*
2. *Si* A *est triangulaire inférieure (respectivement supérieure), alors :*
$$A \text{ est inversible} \Leftrightarrow a_{i,i} \neq 0 \ , \ \forall i \in \Big\{1, 2, \cdots n\Big\} \ .$$
3. *Soit* $B \in \mathcal{M}_n(\mathbb{R})$. *Si* A *et* B *sont triangulaires inférieures (respectivement supérieures), alors le produit* AB *est triangulaire inférieure (respectivement supérieure).*

1.3 La transposition :

Définition 1.4. *Soit* $A = (a_{i,j}) \in \mathcal{M}_{n,m}(\mathbb{R})$.
On définit la matrice ***transposée*** *de* A *et on note* A^T *la matrice de* $\mathcal{M}_{m,n}(\mathbb{R})$ *par :*
$A^T = (a_{j,i}) \in \mathcal{M}_{m,n}(\mathbb{R})$.

Proposition 1.5. *Soient* $A \in \mathcal{M}_{n,m}(\mathbb{R})$ *et* $B \in \mathcal{M}_{m,p}(\mathbb{R})$*, on a :* $(AB)^T = B^T A^T$.

Proposition 1.6. *Si* $A \in \mathcal{M}_n(\mathbb{R})$ *est inversible, alors* A^T *est inversible et on a :* $\Big(A^T\Big)^{-1} = (A^{-1})^T$ *qui sera notée* A^{-T}.

3. en anglais : Lower triangular matrix.
4. en anglais : Upper triangular matrix.

1.4 Matrices symétriques :

Définition 1.5. *Une matrice $A \in \mathcal{M}_n(\mathbb{R})$ est **symétrique** si $A^T = A$.*

Notation 1.3. *$\mathcal{S}_n(\mathbb{R})$ est l'ensemble des matrices carrées symétriques d'ordre n.*

Proposition 1.7. *Si $A \in \mathcal{M}_n(\mathbb{R})$ et B une matrice quelconque, alors on a : $A + A^T \in \mathcal{S}_n(\mathbb{R})$ et les matrices BB^T et B^TB sont symétriques.*

1.5 Le déterminant :

On note $\mathfrak{S}_n$ l'ensemble des permutations d'ordre n.

Définition 1.6. *Soit $A \in \mathcal{M}_n(\mathbb{R})$.*
*On définit le **déterminant** de A comme suit : $\det(A) = \sum_{\sigma \in \mathfrak{S}_n} \varepsilon(\sigma) \prod_{i=1}^{n} a_{i,\sigma(i)}$, où $\varepsilon(\sigma)$ est la signature de σ.*

Remarque 1.3. *On voit que pour calculer le $\det(A)$ par la formule précedente il faut effectuer card$(\mathfrak{S}_n) = n!$ sommes et $n \times n!$ produits.*

Proposition 1.8. *Si $A, B \in \mathcal{M}_n(\mathbb{R})$, on a :*

$$\det(AB) = \det(A)\det(B) = \det(BA) \ \text{ et } \ \det(A^T) = \det(A).$$

Le déterminant d'une matrice triangulaire ou diagonale est égal au produit des éléments diagonaux. En particulier : $\det(I_n) = 1$.

Proposition 1.9. *$GL_n(\mathbb{R}) = \{A \in \mathcal{M}_n(\mathbb{R}) : \ \det(A) \neq 0\}$.*
Si $A \in GL_n(\mathbb{R})$, alors : $\det(A^{-1}) = \dfrac{1}{\det(A)}$.

Définition 1.7. *Une matrice $A \in \mathcal{M}_n(\mathbb{R})$ est dite **orthogonale**, si elle vérifie : $AA^T = A^TA = I_n$.*

Proposition 1.10. *Si $A \in \mathcal{M}_n(\mathbb{R})$ est orthogonale, alors A est inversible, $A^{-1} = A^T$ et $\det(A) = \pm 1$.*

1.6 Matrices définies positives :

Définition 1.8. *Une matrice $A \in \mathcal{M}_n(\mathbb{R})$ symétrique est dite **définie positive** si :*

i. $\forall x \in \mathbb{R}^n$, $x^TAx \geq 0$;
ii. $\forall x \in \mathbb{R}^n$, on a : $x^TAx = 0 \Leftrightarrow x = 0$.

Proposition 1.11. *Si A une matrice carrée réelle inversible, alors A^TA est une matrice symétrique définie positive.*

Proposition 1.12. *Si $A \in \mathcal{M}_n(\mathbb{R})$ une matrice définie positive, alors : $\det(A) > 0$*
et $\quad \forall i \in \{1, 2, \cdots, n\} : a_{i,i} > 0$.

Remarque 1.4. *Une matrice définie positive est inversible, mais la réciproque n'est pas vraie.*

Définition 1.9. ***(Mineurs principaux)***
Soit $A = (a_{i,j})_{1 \leqslant i,j \leqslant n} \in \mathcal{M}_n(\mathbb{R})$ une matrice.

*On appelle **mineurs principaux** de A les déterminants des matrices $A_k := \begin{pmatrix} a_{1,1} & \cdots & a_{1,k} \\ \vdots & \ddots & \vdots \\ a_{k,1} & \cdots & a_{k,k} \end{pmatrix}$ pour k entier compris entre 1 et n.*

Théorème 1.2. ***(Critère de Sylvester[5])***
Soit $A \in \mathcal{M}_n(\mathbb{R})$, il y a équivalence entre

1. *A définie positive,*

5. James Joseph Sylvester (1814-1897) est un mathématicien et géomètre anglais.

2. A est symétrique et tous les mineurs principaux sont strictements positifs.

Exemple 1.2. *La matrice* $A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 5 & 5 & 5 \\ 1 & 5 & 14 & 14 \\ 1 & 5 & 14 & 15 \end{pmatrix}$ *est une matrice symétrique définie positive.*

Remarque 1.5. *On peut encore caractériser les matrices définies positives à l'aide des valeurs propres.*

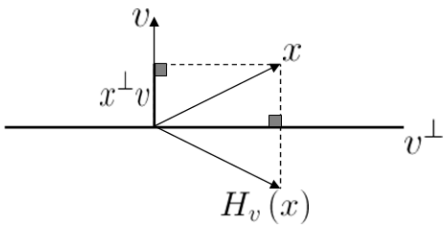
1.7 Matrices de Householder :

Définition 1.10. *On appelle **matrice de Householder** associée au vecteur $v \in \mathbb{R}^n$ non nul, la matrice*

$$H_v = I_n - 2\frac{vv^T}{||v||_2^2}.$$

Par convention $H_0 = I_n$.
Si $x \in \mathbb{R}^n$, on voit que $H_v(x) = x - 2\dfrac{x^T v}{||v||_2^2}v.$

Géométriquement, la matrice H_v est la matrice de la symétrie par rapport à l'orthogonal au sous-espace engendré par v.



Cela se voit facilement car $H_v(v) = -v$ et $H_v(x) = x$ si x appartient à l'orthogonal de v.

Proposition 1.13. *Soit $v \in \mathbb{R}^n$ un vecteur non nul. Alors H_v est une matrice symétrique et orthogonale.*

Proposition 1.14. *Si $x \in \mathbb{R}^n$ et si u est un vecteur unitaire[6], posons $v = x + ||x||u$; alors*

$$H_v(x) = -||x||u.$$

Remarque 1.6. *L'intérêt de ces matrices provient du fait suivant : étant donné un vecteur a, il est toujours possible de trouver une matrice H de Householder telle que : $(Ha)_i = 0, \ \ \forall\, i \neq 1$.*

1.8 Décompositions matricielles :

Théorème 1.3. ***(Théorème d'élimination de Gauss)***
Soit $A \in \mathcal{M}_n(\mathbb{R})$ inversible ou non. Il existe au moins une matrice inversible M telle que MA soit triangulaire supérieure.

Démonstration. *On va en fait construire MA d'une manière itérative sur les lignes en construisant une suite de matrices : $A^{(1)}, A^{(2)}, \cdots$, jusqu'à $A^{(n)} = MA$.*
On donne par la même occasion l'algorithme de construction de MA sans avoir en fait à connaître la matrice M à aucun moment.

- ***Etape 1 :*** *$A^{(1)} = A$, on construit $\widetilde{A}^{(1)} = P_1 A^{(1)}$ où P_1 est une matrice de permutation telle que $\widetilde{a}_{1,1}^{(1)} \neq 0$: Si $a_{1,1}^{(1)} \neq 0$, on prend $P_1 = I_n$. Sinon, si il existe i tel que $a_{i,1}^{(1)} \neq 0$, on permute les lignes i et 1. $a_{i,1}^{(1)}$ devient $\widetilde{a}_{1,1}^{(1)}$ que l'on appelle **pivot** de $\widetilde{A}^{(1)}$.*
  Ensuite, on multiplie $\widetilde{A}^{(1)}$ par une matrice E_1 telle que $A^{(2)} = E_1\widetilde{A}^{(1)}$ ne contienne que des zéros sur la première colonne à partir de la deuxième ligne. La $i^{\text{ème}}$ ligne de $A^{(2)}$ est en fait obtenue par la combinaison des $i^{\text{ème}}$ et $1^{\text{ère}}$ lignes de $\widetilde{A}^{(1)}$ qui fait apparaître 0 dans la première colonne

$$a_{i,j}^{(2)} = \widetilde{a}_{i,j}^{(1)} - \frac{\widetilde{a}_{i,1}^{(1)}}{\widetilde{a}_{1,1}^{(1)}}\widetilde{a}_{1,j}^{(1)}.$$

  La matrice E_1 est définie par :

6. vecteur de norme 1.

$$\left|\begin{array}{l} e^1_{i,i} = 1, \ \ \forall i \in 1, 2, \cdots, n \\ \textit{La première colonne est } \left(1, \ -\dfrac{\widetilde{a}^{(1)}_{2,1}}{\widetilde{a}^{(1)}_{1,1}}, \ \cdots, \ -\dfrac{\widetilde{a}^{(1)}_{n,1}}{\widetilde{a}^{(1)}_{1,1}}\right)^T \\ \textit{Tous les autres éléments de la matrice sont nuls.} \end{array}\right.$$

- ***Etape*** k ***:*** $A^{(k)}$ *connue avec que des zéros en dessous de la diagonale sur les* $(k-1)$ *premières colonnes. On construit* $\widetilde{A}^{(k)} = P_k A^{(k)}$ *où* P_k *est une matrice de permutation telle que* $\widetilde{a}^{(k)}_{k,k} \neq 0$ *: Si* $a^{(k)}_{k,k} \neq 0$*, on prend* $P_k = I_n$*. Sinon, si il existe* $i > k$ *tel que* $a^{(k)}_{i,k} \neq 0$*, on permute les lignes* i *et* k*.* $a^{(k)}_{i,k}$ *devient* $\widetilde{a}^{(k)}_{k,k}$ *que l'on appelle* ***pivot*** *de* $\widetilde{A}^{(k)}$*.*
  La matrice P_k *est définie par :*
  $$\left|\begin{array}{l} p^k_{j,j} = 1, \ \ si \ \ j \neq i \ \ et \ \ j \neq k \\ p^k_{k,i} = p^k_{i,k} = 1 \\ \textit{Tous les autres éléments de la matrice sont nuls.} \end{array}\right.$$
  Ensuite, on multiplie $\widetilde{A}^{(k)}$ *par une matrice* E_k *telle que* $A^{(k+1)} = E_k \widetilde{A}^{(k)}$ *ne contienne que des zéros sur la* $k^{\text{ème}}$ *colonne à partir de la* $(k+1)^{\text{ème}}$ *ligne.Pour* $i > k$*, la* $i^{\text{ème}}$ *ligne de* $A^{(k+1)}$ *est en fait obtenue par la combinaison des* $i^{\text{ème}}$ *et* $k^{\text{ème}}$ *lignes de* $\widetilde{A}^{(k)}$ *qui fait apparaître 0 dans la* $k^{\text{ème}}$ *colonne*
  $$a^{(k)}_{i,j} = \widetilde{a}^{(k)}_{i,j} - \frac{\widetilde{a}^{(k)}_{i,k}}{\widetilde{a}^{(k)}_{k,k}} \widetilde{a}^{(k)}_{k,j}.$$
  La matrice E_k *est définie par :*
  $$\left|\begin{array}{l} e^k_{i,i} = 1, \ \ \forall i \in 1, 2, \cdots, n \\ \textit{La } k^{\text{ième}} \textit{ colonne est } \left(0\,, \cdots, \ 0, \ -\dfrac{\widetilde{a}^{(k)}_{k+1,k}}{\widetilde{a}^{(k)}_{k,k}}, \ \cdots, \ -\dfrac{\widetilde{a}^{(k)}_{n,k}}{\widetilde{a}^{(k)}_{k,k}}\right)^T \\ \textit{Tous les autres éléments de la matrice sont nuls.} \end{array}\right.$$
- ***Après l'étape*** $n-1$ ***:*** *On obtient* $A^{(n)}$ *telle que* $A^{(n)}$ *triangulaire surpérieure.*
  $A^{(n)} = MA$ *où* $M = E_{n-1}P_{n-1} \cdots E_1 P_1$*.*
  $det(M) = \pm 1$ *par définition des* E_i *et* P_i*.*

Remarque 1.7. *En pratique, on ne calcule pas explicitement la matrice* M*, mais seulement la matrice* MA*. On fait subir la matrice* A *à des permutation et combinaisons de lignes.*

Exemple 1.3.

$$A = A^{(1)} = \begin{pmatrix} 5 & 2 & 1 \\ 5 & -6 & 2 \\ -4 & 2 & 1 \end{pmatrix}$$

$$P_1 = I_3, \ \ E_1 = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ \dfrac{4}{5} & 0 & 1 \end{pmatrix}, \ \ A^{(2)} = \begin{pmatrix} 5 & 2 & 1 \\ 0 & -8 & 1 \\ 0 & \dfrac{18}{5} & \dfrac{9}{5} \end{pmatrix}$$

$$P_2 = I_3, \ \ E_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \dfrac{9}{20} & 1 \end{pmatrix}, \ \ A^{(3)} = \begin{pmatrix} 5 & 2 & 1 \\ 0 & -8 & 1 \\ 0 & 0 & \dfrac{9}{4} \end{pmatrix}$$

On peut évidemment calculer la matrice $M = E_2 E_1 = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ \dfrac{7}{20} & \dfrac{9}{20} & 1 \end{pmatrix}$*, ce qui permet d'écrire* $MA = A^{(3)}$*, soit* $\begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ \dfrac{7}{20} & \dfrac{9}{20} & 1 \end{pmatrix} \begin{pmatrix} 5 & 2 & 1 \\ 5 & -6 & 2 \\ -4 & 2 & 1 \end{pmatrix} = \begin{pmatrix} 5 & 2 & 1 \\ 0 & -8 & 1 \\ 0 & 0 & \dfrac{9}{4} \end{pmatrix}$ *qui est bien une matrice triangulaire supérieure.*

Théorème 1.4. ***(Factorisation ou décomposition** LU [7]**)***
Soit $A \in \mathcal{M}_n(\mathbb{R})$ une matrice dont tous les mineurs principaux sont non nuls. Il existe un unique couple de matrices (L, U), avec U triangulaire supérieure, et L triangulaire inférieure à diagonale unité (i.e. $l_{i,i} = 1$), tel que $A = LU$.

Démonstration. *On doit vérifier l'unicité et l'existence :*

- ***L'unicité :*** *Supposons qu'il existe deux décompositions LU de A : $A = L_1U_1 = L_2U_2$ avec L_i $(i = 1, 2)$ triangulaire inférieure, et U_i $(i = 1, 2)$ triangulaire supérieure.*
  Alors, $L_1U_1 = L_2U_2 \Leftrightarrow U_1U_2^{-1} = L_1^{-1}L_2$ qui est à la fois triangulaire supérieure (à gauche) et triangulaire inférieure (à droite).
  Ce qui prouve que $U_1U_2^{-1}$ et $L_1^{-1}L_2$ sont des matrices diagonales, et puisque $l_{i,i} = 1$, on a : $L_1 = L_2$ et $U_1 = U_2$.
  D'où l'unicité.
- ***L'existence :*** *Par récurrence sur n.*
  Pour $n = 1$ le résultat est évidement vrai.
  On suppose que le résultat vrai pour $n - 1$.
  On décompose la matrice A par blocs sous la forme : $A = \begin{pmatrix} A_{n-1} & c \\ b^T & a_{n,n} \end{pmatrix}$ *où A_{n-1} est la matrice carrée d'ordre $n-1$ formée des $n-1$ premières lignes et colonnes de A, b et c sont des vecteurs colonnes données par :* $b = \begin{pmatrix} a_{n,1} & a_{n,2} & \cdots & a_{n,n-1} \end{pmatrix}^T$, $c = \begin{pmatrix} a_{1,n} & a_{2,n} & \cdots & a_{n-1,n} \end{pmatrix}^T$.
  On a A_{n-1} vérifie l'hypothèse de récurrence, donc : $A_{n-1} = L_{n-1}U_{n-1}$.
  Cherchons alors L et U décomposées par blocs sous la forme :
  $$L = \begin{pmatrix} L_{n-1} & 0 \\ l^T & 1 \end{pmatrix}, U = \begin{pmatrix} U_{n-1} & u \\ 0 & u_{n,n} \end{pmatrix}.$$
  En effectuant le produit par blocs et en identifiant à la décomposition de A, on obtient le système d'équations : $\begin{cases} A_{n-1} &= L_{n-1}U_{n-1} \\ b^T &= l^TU_{n-1} \\ c &= L_{n-1}u \\ a_{n,n} &= l^Tu + u_{n,n} \end{cases}$ *ce qui donne :* $\begin{cases} l^T &= b^T(U_{n-1})^{-1} \\ u &= (L_{n-1})^{-1}c \\ u_{n,n} &= a_{n,n} - b^T(A_{n-1})^{-1}c \end{cases}$
  Ceci termine la démonstration.

Exemple 1.4. *Soit la matrice suivante :* $A = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix}$.
La matrice A se décompose en un produit d'une matrice triangulaire inférieure par une matrice triangulaire supérieure de la façon suivante : $A = LU$, où $L = \begin{pmatrix} 1 & 0 & 0 \\ \frac{-1}{2} & 1 & 0 \\ 0 & \frac{-2}{3} & 1 \end{pmatrix}$ *et* $U = \begin{pmatrix} 1 & -1 & 0 \\ 0 & \frac{3}{2} & -1 \\ 0 & 0 & \frac{-2}{3} \end{pmatrix}$.
Par contre la matrice $B = \begin{pmatrix} 0 & 1 \\ 2 & 3 \end{pmatrix}$ *n'admet pas de décomposition LU.*

Remarque 1.8. *Si A est simplement supposée inversible, alors A peut s'écrire $A = PLU$, où P est une matrice de permutation.*

Proposition 1.15. *Si A une matrice à diagonale strictement dominante alors elle admet une factorisation LU.*

Démonstration. *En effet les sous-matrices principales A_k sont aussi clairement à diagonale strictement dominante donc inversibles et le théorème précédent s'applique.*

Remarque 1.9. ***(Application de la décomposition** LU **au calcul de déterminant)***
La décomposition LU permet de calculer facilement le déterminant de A , qui est égal au produit des éléments diagonaux de la matrice U.
En effet : $\det(A) = \det(LU) = \det(L)\det(U) = 1 \times \det(U) = \prod_{i=1}^{n} u_{i,i}$.

7. la terminologie *LU* pour *Lower-Upper*.

Théorème 1.5. ***(Factorisation ou décomposition de Cholesky)***
Soit $A \in \mathcal{M}_n(\mathbb{R})$ une matrice symétrique définie positive. Alors il existe (au moins) une matrice réelle triangulaire inférieure L telle que $A = LL^T$.
De plus, on peut imposer que les éléments diagonaux de la matrice L soient strictement positifs, et la décomposition $A = LL^T$ correspondante est alors unique.

Démonstration. *On doit vérifier l'unicité et l'existence :*

- ***L'unicité*** *: Supposons que $A = LL^T = MM^T$ tels que $l_{i,i} > 0$ et $m_{i,i} > 0$ $\forall i \in \{1, ..., n\}$.*
  La matrice $M^{-1}L = M^T L^{-T}$ est à la fois triangulaire inférieure (à gauche) et triangulaire supérieure (à droite), c'est donc une matrice diagonale .
  Donc : $m_{i,i}^{-1} l_{i,i} = m_{i,i} l_{i,i}^{-1}$.
  La condition de la positivité $\Rightarrow m_{i,i}^{-1} \times l_{i,i} = m_{i,i} \times l_{i,i}^{-1} = 1$.
  Ainsi : $M^{-1}L = M^T L^{-T} = I_n$ i.e. $L = M$
  D'où l'unicité.
- ***L'existence*** *: Par récurrence sur n.*
  Pour $n = 1$, on prend : $L = \sqrt{a_{1,1}}$
  Supposons que le décomposition de Cholesky existe pour toute matrice définie positive d'ordre $n-1$:
  $A = \begin{pmatrix} A_{n-1} & a \\ a^T & a_{n,n} \end{pmatrix}$ *avec* $a = \begin{pmatrix} a_{1,n} & a_{2,n} & \cdots & a_{n-1,n} \end{pmatrix}^T$.
  Notons : $A_{n-1} = L_{n-1} L_{n-1}^T$ la décomposition de Cholesky de A_{n-1}, on a :

$$A = \begin{pmatrix} L_{n-1} & 0 \\ u^T & \alpha \end{pmatrix} \times \begin{pmatrix} L_{n-1}^T & u \\ 0 & \beta \end{pmatrix}.$$

  En prenant $L_{n-1}u = a$ et $||u||^2 + \alpha\beta = a_{n,n}$ on obtiendra la décomposition de A si l'on prend $\alpha = \beta$ ce qui sera impossible si $\alpha\beta < 0$
  L'égalité ci-dessus prouve que : $\det(A) = \det(L_{n-1})\alpha \det(L_{n-1}^T)\beta$.
  Comme, par hypothèse $\det(A) > 0$ et $(\det(L_{n-1}))^2 > 0$, donc on a bien $\alpha\beta > 0$.

Exemple 1.5. *Soit la matrice suivante :* $A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 5 & 5 & 5 \\ 1 & 5 & 14 & 14 \\ 1 & 5 & 14 & 15 \end{pmatrix}$.

La matrice A se décompose de la façon suivante : $A = LL^T$, où $L = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ 1 & 2 & 3 & 0 \\ 1 & 2 & 3 & 1 \end{pmatrix}$.

Remarque 1.10. ***(Application de la décomposition de Cholesky au calcul de déterminant)***
La décomposition de Cholesky permet de calculer facilement le déterminant de A, qui est égal au carré du produit des éléments diagonaux de la matrice L, puisque :

$$\det(A) = \det\left(LL^T\right) = (\det(L))^2 = \left(\prod_{i=1}^{n} l_{i,i}\right)^2.$$

Théorème 1.6. ***(Factorisation ou décomposition** QR)*
Soit $A \in \mathcal{M}_n(\mathbb{R})$ une matrice inversible.
Alors, il existe une matrice orthogonale Q et une matrice triangulaire supérieure R telles que $A = QR$.
De plus, on peut trouver R à éléments diagonaux strictement positifs. La factorisation est alors unique.

Démonstration. *On doit vérifier l'unicité et l'existence.*

- ***L'unicité*** *:*
  Supposons que la matrice A possède deux décompositions $A = Q_1R_1$ et $A = Q_2R_2$, les cœfficients diagonaux respectifs de R_1 et de R_2 ayant le même signe.
  Alors $Q_1R_1 = Q_2R_2 \Leftrightarrow Q_2^{-1}Q_1 = R_2R_1^{-1}$.
  La matrice $M = R_2R_1^{-1}$ est triangulaire supérieure, ses cœfficients diagonaux sont les quotients deux à deux de ceux de R_2 et R_1 et sont donc strictement positifs.

La matrice $M = Q_2^{-1}Q_1$ est orthogonale.
La matrice M^{-1} est donc à la fois triangulaire supérieure et triangulaire inférieure.
Les matrices M et M^{-1} sont donc diagonales .
Du fait de l'égalité $M^{-1} = M^T$, chaque cœfficient diagonal de M est égal à son propre inverse et vaut ± 1, mais on sait que ces cœfficients sont positifs : ils sont donc égaux à 1.
On en déduit que $M = I_n$ (c-à-d $Q_1 = Q_2$ et $R_1 = R_2$).
D'où l'unicité.

- ***L'existence** :*
Soit $A = Q^+R^+$ la décomposition de A (les cœfficients diagonaux de R étant strictement positifs).
Soit X un sous-ensemble quelconque de $\{1, ..., n\}$.
Soit J la matrice diagonale dont les cœfficients diagonaux successifs λ_i valent 1 si $i \in X$ et -1 sinon.
J est donc une matrice orthogonale qui vérifie $J^2 = I_n$.
L'égalité $A = Q^+R^+$ s'écrit donc aussi $A = QR$ avec $Q = Q^+J$ et $R = JR^+$.
La matrice Q est orthogonale (elle se déduit de Q^+ en inversant les signes dans les colonnes d'indice j n'appartient pas à X) et R est triangulaire supérieure (elle se déduit de R^+ en inversant les signes dans les lignes d'indice i n'appartient pas à X).
Alors on a formé une décomposition de $A = QR$ pour laquelle le signe des cœfficients diagonaux de R est fixé à l'avance.

Remarque 1.11. *La factorisation QR d'une matrice A consiste à trouver une matrice Q orthogonale et une matrice R triangulaire supérieure telles que : $A = QR$. Il existe une autre démonstration (basée sur la **remarque 1.6**) qui utilise les propriétés des matrices de Householder "voir [**1**], pages 92 et 93", qui permet de construire une suite de $n-1$ matrices de Householder telles que : $H_{n-1}H_{n-2}\cdots H_2H_1A = R$ et $Q = H_1H_2\cdots H_{n-1}$. Et comme application de cette méthode, on voit qu'il est facile de calculer le déterminant de la matrice A :* $\det(A) = \pm\det(R) = \pm\prod_{i=1}^{n} r_{i,i}$.

2 La méthode d'élimination de Gauss

La biographie de Gauss :

Johann Carl Friedrich Gauss (30 avril 1777 - 23 février 1855) est un mathématicien, astronome et physicien allemand. Doté d'un grand génie, il a apporté de très importantes contributions à ces trois sciences. Surnommé « le prince des mathématiciens », il est considéré comme l'un des plus grands mathématiciens de tous les temps.

Johann Carl Friedrich Gauss (1777-1855)

2.1 Résolution des systèmes linéaires triangulaires :

On considère le système suivant : $Ax = b$, où A est triangulaire inversible (donc les $a_{i,i} \neq 0$ pour tout $i \in \{1, 2, \cdots, n\}$).

1. Si A triangulaire inférieure, alors on a n équations sous la forme :

$$\begin{array}{lllll} a_{1,1}x_1 & & & & = b_1 \\ a_{2,1}x_1 & + a_{2,2}x_2 & & & = b_2 \\ & & \ddots & & \vdots \\ a_{n,1}x_1 & + a_{n,2}x_2 & + \cdots + & a_{n,n}x_n & = b_n \end{array}$$

On aura donc : $x_1 = \dfrac{b_1}{a_{1,1}}$ et pour $i = 2, 3, \cdots, n$

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

Cet algorithme est appelé *méthode de descente*[8].

2. Si A triangulaire supérieure, alors on a n équations sous la forme :

$$\begin{array}{cccccc} a_{1,1}x_1 + & a_{1,2}x_2 + & \cdots & + a_{1,n}x_n & = & b_1 \\ & a_{2,2}x_2 + & \cdots & + a_{2,n}x_n & = & b_2 \\ & & \ddots & & & \vdots \\ & & & a_{n,n}x_n & = & b_n \end{array}$$

On aura donc : $x_n = \dfrac{b_n}{a_{n,n}}$ et pour $i = n-1, \cdots, 2, 1$

$$x_i = \frac{1}{a_{i,i}} \left(b_i - \sum_{j=i+1}^{n} a_{i,j} x_j \right)$$

Cet algorithme est appelé *méthode de remontée*[9].

Ces deux algorithmes nécessitent le même nombre d'opérations, tel que :
- $\dfrac{n(n-1)}{2}$ additions,
- $\dfrac{n(n-1)}{2}$ multiplications,
- n divisions.

2.2 Le principe de la méthode d'élimination de Gauss :

Soit le système linéaire $Ax = b$ avec A une matrice carrée d'ordre n qui est inversible.
L'idée de la méthode de Gauss :
Si A n'est pas triangulaire, on est amené à trouver une matrice inversible M telle que MA soit triangulaire supérieure.

Remarque 2.1. *L'existence d'une telle matrice M est garantie par le **théorème 1.3**.*

La méthode d'élimination de Gauss se décompose en trois étapes :
- **Etape 1 :** trouver une matrice inversible M telle que la matrice MA soit triangulaire supérieure (élimination successive des inconnues) ;
- **Etape 2 :** calcul simultané du vecteur Mb ;
- **Etape 3 :** résolution du système triangulaire supérieure $MAx = Mb$ (remontée).

Pour la méthode d'élimination de Gauss on distingue deux cas :
- **Sans pivotement :** on ne permet pas la permutation de deux lignes ; dans certain cas on ne pourra pas obtenir de matrice triangulaire supérieure.
- **Avec pivotement :** on permet la permutation des lignes ; dans ce cas on obtiendra toujours une matrice triangulaire supérieure si la matrice est inversible.

2.3 Les opérations élémentaires pour la triangularisation :

Décrivons tout d'abord les trois opérations élémentaires sur les lignes pour triangulariser un système linéaire :
- Multiplication d'une ligne L_i par un scalaire λ non nul : $L_i \leftarrow \lambda L_i$.
- Addition de deux lignes : $L_i \leftarrow L_i + L_j$.
- Permutation de deux lignes : $L_i \leftrightarrow L_j$.

8. Méthode de substitution avant, en anglais : Forward substitution method.
9. Méthode de substitution arrière, en anglais : Backward substitution method.

On peut interpréter ces trois opérations de base de manière matricielle.
L'opération $L_i \leftarrow \lambda L_i$ correspond à la matrice élémentaire

$$\begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & & & & \vdots \\ \vdots & \ddots & 1 & \ddots & & & \vdots \\ \vdots & & \ddots & \lambda & \ddots & & \vdots \\ \vdots & & & \ddots & 1 & \ddots & \vdots \\ \vdots & & & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & 1 \end{pmatrix} \begin{array}{l} \\ \\ \\ \leftarrow i^{\text{ème}} \, ligne \\ \\ \\ \\ \end{array}$$

L'opération $L_i \leftarrow L_i + L_j$ correspond à la matrice élémentaire

$$\begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & & & \vdots \\ \vdots & 0 & 1 & \ddots & & \vdots \\ \vdots & \vdots & 1 & 1 & \ddots & \vdots \\ \vdots & 0 & \cdots & 0 & \ddots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & 1 \end{pmatrix} \begin{array}{l} \\ \\ \leftarrow i^{\text{ème}} \, ligne \\ \leftarrow j^{\text{ème}} \, ligne \\ \\ \\ \end{array}$$

L'opération $L_i \leftrightarrow L_j$ correspond à la matrice élémentaire

$$\begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & 1 & \ddots & & & \vdots \\ \vdots & 0 & 0 & 1 & & \vdots \\ \vdots & \vdots & 1 & 0 & \ddots & \vdots \\ \vdots & 0 & \cdots & 0 & 1 & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & 1 \end{pmatrix} \begin{array}{l} \\ \\ \leftarrow j^{\text{ème}} \, ligne \\ \leftarrow i^{\text{ème}} \, ligne \\ \\ \\ \end{array}$$

Toutes ces matrices sont construites en appliquant l'opération élémentaire correspondante sur la matrice identité.

2.4 Méthode d'élimination de Gauss avec pivot non nul :

2.4.1 Description de l'algorithme :

Algorithme d'élimination de Gauss dans le cas d'un pivot non nul :

Entrées : une matrice A d'ordre n et un vecteur b

Initialisation : $U = A$ et $v = b$
pour $k = 1$ à $n - 1$ faire
  $p = U[k, k]$
  pour $i = k + 1$ à n faire
    $q = U[i, k]$
    $U[i, k] = 0$
    pour $j = k + 1$ à n faire
      $U[i, j] = U[i, j] - \dfrac{q}{p} U[k, j]$
    fin pour
    $v[i] = v[i] - \dfrac{q}{p} v[k]$
  fin pour
fin pour

Sorties : une matrice triangulaire supérieure U et un vecteur v

Algorithme de résolution d'un système triangulaire supérieur :
Entrées : une matrice triangulaire supérieure U d'ordre n et un vecteur v
Initialisation : $x[1:n]=0$ $x[n]=\dfrac{v[n]}{U[n,n]}$ pour $i=n-1$ à 1 faire $\quad x[i]=\dfrac{1}{U[i,i]}\left(v[i]-\displaystyle\sum_{j=i+1}^{n}U[i,j]x[j]\right)$ fin pour
Sortie : le vecteur x

2.4.2 Etude d'un exemple :

Pour illustrer cette méthode on va traiter un exemple à la main, puis en utilisant Maple dont le programme sera joint en annexe.
Examinons le système linéaire ci-dessous :

$$(\mathcal{S}):\begin{cases} 3x+5y+7z=101 & L_1\\ 2x+10y+6z=134 & L_2\\ x+2y+3z=40 & L_3\end{cases}$$

On le transforme le système initial en un système triangulaire supérieur équivalent et pour çela, l'élimination de Gauss nous facilite la tâche en procèdant par trois étapes :

1. **Elimination :**
Pour former un système triangulaire supérieur équivalent, on dispose des opérations élémentaires cités précédement en 2.3 appliquées au système homogène associé à $(\mathcal{S})$.
Itération 1 :
$$(\mathcal{S}_h):\begin{cases} 3x+5y+7z=0 & L_1\leftarrow L_1\\ \quad 20y+4z=0 & L_2\leftarrow 3L_2-2L_1\\ \quad y+2z=0 & L_3\leftarrow 3L_3-L_1\end{cases}$$
Itération 2 :
$$\begin{cases} 3x+5y+7z=0 & L_1\leftarrow L_1\\ \quad 20y+4z=0 & L_2\leftarrow L_2\\ \quad\quad 36z=0 & L_3\leftarrow 20L_3-L_2\end{cases}$$
En termes matriciels, la première itération consiste à établir $A^{(2)}=E_1A$ avec :
$$A^{(2)}=\begin{pmatrix}3&5&7\\0&20&4\\0&1&2\end{pmatrix}\quad et\quad E_1=\begin{pmatrix}1&0&0\\-2&3&0\\-1&0&3\end{pmatrix},$$
puis $A^{(3)}=E_2P_2A^{(2)}$ avec : $A^{(3)}=\begin{pmatrix}3&5&7\\0&20&4\\0&0&36\end{pmatrix}$ qui est une matrice triangulaire supérieure,
$E_2=\begin{pmatrix}1&0&0\\0&1&0\\0&-1&20\end{pmatrix}$ et $P_2=I_3$.
2. **Simultanément on calcule** $E_2P_2E_1b$:
D'après le système, on a : $b=\begin{pmatrix}101&134&40\end{pmatrix}^T$, donc : $E_2P_2E_1b=\begin{pmatrix}101\\200\\180\end{pmatrix}$.
3. **Résolution de** $A^{(3)}x=E_2P_2E_1b$ **:**
Le nouveau système triangulaire supérieur est très simple à résoudre.
$L_3: z=\dfrac{180}{36}=5.$
$L_2: 20y+20=200$ donc $y=\dfrac{200-20}{20}=9$

Enfin dans $L_1 : 3x + 45 + 35 = 101$ donc $x = \dfrac{101 - 45 - 35}{3} = 7$.

Finalement : $X = \begin{pmatrix} 7 & 9 & 5 \end{pmatrix}^T$, où $X = \begin{pmatrix} x & y & z \end{pmatrix}^T$.

2.4.3 Complexité de l'algorithme :

Comptons enfin le nombre d'opérations élémentaires requises dans la méthode de Gauss.

(i) L'élimination successives des inconnus :

- $\dfrac{n^3 - n}{3}$ additions,
- $\dfrac{n^3 - n}{3}$ multiplications,
- $\dfrac{n(n-1)}{2}$ divisions.

(ii) les opérations effectuées sur le second membre :

- $\dfrac{n(n-1)}{2}$ additions,
- $\dfrac{n(n-1)}{2}$ multiplications.

(iii) La remontée nécessite :

- $\dfrac{n(n-1)}{2}$ additions,
- $\dfrac{n(n-1)}{2}$ multiplications,
- n divisions.

En fin de compte, la méthode de Gauss nécessite donc de l'ordre de :

- $\dfrac{n^3}{3}$ additions,
- $\dfrac{n^3}{3}$ multiplications,
- $\dfrac{n^2}{2}$ divisions.

Donc la complexité de la méthode de Gauss est en $\mathcal{O}\left(\dfrac{2n^3}{3}\right)$.

2.5 Méthode d'élimination de Gauss avec pivot nul :

2.5.1 Influence du pivotage sur la précision

Il peut se passer dans la pratique que l'un des pivots $\widetilde{a}_{k,k}^{(k)}$ soit nul.

Examinons le système suivant (tiré de [1], pages 78 et 79) : $\begin{cases} 10^{-4}x & + & y & = & 1 \\ x & + & y & = & 2 \end{cases}$, dont la solution exacte est $x = 1,00010\ldots \simeq 1$ et $y = 0,99990\ldots \simeq 1$.
Appliquons la méthode de Gauss avec comme pivot 10^{-4}.
On obtient formellement $(1 - 10^4)y = 2 - 10^4$.
Ceci, calculé en virgule flottante avec trois chiffres significatifs, donne $y = 1$, puis $x = 0$, ce qui est totalement faux.
Par contre, si on commence par permuter les deux équations

$$\begin{cases} x & + & y & = & 2 \\ 10^{-4}x & + & y & = & 1 \end{cases}$$

et prenons comme pivot 1 . On obtient maintenant $(1 - 10^{-4})y = 1 - 2.10^{-4}$. Ceci, calculé en virgule flottante avec quatres chiffres significatifs, donne $y = 0,9999$, puis $x = 1$.
En fait la raison du problème est que le pivot 10^{-4} est trop petit .
Pour effectuer les calculs approchés dans le cas d'une virgule flottante avec trois et quatres chiffres on

utilise Maple avec les codes suivants :

```
>restart;
Digits:=3;
a1:=1-10^4;
a2:=2-10^4;
y:=evalf(a2/a1);
x:=(1-y)/(10^(-4));
```

$$Digits := 3$$
$$a1 := -9999$$
$$a2 := -9998$$
$$y := 1.00$$
$$x := 0.$$

```
>restart;
Digits:=4;
a1:=evalf(1-10^(-4));
a2:=evalf(1-2*10^(-4));
y:=evalf(a2/a1);
x:=(1-y)/(10^(-4));
```

$$Digits := 4$$
$$a1 := 0.9999$$
$$a2 := 0.9998$$
$$y := 0.9999$$
$$x := 1.$$

Remarque 2.2. ***(Le choix du pivot)***
Pour éviter les erreurs d'arrondi, on peut utiliser, pour le choix du pivot :
- ***la stratégie du pivot partiel :*** *à chaque étape k, on choisit* $\widetilde{a}_{i,k}^{(k)} = \underset{k \le j \le n}{max} \left|\widetilde{a}_{j,k}^{(k)}\right|$,
- ***la stratégie du pivot total :*** *à chaque étape k, on choisit pour pivot* $\widetilde{a}_{i,j}^{(k)} = \underset{k \le l,l' \le n}{max} \left|\widetilde{a}_{l,l'}^{(k)}\right|$.

Si le pivot choisi n'est pas sur la $k^{\text{ème}}$ ligne, il faut aussi effectuer une permutation des colonnes.

2.5.2 Description de l'algorithme :

Algorithme d'élimination de Gauss dans le cas d'un pivot nul :

Entrées : une matrice A d'ordre n et un vecteur b

```
Initialisation : AA = [A, b]
pour k = 1 à n faire
    p ← AA[k, k]
    l ← k
    pour i = k à n faire
        si |AA[i, k]| > 0 alors
            p ← AA[i, k]
            l ← i
        fin si
    fin pour
    si l ≠ k alors
        pour j = k à n + 1 faire
            temp ← AA[k, j]
            AA[l, j] ← AA[l, j]
            AA[l, j] ← temp
        fin pour
    fin si
    pour i = k + 1 à n faire
        q ← AA[i, k]
        AA[i, k] ← 0
        pour j = k + 1 à n + 1 faire
            AA[i, j] = AA[i, j] − (q/p) AA[k, j]
        fin pour
    fin pour
fin pour
```

Sortie : AA la matrice triangularisée

2.5.3 Etude d'un exemple :

Pour illustrer cette méthode on va traiter un exemple à la main, puis en utilisant Maple dont le programme sera joint en annexe.

Soit le système linéaire suivant :

$$(\mathcal{S}) : \left\{ \begin{array}{ll} x_1 - x_2 + 2x_3 - x_4 = -8 & L_1 \\ 2x_1 - 2x_2 + 3x_3 - 3x_4 = -20 & L_2 \\ x_1 + x_2 + x_3 = -2 & L_3 \\ x_1 - x_2 + 4x_3 + 3x_4 = 4 & L_4 \end{array} \right.$$

dont la matrice associée est :

$$A = \begin{pmatrix} \boxed{1} & -1 & 2 & -1 \\ 2 & -2 & 3 & -3 \\ 1 & 1 & 1 & 0 \\ 1 & -1 & 4 & 3 \end{pmatrix}$$

En faisant les opérations indiquées ci-dessous (le pivot étant l'élément 1) on obtient :

$$\left\{ \begin{array}{rl} x_1 - x_2 + 2x_3 - x_4 = -8 & L_1 \leftarrow L_1 \\ -x_3 - x_4 = -4 & L_2 \leftarrow L_2 - (2/\boxed{1})L_1 \\ 2x_2 - x_3 + x_4 = 6 & L_3 \leftarrow L_3 - (1/\boxed{1})L_1 \\ 2x_3 + 4x_4 = 12 & L_4 \leftarrow L_4 - (1/\boxed{1})L_1 \end{array} \right.$$

C'est un système équivalent à $(\mathcal{S})$ et dont la matrice est :

$$\begin{pmatrix} 1 & -1 & 2 & -1 \\ 0 & \boxed{0} & -1 & -1 \\ 0 & 2 & -1 & 1 \\ 0 & 0 & 2 & 4 \end{pmatrix}$$

Ici, la procédure est interrompue par le fait que le deuxième pivot d'élimination vaut 0 et il est donc impossible d'éliminer les termes se situant sous ce pivot. Mais comme on dispose des opérations élémentaires on peut intervertir la ligne 2 et la ligne 3, et notre système devient :

$$\left\{ \begin{array}{rl} x_1 - x_2 + 2x_3 - x_4 = -8 & \\ 2x_2 - x_3 - x_4 = 6 & L_2 \leftrightarrow L_3 \\ -x_3 - x_4 = -4 & \\ 2x_3 + 4x_4 = 12 & \end{array} \right.$$

Sa matrice associée est :

$$\begin{pmatrix} 1 & -1 & 2 & -1 \\ 0 & 2 & -1 & 1 \\ 0 & 0 & \boxed{-1} & -1 \\ 0 & 0 & 2 & 4 \end{pmatrix}$$

Il ne nous reste plus qu'à éliminer la variable x_3 dans la dernière ligne par une combinaison linéaire de la troisième et la quatrième ligne ; On en déduit le système suivant :

$$\left\{ \begin{array}{rl} x_1 - x_2 + 2x_3 - x_4 = -8 & \\ 2x_2 - x_3 + x_4 = 6 & \\ -x_3 - x_4 = -4 & \\ 2x_4 = 4 & L_4 \leftarrow L_4 - (2/\boxed{-1})L_3 \end{array} \right.$$

Comme le dernier système est triangulaire supérieur, on peut le résoudre facilement grâce à la méthode de la remontée.

$L_4 : x_4 = \dfrac{4}{2} = 2.$

$L_3 : -x_3 - 2 = -4 \Rightarrow x_3 = 2.$

$L_2 : 2x_2 = 6 \Rightarrow x_2 = 3.$

$L_1 : x_1 - 1 = -8 \Rightarrow x_1 = -7.$

Finalement : $\begin{pmatrix} x_1 & x_2 & x_3 & x_4 \end{pmatrix}^T = \begin{pmatrix} -7 & 3 & 2 & 2 \end{pmatrix}^T.$

2.6 Quantification de l'erreur :

Par conséquent les solutions obtenues numériquement (à l'aide d'un calculateur) ne sont que des valeurs approchées de la solution exacte.
En général, si x la solution exacte du système $Ax = b$ et $\tilde{x}$ la solution approchée calculée, nous avons donc les relations suivantes : $Ax - b = 0$ et $A\tilde{x} - b = r$ où r est appelé le résidu.
Soit z l'erreur commise sur la solution : $x = \tilde{x} + z$.
On montre facilement que z vérifie la relation : $Az + r = 0$.
Ainsi, on peut calculer l'erreur z à partir du résidu r.

3 La méthode de décomposition *LU*

3.1 Le principe de la méthode de décomposition *LU* :

Le principe de la méthode est de se ramener à deux systèmes triangulaires.
Soit le système linéaire suivant : $Ax = b$, où A est une matrice dont tous les mineurs pricipaux sont non nuls, alors pour la résolution on procède par les étapes suivantes :

1. **Décomposition de A :** $A = LU$ avec L est triangulaire inférieure à diagonale unité et U triangulaire supérieure ;
2. **Résolution du système triangulaire inférieur :** $Ly = b$ (descente) ;
3. **Résolution du système triangulaire supérieur :** $Ux = y$ (remontée).

$$Ax = b \xrightarrow[A=LU]{\textbf{Décomposition LU}} LUx = b \rightarrow \begin{cases} Ly = b \\ Ux = y \end{cases}$$

Remarque 3.1. *L'existence de la décomposition LU est assurée par le **théorème 1.4**.*

3.2 L'algorithme de la méthode de décomposition *LU* :

3.2.1 Description de l'algorithme :

Algorithme de factorisation LU :
Entrée : une matrice carrée A d'ordre n dont tous les mineurs principaux sont non nuls

Initialisation : $U = A$ et $L = I_n$
pour $k = 1$ à $n-1$ faire
    $p = U[k,k]$
    pour $i = k+1$ à n faire
        $q = U[i,k]$
        $U[i,k] = 0$
        $L[i,k] = \frac{q}{p}$
        pour $j = k+1$ à n faire
            $U[i,j] = U[i,j] - \frac{q}{p}U[k,j]$
        fin pour
    fin pour
fin pour

Sorties : la matrice triangulaire inférieure L et la matrice triangulaire supérieure U

Algorithme de résolution du système triangulaire inférieur :
Entrées : la matrice triangulaire inférieure L et le vecteur b
Initialisation : $y[1:n] = 0$ $y[1] = b[1]$ pour $i = 2$ à n faire $y[i] = b[i] - \sum_{j=1}^{i-1} L[i,j]y[j]$ fin pour
Sortie : le vecteur y

Algorithme de résolution du système triangulaire supérieur :
Entrées : la matrice triangulaire supérieure U et le vecteur y
Initialisation : $x[1:n]=0$ $x[n]=\dfrac{y[n]}{U[n,n]}$ pour $i=n-1$ à 1 faire $\quad x[i]=\dfrac{1}{U[i,i]}\left(y[i]-\sum_{j=i+1}^{n}U[i,j]x[j]\right)$ fin pour
Sortie : le vecteur x

3.2.2 Etude d'un exemple :

Pour illustrer cette méthode on va traiter un exemple à la main, puis en utilisant Maple dont le programme sera joint en annexe.

Soit la matrice

$$A=\begin{pmatrix}1&4&5\\2&10&16\\3&20&42\end{pmatrix}$$

qui est bien une matrice inversible dont tous les mineurs principaux sont non nuls.

Résoudre le système linéaire $Ax=b$ avec $b=\begin{pmatrix}-1&2&19\end{pmatrix}^T$ et $x=\begin{pmatrix}x_1&x_2&x_3\end{pmatrix}^T$ se ramène par la méthode de décomposition LU à procéder par trois étapes :

1. Calculons d'abord les deux matrices L et U de la factorisation de A.
   D'après ce qui précède $A=LU$ avec L matrice triangulaire inférieure de diagonale unité et U matrice triangulaire supérieure :

$$L=\begin{pmatrix}1&0&0\\l_{2,1}&1&0\\l_{3,1}&l_{3,2}&1\end{pmatrix}\ et\ \ U=\begin{pmatrix}u_{1,1}&u_{1,2}&u_{1,3}\\0&u_{2,2}&u_{2,3}\\0&0&u_{3,3}\end{pmatrix}$$

   Alors :
   - $1\times u_{1,1}=1\Rightarrow u_{1,1}=1$
   - $1\times u_{1,2}=4\Rightarrow u_{1,2}=4$
   - $1\times u_{1,3}=5\Rightarrow u_{1,3}=5$
   - $l_{2,1}\times 1=2\Rightarrow l_{2,1}=2$
   - $2\times 4+1\times u_{2,2}=10\Rightarrow u_{2,2}=2$
   - $2\times 5+1\times u_{2,3}=16\Rightarrow u_{2,3}=6$
   - $1\times l_{3,1}=3\Rightarrow l_{3,1}=3$
   - $3\times 4+l_{3,2}\times 2=20\Rightarrow l_{3,2}=4$
   - $3\times 5+4\times 6+1\times u_{3,3}=42\Rightarrow u_{3,3}=3$

   On obtient :

$$L=\begin{pmatrix}1&0&0\\2&1&0\\3&4&1\end{pmatrix}\ et\ U=\begin{pmatrix}1&4&5\\0&2&6\\0&0&3\end{pmatrix}.$$

2. Cherchons $y=\begin{pmatrix}y_1&y_2&y_3\end{pmatrix}^T$ la solution du système linéaire $Ly=b$.

$$Ly=b\Leftrightarrow\begin{cases}y_1=-1\\2y_1+y_2=2\\3y_1+4y_2+y_3=19\end{cases}\Leftrightarrow\begin{cases}y_1=-1\\y_2=2-2y_1=4\\y_3=19-3y_1-4y_2=6\end{cases}$$

   Donc : $y=\begin{pmatrix}-1&4&6\end{pmatrix}^T$.

3. Après avoir calculer le vecteur y il reste à trouver la solution du système linéaire

$$Ux=y\Leftrightarrow\begin{cases}x_1+4x_2+5x_3=-1\\\qquad 2x_2+6x_3=4\\\qquad\qquad 3x_3=6\end{cases}$$

On obtient : $\begin{cases} x_1 = -1 - 4x_2 - 5x_3 = -5 \\ x_2 = 2 - 3x_3 = -4 \\ x_3 = 2 \end{cases}$

Finalement : $x = \begin{pmatrix} 5 & -4 & 2 \end{pmatrix}^T$.

3.2.3 Complexité de l'algorithme :

Une décomposition LU pour la résolution d'un système linéaire nécessite :

- $\dfrac{n^3 - n}{3}$ multiplications ;
- $\dfrac{2n^3 - 3n^2 + n}{6}$ additions.

à l'étape de décomposition en un produit LU.

Soit donc au total de l'ordre de $\dfrac{n^3}{3}$, de plus la remontée et la descente nécessitent de l'ordre de n^2. On note que pour n grand on néglige les puissances de n inférieures à 3, ce qui permet de dire que le gros travail se trouve dans la décomposition elle-même.

3.3 Analyse de sensibilité :

Considérons le système linéaire suivant

$$\begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 9 & 9 & 10 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 32 \\ 23 \\ 33 \\ 31 \end{pmatrix}$$

dont la solution exacte est $x = \begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix}^T$, mais en changeant un peu le vecteur b tel que $b + \delta b = \begin{pmatrix} 32,1 & 22,9 & 33,1 & 30,9 \end{pmatrix}^T$ on trouve $\tilde{x} = \begin{pmatrix} 9,2 & -12,6 & 4,5 & -1,1 \end{pmatrix}^T$.

L'erreur relative de l'ordre $\dfrac{||\delta b||_\infty}{||b||_\infty} \simeq 3.10^{-3}$ sur b a entraîné une erreur relative de l'ordre $\dfrac{||\delta x||_\infty}{||x||_\infty} = \dfrac{||x - \tilde{x}||_\infty}{||x||_\infty} = 13,6$ sur la solution x qui est 4488 fois plus.

En général pour un système linéaire $Ax = b$ avec A inversible, de très petites variations des données sur le vecteur b ou\et sur la matrice A peuvent modifier complètement la solution.

On obtient donc pour un système perturbé $A(x + \delta x) = b + \delta b$ une perturbation qu'on peut écrire : $\delta x = A^{-1}\delta b$

D'où les relations avec des normes subordonnées appropriées : $\|\delta x\| \leq \|A^{-1}\|\|\delta b\|$ et $\|A\|\|x\| \geq \|b\|$

Ainsi on obtient la majoration suivante :

$$\frac{\|\delta x\|}{\|x\|} \leq \|A\|\|A^{-1}\|\frac{\|\delta b\|}{\|b\|}.$$

Par conséquent, le rapport d'amplification des erreurs relatives est au plus égal à $\|A\|\|A^{-1}\|$ (c-à-d que la majoration est optimale). Dans le cas où la matrice A varie, il s'agit de comparer les solutions exactes x et $(x + \Delta x)$ des deux systèmes $Ax = b$ et $(A + \delta A)(x + \delta x) = b$.

On a donc : $\|\delta x\| \leq \|A^{-1}\|\|\delta A\|\|x + \delta x\|$.

Et par suite : $\dfrac{\|\delta x\|}{\|x + \delta x\|} \leq \|A\|\|A^{-1}\|\dfrac{\|\delta A\|}{\|A\|}$.

La quantité [10] $\|A\|\|A^{-1}\|$ apparaît à nouveau pour contrôler l'amplification des erreurs relatives.

10. le nombre conditionnement de la matrice inversible A qui se note $cond(A)$.

4 La méthode de Cholesky

C'est qui Cholesky ?

André-Louis Cholesky (15 octobre 1875 - 31 août 1918) était un mathématicien et un militaire français. Il a effectué sa carrière dans les services géographiques et topographiques de l'armée. C'est pour résoudre des problèmes de moindres carrés posés par la géodésie que Cholesky inventa la décomposition qui porte aujourd'hui son nom.

André-Louis Cholesky (1875-1918)

4.1 Le principe de la méthode de Cholesky :

Soit le système linéaire suivant : $Ax = b$, où A est une matrice symétrique définie positive, alors pour la résolution on procède par les étapes suivantes :

1. **Décomposition de A :** $A = LL^T$ avec L est triangulaire inférieure à éléments diagonaux positifs ;
2. **Résolution du système triangulaire inférieur :** $Ly = b$ (descente) ;
3. **Résolution du système triangulaire supérieur :** $L^T x = y$ (remontée).

$$Ax = b \xrightarrow[A=LL^T]{\textbf{Cholesky}} LL^T x = b \rightarrow \begin{cases} Ly = b \\ L^T x = y \end{cases}$$

Remarque 4.1. *L'existence de la décomposition de A est garantie par le **théorème 1.5**.*

4.2 L'algorithme de la méthode de Cholesky :

La méthode pour trouver la décomposition de Cholesky est d'écrire $A = LL^T$ et d'identifier les cœfficients de façon progressive.

4.2.1 Description de l'algorithme :

Algorithme de décomposition de Cholesky :
Entrée : une matrice symétrique définie positive A d'ordre n

Initialisation : $L[1:n, 1:n] = 0$
$L[1,1] = \sqrt{A[1,1]}$
pour $i = 2$ à n faire
    $L[i,1] = \dfrac{A[i,1]}{L[1,1]}$
fin pour
pour $j = 2$ à n faire
    $L[j,j] = \sqrt{A[j,j] - \sum_{k=1}^{j-1} L[j,k]^2}$
    pour $i = j$ à n faire
        $L[i,j] = \dfrac{1}{L[j,j]} \left(A[i,j] - \sum_{k=1}^{j-1} L[i,k] L[j,k] \right)$
    fin pour
fin pour

Sortie : la matrice triangulaire inférieure L

Algorithme de résolution du système triangulaire inférieur :
Entrées : la matrice triangulaire inférieure L et le vecteur b
Initialisation : $y[1:n]=0$ $y[1]=\dfrac{b[1]}{L[1,1]}$ pour $i=2$ à n faire $\quad y[i]=\dfrac{1}{L[i,i]}\left(b[i]-\sum_{j=1}^{i-1}L[i,j]y[j]\right)$ fin pour
Sortie : le vecteur y

Algorithme de résolution du système triangulaire supérieur :
Entrées : la matrice triangulaire inférieure L et le vecteur y
Initialisation : $x[1:n]=0$ et $U=L^T$ $x[n]=\dfrac{y[n]}{U[n,n]}$ pour $i=n-1$ à 1 faire $\quad x[i]=\dfrac{1}{U[i,i]}\left(y[i]-\sum_{j=i+1}^{n}U[i,j]x[j]\right)$ fin pour
Sortie : le vecteur x

4.2.2 Etude d'un exemple :

Pour illustrer cette méthode on va traiter un exemple à la main, puis en utilisant Maple dont le programme sera joint en annexe.
Soit la matrice

$$A=\begin{pmatrix}4&-6&8&2\\-6&10&-15&-3\\8&-15&26&-1\\2&-3&-1&62\end{pmatrix}$$

qui est bien une matrice symétrique définie positive.
Résoudre le système linéaire $Ax=b$ avec $b=\begin{pmatrix}14&-10&9&45\end{pmatrix}^T$ et $x=\begin{pmatrix}x_1&x_2&x_3&x_4\end{pmatrix}^T$ se ramène par la méthode de Cholesky à procéder par trois étapes :

1. Calculons d'abord L la factorisation de A.
   D'après ce qui précède $A=LL^T$ avec L la matrice triangulaire inférieure suivante :

$$L=\begin{pmatrix}l_{1,1}&0&0&0\\l_{2,1}&l_{2,2}&0&0\\l_{3,1}&l_{3,2}&l_{3,3}&0\\l_{4,1}&l_{4,2}&l_{4,3}&l_{4,4}\end{pmatrix}$$

   Alors :
   - $l_{1,1}=\sqrt{4}\Rightarrow l_{1,1}=2$
   - $l_{2,1}=\dfrac{-6}{2}\Rightarrow l_{2,1}=-3$
   - $l_{3,1}=\dfrac{8}{2}\Rightarrow l_{3,1}=4$
   - $l_{4,1}=\dfrac{2}{2}\Rightarrow l_{4,1}=1$
   - $(-3)^2+l_{2,2}^2=10\Rightarrow l_{2,2}=1$
   - $(-3)\times 4+l_{3,2}=-15\Rightarrow l_{3,2}=-3$
   - $(-3)+l_{4,2}=-3\Rightarrow l_{4,2}=0$
   - $4^2+(-3)^2+l_{3,3}^2=26\Rightarrow l_{3,3}=1$
   - $4+1\times l_{4,3}=-1\Rightarrow l_{4,3}=-5$
   - $1^2+0+(-5)^2+l_{4,4}^2=62\Rightarrow l_{4,4}=6$

On obtient :

$$L = \begin{pmatrix} 2 & 0 & 0 & 0 \\ -3 & 1 & 0 & 0 \\ 4 & -3 & 1 & 0 \\ 1 & 0 & -5 & 6 \end{pmatrix}.$$

2. Cherchons $y = \begin{pmatrix} y_1 & y_2 & y_3 & y_4 \end{pmatrix}^T$ la solution du système linéaire $Ly = b$.

$$Ly = b \Leftrightarrow \begin{cases} 2y_1 = 14 \\ -3y_1 + y_2 = -10 \\ 4y_1 - 3y_2 + y_3 = 9 \\ y_1 - 5y_3 + 6y_4 = 45 \end{cases} \Leftrightarrow \begin{cases} y_1 = 7 \\ y_2 = -10 + 3y_1 = 11 \\ y_3 = 9 - 4y_1 + 3y_2 = 14 \\ y_4 = \dfrac{45 - y_1 + 5y_3}{6} = 18 \end{cases}$$

Donc : $y = \begin{pmatrix} 7 & 11 & 14 & 18 \end{pmatrix}^T$.

3. Après avoir calculer le vecteur y il reste à trouver $x = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 \end{pmatrix}^T$ la solution du système linéaire

$$L^T x = y \Leftrightarrow \begin{cases} 2x_1 - 3x_2 + 4x_3 + x_4 = 7 \\ \qquad\quad x_2 - 3x_3 \qquad\quad = 11 \\ \qquad\qquad\qquad x_3 - 5x_4 = 14 \\ \qquad\qquad\qquad\qquad 6x_4 = 18 \end{cases}$$

On obtient : $\begin{cases} x_1 = \dfrac{7 + 3x_2 - 4x_3 - x_4}{2} = 91 \\ x_2 = 11 + 3x_3 = 98 \\ x_3 = 14 + 5x_4 = 29 \\ x_4 = 3 \end{cases}$

Finalement : $x = \begin{pmatrix} 91 & 98 & 29 & 3 \end{pmatrix}^T$.

4.2.3 Complexité de l'algorithme :

On va calculer dans cette section le nombre d'opérations nécessaire à la factorisation de la matrice en premier temps, puis le nombre d'opérations nécessaire à la résolution des deux systèmes triangulaires. On considère que chaque opération, prend le même temps de calcul.
Comptons maintenant le nombre d'opérations élémentaires dans la méthode de résolution de Cholesky.

1. **Décomposition. Le calcul des éléments de la matrice L nécessite :**
   - $\dfrac{n^3 - n}{6}$ multiplications,
   - $\dfrac{n^3 - n}{6}$ additions,
   - $\dfrac{n(n-1)}{2}$ divisions,
   - n extractions de racines carrées.

   Donc la complexité de la décomposition est en $\mathcal{O}\left(\dfrac{n^3}{3}\right)$.

2. **Résolution des deux systèmes linéaires triangulaires.** Comme déjà vu, ces résolutions nécessitent :
   - $n(n-1)$ multiplications,
   - $n(n-1)$ additions,
   - $2n$ divisions.

   Au total, la méthode de Cholesky nécessite de l'ordre de :
   - $\dfrac{n^3}{6}$ multiplications,
   - $\dfrac{n^3}{6}$ additions,
   - $\dfrac{n^2}{2}$ divisions,
   - n extractions de racines carrées.

4.3 Résolution des systèmes linéaires : cas non symétrique

Soit $A \in \mathcal{M}_n(\mathbb{R})$ inversible, non symétrique.
On cherche à calculer $x \in \mathbb{R}^n$ solution du système linéaire $Ax = b$ par la méthode de Cholesky.
On a : $Ax = b \iff A^T Ax = A^T b$ car $\det(A^T) = \det(A) \neq 0$.
D'après la ***proposition 1.11*** la matrice $A^T A$ est bien symétrique définie positive, alors il suffit de calculer $A^T A$ et $A^T b$ puis résoudre le système linéaire $A^T Ax = A^T b$ par la méthode de Cholesky.

5 La méthode de Householder

La biographie de Householder :

Alston Scott Householder (1904-1993) est un mathématicien spécialisé en mathématiques appliquées à la biologie et en analyse numérique. On lui doit les transformations de Householder et les méthodes de Householder pour la résolution des équations algébriques ou linéaires. Il a contribué au développment de plusieurs revues mathématiques, il a été président de l'*American Mathematical Society*.

Alston Scott Householder (1904-1993).

5.1 Le principe de la méthode de Householder :

Soit le système linéaire suivant : $Ax = b$, où A est une matrice inversible, alors pour la résolution on procède par les étapes suivantes :

1. **Décomposition de A :** $A = QR$ avec Q est orthogonale et R triangulaire supérieure à éléments diagonaux positifs ;
2. **Résolution du système :** $Qy = b$;
3. **Résolution du système triangulaire supérieur :** $Rx = y$ (remontée).

$$Ax = b \xrightarrow[A=QR]{\textbf{Householder}} QRx = b \rightarrow \begin{cases} Qy = b \\ Rx = y \end{cases}$$

Remarque 5.1. ***La résolution du système à matrice orthogonale $Qy = b$:***

$$Qy = b \iff y = Q^{-1}b = Q^T b$$

donc la résolution de ce système se fait par un produit d'une matrice par un vecteur colonne.

5.2 L'algorithme de La méthode de Householder :

5.2.1 Description de l'algorithme :

Algorithme de décomposition de Householder :

Entrée : une matrice carrée A d'ordre n

Initialisation : $H = I_n$
pour $k = 1$ à $n-1$ faire
  $\alpha \leftarrow \sqrt{\sum_{i=k}^{n} A[i,k]^2}$
  $\beta \leftarrow \alpha^2 - \alpha A[k,k]$
  $v[k] \leftarrow A[k,k] - \alpha$
  pour $i = k+1$ à n faire
    $v[i] \leftarrow A[i,k]$
  fin pour
  pour $j = k$ à n faire
    $c \leftarrow \frac{1}{\beta}\sum_{i=k}^{n} v[i]A[i,j]$
    pour $i = k$ à n faire
      $A[i,j] \leftarrow A[i,j] - cv[i]$
    fin pour
  fin pour
  pour $j = 1$ à n faire
    $c \leftarrow \frac{1}{\beta}\sum_{i=k}^{n} v[i]H[i,j]$
    pour $i = k$ à n faire
      $H[i,j] \leftarrow H[i,j] - cv[i]$
    fin pour
  fin pour
fin pour
$Q \leftarrow H^T$ et $R \leftarrow A$

Sorties : une matrice orthogonale Q et une matrice triangulaire supérieure R.

Algorithme de résolution du système à matrice orthogonale :

Entrées : la matrice orthogonale Q et le vecteur b

Initialisation : $y[1:n] = 0$
pour $i = 1$ à n faire
  $y[i] = \sum_{j=1}^{n} Q[j,i]b[j]$
fin pour

Sortie : le vecteur y

Algorithme de résolution du système triangulaire supérieur :

Entrées : la matrice triangulaire supérieure R et le vecteur y

Initialisation : $x[1:n] = 0$
$x[n] = \frac{y[n]}{R[n,n]}$
pour $i = n-1$ à 1 faire
  $x[i] = \frac{1}{R[i,i]}\left(y[i] - \sum_{j=i+1}^{n} R[i,j]x[j]\right)$
fin pour

Sortie : le vecteur x

5.2.2 Etude d'un exemple :

Pour illustrer cette méthode on va traiter un exemple à la main, puis en utilisant Maple dont le programme sera joint en annexe.

Soit le système linéaire $Ax = b$ avec

$$A = \begin{pmatrix} -4 & 20 & 35 & 5 \\ -4 & -30 & -15 & 55 \\ -8 & 40 & -80 & -65 \\ 23 & -15 & 30 & 15 \end{pmatrix}, \quad b = \begin{pmatrix} -29 \\ 121 \\ -58 \\ 23 \end{pmatrix} \quad et \;\; x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}.$$

Pour la résolution de ce système par la méthode de décomposition QR on procède par les trois étapes suivantes :

1. **La décomposition QR de la matrice A :**
   On notera dans la suite $[a]$ la matrice colonne des coordonnées d'un vecteur a de $\mathbb{R}^n$.
   (i) Traitement de la première colonne :
   Le premier vecteur colonne de A est $u_1 = (-4, -4, -8, 23)$, alors $\|u_1\| = \sqrt{4^2 + 4^2 + 8^2 + 23^2} = 25$.
   On va transformer u_1 en $v = (25, 0, 0, 0)$ au moyen de la réflexion h_1 par rapport à l'hyperplan orthogonal au vecteur $a_1 = u_1 - \|u_1\|e_1 = (-29, -4, -8, 23)$.
   Le carré de la norme de a_1 est

$$\|a_1\|^2 = 2\|u_1\|^2 - 2\|u_1\| < u_1, e_1 >= 2\|u_1\|^2 + 8\|u_1\| = 2 \times 25 \times 29 = 2 \times 725.$$

   La matrice de Householder associée est

$$H_1 = I_4 - \frac{2}{\|a_1\|^2}[a_1]^T[a_1] = I_4 - \frac{1}{725}\begin{pmatrix} -29 \\ -4 \\ -8 \\ 23 \end{pmatrix}(-29 \;\; -4 \;\; -8 \;\; 23)$$

$$= I_4 - \frac{1}{725}\begin{pmatrix} 841 & 116 & 232 & -667 \\ 116 & 16 & 32 & -92 \\ 232 & 32 & 64 & -184 \\ -667 & -92 & -184 & 529 \end{pmatrix} = \frac{1}{725}\begin{pmatrix} -116 & -116 & -232 & 667 \\ -116 & 709 & -32 & 92 \\ -232 & -32 & 661 & 184 \\ 667 & 92 & 184 & 196 \end{pmatrix}$$

   On en déduit :

$$A_1 = H_1 A = \begin{pmatrix} -116 & -116 & -232 & 667 \\ -116 & 709 & -32 & 92 \\ -232 & -32 & 661 & 184 \\ 667 & 92 & 184 & 196 \end{pmatrix}\begin{pmatrix} -4 & 20 & 35 & 5 \\ -4 & -30 & -15 & 55 \\ -8 & 40 & -80 & -65 \\ 23 & -15 & 30 & 15 \end{pmatrix}$$

$$= \frac{25}{29}\begin{pmatrix} 29 & -29 & 58 & 29 \\ 0 & -42 & -15 & 167 \\ 0 & 32 & -88 & -69 \\ 0 & 24 & 21 & -1 \end{pmatrix}$$

   (ii) Traitement de la deuxième colonne :
   Le deuxième vecteur colonne de A_1 est $u_2 = \frac{25}{29}(-29, -42, 32, 24)$.
   Posons $w = \frac{25}{29}(0, -42, 32, 24)$, alors $\|w\| = \frac{25}{29}\sqrt{42^2 + 32^2 + 24^2} = 50$.
   On va appliquer la réflexion h_2 par rapport à l'hyperplan orthogonal à $w - \|w\|e_2 = \frac{100}{29}(0, -25, 8, 6)$.
   Pour simplifier on choisit $a_2 = (0, -25, 8, 6)$, alors $\|a_2\|^2 = 25^2 + 8^2 + 6^2 = 725$.
   La matrice de Householder associée est

$$H_2 = I_4 - \frac{2}{\|a_2\|^2}[a_2]^T[a_2] = I_4 - \frac{2}{725}\begin{pmatrix} 0 \\ -25 \\ 8 \\ 6 \end{pmatrix}(0 \;\; -25 \;\; 8 \;\; 6)$$

$$= I_4 - \frac{2}{725}\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 625 & -200 & -150 \\ 0 & -200 & 64 & 48 \\ 0 & -150 & 48 & 36 \end{pmatrix} = \frac{1}{725}\begin{pmatrix} 725 & 0 & 0 & 0 \\ 0 & -525 & 400 & 300 \\ 0 & 400 & 597 & -96 \\ 0 & 300 & -96 & 653 \end{pmatrix}$$

On en déduit $A_2 = H_2 A_1 = H_2 H_1 A$:

$$A_2 = \frac{1}{725}\begin{pmatrix} 725 & 0 & 0 & 0 \\ 0 & -525 & 400 & 300 \\ 0 & 400 & 597 & -96 \\ 0 & 300 & -96 & 653 \end{pmatrix} \frac{25}{29} \begin{pmatrix} 29 & -29 & 58 & 29 \\ 0 & -42 & -15 & 167 \\ 0 & 32 & -88 & -69 \\ 0 & 24 & 21 & -1 \end{pmatrix}$$

$$= \begin{pmatrix} 25 & -25 & 50 & 25 \\ 0 & 50 & -25 & -75 \\ 0 & 0 & -72 & -17 \\ 0 & 0 & 21 & 31 \end{pmatrix}$$

On trouve alors la matrice $H_1 H_2$:

$$H_1 H_2 = \frac{1}{725}\begin{pmatrix} -116 & -116 & -232 & 667 \\ -116 & 709 & -32 & 92 \\ -232 & -32 & 661 & 184 \\ 667 & 92 & 184 & 196 \end{pmatrix} \frac{1}{725} \begin{pmatrix} 725 & 0 & 0 & 0 \\ 0 & -525 & 400 & 300 \\ 0 & 400 & 597 & -96 \\ 0 & 300 & -96 & 653 \end{pmatrix}$$

$$= \frac{1}{625}\begin{pmatrix} -100 & 200 & -296 & 503 \\ -100 & -425 & 304 & 328 \\ -200 & 400 & 433 & 56 \\ 575 & 100 & 152 & 164 \end{pmatrix}$$

(iii) Traitement de la troisième colonne :
Le troisième vecteur colonne de la matrice A_2 est $u_3 = (50, -25, -72, 21)$.
Posons $w = (0, 0, -72, 21)$, alors $\|w\| = \sqrt{72^2 + 21^2} = 75$.
On va appliquer la réflexion h_3 par rapport à l'hyperplan ortogonal à $a_3 = w - \|w\| e_3 = (0, 0, -147, 21)$, alors $\|a_3\|^2 = 147^2 + 21^2 = 22050$.
La matrice de Householder associée est

$$H_3 = I_4 - \frac{2}{\|a_3\|^2}[a_3]^T[a_3] = I_4 - \frac{1}{11025}\begin{pmatrix} 0 \\ 0 \\ -147 \\ 21 \end{pmatrix} (0 \;\; 0 \;\; -147 \;\; 21)$$

$$= I_4 - \frac{1}{11025}\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 21609 & -3087 \\ 0 & 0 & -3087 & 441 \end{pmatrix} = \frac{1}{25}\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -24 & 7 \\ 0 & 0 & 7 & 24 \end{pmatrix}$$

On en déduit la matrice $R = A_3 = H_3 A_2 = H_3 H_2 H_1 A$:

$$R = \frac{1}{25}\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -24 & 7 \\ 0 & 0 & 7 & 24 \end{pmatrix} \begin{pmatrix} 25 & -25 & 50 & 25 \\ 0 & 50 & -25 & -75 \\ 0 & 0 & -72 & -17 \\ 0 & 0 & 21 & 31 \end{pmatrix} = \begin{pmatrix} 25 & -25 & 50 & 25 \\ 0 & 50 & -25 & -75 \\ 0 & 0 & 75 & 25 \\ 0 & 0 & 0 & 25 \end{pmatrix}$$

Enfin $Q = H_3 H_2 H_1$:

$$Q = \frac{1}{625}\begin{pmatrix} -100 & 200 & -296 & 503 \\ -100 & -425 & 304 & 328 \\ -200 & 400 & 433 & 56 \\ 575 & 100 & 152 & 164 \end{pmatrix} \frac{1}{25}\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -24 & 7 \\ 0 & 0 & 7 & 24 \end{pmatrix}$$

$$= \frac{1}{25}\begin{pmatrix} -4 & 20 & 35 & 5 \\ -4 & -30 & -15 & 55 \\ -8 & 40 & -80 & -65 \\ 23 & -15 & 30 & 15 \end{pmatrix}$$

La décomposition de A est alors terminée :

$$\underbrace{\begin{pmatrix} -4 & 20 & 35 & 5 \\ -4 & -30 & -15 & 55 \\ -8 & 40 & -80 & -65 \\ 23 & -15 & 30 & 15 \end{pmatrix}}_{\text{A inversible}} = \underbrace{\frac{1}{25}\begin{pmatrix} -4 & 8 & 17 & 16 \\ -4 & -17 & -15 & 16 \\ -8 & 16 & -16 & 7 \\ 23 & 4 & -4 & 8 \end{pmatrix}}_{\text{Q orthogonale}} \underbrace{\begin{pmatrix} 25 & -25 & 50 & 25 \\ 0 & 50 & -25 & -75 \\ 0 & 0 & 75 & 25 \\ 0 & 0 & 0 & 25 \end{pmatrix}}_{\text{R triangulaire}}$$

2. **Résolution du système $Qy = b$:** Cherchons y la solution du système linéaire $Qy = b$.

$$Qy = b \quad \Leftrightarrow \quad y = Q^{-1}b \quad \Leftrightarrow \quad y = Q^T b \ \textit{ car la matrice } Q \textit{ est orthogonale},$$

alors :

$$y = \frac{1}{25}\begin{pmatrix} -4 & -4 & -8 & 23 \\ 8 & -17 & 16 & 4 \\ 17 & -8 & -16 & -4 \\ 16 & 16 & 7 & 8 \end{pmatrix}\begin{pmatrix} -29 \\ 121 \\ -58 \\ 23 \end{pmatrix}$$

Donc : $y = \begin{pmatrix} 25 & -25 & -25 & 50 \end{pmatrix}^T$.

3. **Résolution du système triangulaire supérieur $Rx = y$:** Après avoir calculer le vecteur y il reste à trouver x la solution du système linéaire $Rx = y$:

$$Rx = y \quad \Leftrightarrow \quad \begin{cases} 25x_1 - 25x_2 + 50x_3 + 25x_4 = 25 \\ \qquad 50x_2 - 25x_3 - 75x_4 = -25 \\ \qquad\qquad 75x_3 + 25x_4 = -25 \\ \qquad\qquad\qquad 25x_4 = 50 \end{cases}$$

On obtient :

$$\begin{cases} x_1 = 1 + x_2 - 2x_3 - x_4 \\ x_2 = \dfrac{-1 + x_3 + 3x_4}{2} \\ x_3 = \dfrac{-1 - x_4}{3} \\ x_4 = \dfrac{50}{25} = 2 \end{cases}$$

Finalement : $x = \begin{pmatrix} 1 & 0 & -1 & 2 \end{pmatrix}^T$.

5.2.3 Complexité de l'algorithme :

Le coût de cette méthode pour une matrice carrée d'ordre n est en : $\frac{4n^3}{3}$. Ce coût est relativement élevé, cependant la méthode de Householder présente l'avantage considérable d'être beaucoup plus stable numériquement, en limitant les divisions par des nombres petits.

Conclusion

Nous venons donc de voir quatres méthodes directes de résolution des systèmes linéaires qu'à travers notre rapport, nous les avons étudié théoriquement (l'existence et l'unicité des décompositions matricielles). Puis nous avons mis la lumière sur le côté pratique en donnant les algorithmes de ces méthodes avec leurs complexités et des programmes en Maple pour chaque méthode.
Nous donnons ci-après un tabeau récapitulatif résumant notre travail et dans lequel on compare le coût des quatres méthodes en citant quelques avantages et inconvénients :

La méthode	**Le coût**	**Avantages**	**Inconvénients**
L'élimination de Gauss	$\mathcal{O}\left(\frac{2n^3}{3}\right)$	– Résolution d'un seul système linéaire (comparaison avec les autres méthodes) ; – Utilisation des opérations élémentaires d'additions et multiplications.	– On change la structure de A ; – Blocage dans le cas d'un pivot nul sans pivotement ; – L'implémentation sur machine donne généralement de mauvais résultats à cause des erreurs d'arrondi qui s'accumulent.
La factorisation LU	$\mathcal{O}\left(\frac{n^3}{3}\right)$	– Cette méthode est très utile lors de la résolution d'une suite de systèmes de même matrice A où seul le vecteur b change ; – Utilisation des opérations élémentaires d'additions et multiplications.	– La résolution d'un seul système linéaire nécessite la résolution de deux systèmes linéaires triangulaires de matrices L et U (stockage des deux matrices L et U) ; – Particularité de la matrice carrée A (mineurs principaux non nuls).
La factorisation de Cholesky	$\mathcal{O}\left(\frac{n^3}{6}\right)$	– Cette méthode est très utile lors de la résolution d'une suite de systèmes de même matrice A où seul le vecteur b change ; – Elle nécessite le calcul d'une seule matrice L ; – Elle demande une place mémoire deux fois inférieure à celles de la factorisation LU car on ne doit stocker que L.	– Elle ne permet pas la résolution d'un système linéaire quelconque (particularité de la matrice A carrée symétrique et définie positive) ; – La résolution d'un seul système linéaire nécessite la résolution de deux systèmes linéaires triangulaires de matrices L et L^T.
La factorisation de Householder	$\mathcal{O}\left(\frac{4n^3}{3}\right)$	– Cette méthode est très utile lors de la résolution d'une suite de systèmes de même matrice A où seul le vecteur b change ; – Résoudre le système $Qy = b$ est facile ; – Plus stable numériquement.	– La résolution d'un seul système linéaire nécessite la résolution de deux systèmes linéaires triangulaires de matrices Q et R (stockage des deux matrices Q et R) ; – La quantité de mémoire et de calculs requis.

D'après le tableau on constate que les principaux avantages d'une méthode directe sont la précision de la solution et le fait qu'une fois la matrice A est factorisée, on peut résoudre le système avec différents seconds membres.
On remarque aussi que la factorisation de Cholesky est la moins coûteuse parmi les autres, et que celle de Householder est la plus coûteuse cependant elle reste la plus stable du point de vue algorithmique.

Annexe :
Implémentation en Maple

Méthode de Gauss

Résolution dans le cas d'un pivot non nul :

```
>restart;with(linalg):
 Gauss1:=proc(A,b)
 local U,v,i,j,k,p,n,X;
 n:=coldim(A);
 U:=matrix(n,n);
 U:=copy(A);
 v:=copy(b);
 X:=vector(n);
 for k from 1 to n-1 do
 for i from k+1 to n do
 if (U[k,k]=0) then error (`pivot nul`) fi;
 p:=U[i,k]/U[k,k];
 for j from k+1 to n do
 U[i,j]:=U[i,j]-U[k,j]*p;
 od;
 v[i]:=v[i]-v[k]*p;
 U[i,k]:=0;
 od;
 od;
 X[n]:=v[n]/U[n,n];
 for i from n-1 to 1 by -1 do
 X[i]:=(v[i]-add(U[i,j]*X[j],j=i+1..n))/U[i,i];
 od;
 evalm(X):
 end proc:
```

L'appel de la procédure précédente sur l'exemple traité à la main se fait comme suit :

```
>A:=matrix([[3,5,7],[2,10,6],[1,2,3]]); b:=[101,134,40]; X:=Gauss1(A,b);
```

Résolution dans le cas d'un pivot nul :

```
>with(linalg);
 Gauss2:=proc(A,b)
 local AA, n, i, j, p, max;
 n:=coldim(A);
 AA:=augment(A,b);
 for j from 1 to n-1 do
 for i from j+1 to n do
 if AA[j,j]=0 then
 if abs(AA[i+1,j])<abs(AA[i,j]) then
```

```
 max:=AA[i,j];
 evalm(i);
 end if;
 AA:=swaprow(AA,j,i);
 end if;
 p:=-AA[i,j]/AA[j,j];
 AA := addrow(AA,j,i,p);
 end do;
 end do;
 backsub(AA);
 end proc;
```

L'appel de la procédure précédente sur l'exemple traité à la main se fait comme suit :

```
>A:=matrix([[1,-1,2,-1],[2,-2,3,-3],[1,1,1,0],[1,-1,4,3]]); b:=[-8,-20,-2,4];
 X:=Gauss2(A,b);
```

Méthode LU

Décomposition LU d'une matrice :

```
>decomposeLU:=proc(A,n)
 local i,j,k,p,L,U;
 L:=Matrix(1..n,1..n);
 U:=Matrix(1..n,1..n);
 for i from 1 to n do
 L[i,i]:=1;
 end do;
 for i from 1 to n do
 for j from 1 to n do
 U[i,j]:=A[i,j];
 end do;
 end do;
 for k from 1 to n-1 do
 for i from k+1 to n do
 p:=U[i,k]/U[k,k];
 L[i,k]:= p;
 for j from k+1 to n do
 U[i,j]:=U[i,j]-p*U[k,j];
 end do;
 end do;
 end do;
 return(L,U):
 end proc:
```

Résolution du système triangulaire inférieur :

```
>VecteurY:=proc(L,b,n)
 local y,i,som,j;
 y:=Array(1..n);
 y[1]:=b[1]/L[1,1];
 for i from 2 to n do
 som:=0;
 for j from 1 to i-1 do
 som:=som+L[i,j]*y[j] ;
 end do;
 y[i]:=(b[i]-som)/L[i,i];
```

```
 end do;
 return(y):
 end proc:
```

Résolution du système triangulaire supérieur :

```
>VecteurX:=proc(U,y,n)
 local i,som,j,x;
 x:=Array(1..n);
 x[n]:=y[n]/U[n,n];
 for i from n-1 by -1 to 1 do
 som:=0;
 for j from i+1 to n do
 som:=som+U[i,j]*x[j];
 end do:
 x[i]:=(y[i]-som)/U[i,i];
 end do:
 return(x);
 end proc:
```

L'appel de la procédure précédente sur l'exemple traité à la main se fait comme suit :

```
>n:=3; A:=matrix([[1,4,5],[2,10,16],[3,20,42]]);
 LU:=decomposeLU(A,n);
 L:=LU[1]; U:=LU[2];
 Y:=VecteurY(L,b,n);
 X:=VecteurX(U,Y,n);
```

Méthode de Cholesky

Décomposition de Cholesky d'une matrice :

```
>Decompo_Cholesky:=proc(A,n)
 local AA,L,tL,i,j,k,s:
 L:=Matrix(1..n,1..n);
 tL:=Matrix(1..n,1..n);
 for i from 1 to n do
 for j from 1 to n do
 AA[i,j]:=A[i,j];
 end do:
 end do:
 L[1,1]:=sqrt(AA[1,1]);
 for i from 2 to n do
 L[i,1]:=AA[i,1]/L[1,1];
 end do:
 for i from 2 to n-1 do
 s:=0;
 for k from 1 to i-1 do
 s:=s-L[i,k]*L[i,k];
 L[i,i]:=sqrt(AA[i,i]+s);
 end do:
 for j from i+1 to n do
 s:=0;
 for k from 1 to i-1 do
 s:=s-L[j,k]*L[i,k];
 L[j,i]:=(AA[j,i]+s)/L[i,i];
 end do:
```

```
 end do:
 s:=0;
 for k from 1 to n-1 do
 s:=s-L[n,k]*L[n,k];
 L[n,n]:=sqrt(AA[n,n]+s);
 end do:
 end do:
 for i from 1 to n do
 for j from 1 to n do
 tL[i,j]:=L[j,i];
 end do:
 end do:
 return(L,tL):
 end proc:
```

Résolution du système triangulaire inférieur :

```
>Vecteur_Y:=proc(L,b,n)
 local y,i,som,j;
 y:=Array(1..n);
 y[1]:=b[1]/L[1,1];
 for i from 2 to n do
 som:=0;
 for j from 1 to i-1 do
 som:=som+L[i,j]*y[j];
 end do;
 y[i]:=(b[i]-som)/L[i,i];
 end do;
 return(y):
 end proc:
```

Résolution du système triangulaire supérieur :

```
>Vecteur_X:=proc(tL,y,n)
 local i,som,j,x:
 x:=Array(1..n);
 x[n]:=y[n]/tL[n,n];
 for i from n-1 by -1 to 1 do
 som:=0;
 for j from i+1 to n do
 som:=som+tL[i,j]*x[j];
 end do:
 x[i]:=(y[i]-som)/tL[i,i];
 end do:
 return(x);
 end proc:
```

L'appel de la procédure précédente sur l'exemple traité à la main se fait comme suit :

```
>n:=4; A:=Matrix([[4,-6,8,2],[-6,10,-15,-3],[8,-15,26,-1],[2,-3,-1,62]]); b:=[14,-10,9,45];
 DecomChol:=Decompo_Cholesky(A,n):
 L:=DecomChol[1]; tL:=DecomChol[2];
 Y:=Vecteur_Y(L,b,n);
 X:=Vecteur_X(tL,Y,n);
```

A noter que Maple dispose d'une fonction qui peut donner directement la décomposition de Cholesky d'une telle matrice A :

```
>with(linalg): cholesky(A);
```

Méthode de Householder

```
>restart;with(linalg):
>house:=proc(x)
 local n,sigma,beta,v,mu;
 n:=vectdim(x);
 sigma:=add(x[i]^2,i=2..n);
 v:=subs(x[1]=1,x);
 if sigma=0 then beta:=0;
 else mu:=sqrt(x[1]^2+sigma);
 if evalf(x[1])<=0 then v[1]:=x[1]-mu;
 else v[1]:=-sigma/(x[1]+mu); end if;
 end if;
 beta:=2*v[1]^2/(sigma+v[1]^2);
 beta,evalm(v/v[1]);
 end proc:
>QR:=proc(A)
 local p,n,j,beta,v,BB,vv,P,Q,II,B;
 B:=copy(A);
 p:=rowdim(A);
 n:=coldim(A);
 Q:=evalm(array(identity,1..p,1..p));
 for j from 1 to n do
 beta:=house(subvector(B,j .. p,j))[1];
 v:=house(subvector(B,j.. p,j))[2];
 P:=evalm(array(identity,1..p-j+1,1..p-j+1)-beta*(`&*`(v,transpose(v))));
 BB:=`&*`(P,submatrix(B,j..p,j..n));
 copyinto(BB,B,j,j);
 if j=1 then II:=P else
 II:=diag(array(identity,1..j-1,1..j-1),P);
 end if;
 Q:=evalm(Q&*II);
 end do;
 evalm(Q),evalm(B);
 end proc:
>back:=proc(A,b)
 local n,bb,j,i;
 n:=rowdim(A):
 bb:=copy(b):
 if mul(A[j,j],j=1..n)=0 then ERROR(`matrice non inversible`)
 else bb[n]:=bb[n]/A[n,n]:
 for i from n-1 to 1 by -1 do
 bb[i]:=(bb[i]-add(A[i,j]*bb[j],j=i+1..n))/A[i,i];
 end do;
 evalm(bb);
 end if;
 end proc:
>lsq:=proc(A,b)
 local t,n,R,Q,c;
 n:=coldim(A):
 R:=QR(A)[2]:
 Q:=QR(A)[1]:
 t:=submatrix(R,1..n,1..n):
 c:=evalm(transpose(Q) &* b):
 back(t,subvector(convert(c,matrix),1..n,1));
 end proc:
```

L'appel de la procédure précédente sur l'exemple traité à la main se fait comme suit :

```
>A:=matrix([[-4,20,35,5],[-4,-30,-15,55],[-8,40,-80,-65],[23,-15,30,15]]);
 b:=[-29,121,-58,23];
 house(b): QR(A,b): back(A,b): evalf(lsq(A,b));
```

A noter que Maple dispose d'une fonction qui peut donner directement la décomposition QR d'une telle matrice A :

```
>with(linalg): QRDecomposition(A);
```

Comparaison quantitative :

Le bon choix de la méthode directe de résolution consiste à choisir les algorithmes en fonction de la matrice du système à résoudre. Les critères quantitatifs de choix seront :

- La place occupée en mémoire ;
- L'exactitude des résultats (erreurs d'arrondi) :
  `> sol:=linsolve(A,b): erreur:=evalm(abs(x-sol));`
- Le temps de calcul :
  `> t:=time(): l'appel des procédures de la méthode; time()-t;`

Références

[1] P.G. CIARLET : *Introduction à l'analyse numérique matricielle et à l'optimisation.* MASSON, 1988.

[2] Bertrand HAUCHECORNE et Daniel SURATTEAU : *Des mathématiciens de A à Z.* ELLIPES, 1996.

[3] Jean-Pierre Escofier : *Toute l'algèbre de la licence.* DUNOD, 2006.

[4] Lucas Amodei et Jean-Pierre Dedieu : *Analyse numérique matricielle.* DUNOD, 2008.

[5] *Mathématiques appliquées L3 : Cours complet avec 500 tests et exercices corrigés.* PEARSON Education France, 2009.