Algèbre linéaire

Résolution numérique de systèmes linéaires.

1. Systèmes linéaires

On concidère le système linéaire :

$$(1)\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ \vdots \qquad\qquad\qquad\qquad\qquad \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n \end{cases} \qquad X = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \qquad A = (a_{ij})$$

Les inconnues sont les composantes x_i $(i = 1\_n)$.

$(1) \Leftrightarrow Ax = b$ On suppose A inversible ($\det A \neq 0$).

Formules de CRAMER

colonne n°i

$$x_i = \frac{\begin{vmatrix} a_{11} \dots & a_{1i-1} & \boxed{b_1} & a_{1i+1} \dots & a_{1n} \\ \vdots & \vdots & \vdots & & \vdots \\ a_{n1} \dots & a_{ni-1} & \boxed{b_n} & a_{ni+1} \dots & a_{nn} \end{vmatrix}}{\det A} \qquad i \in [\![1,n]\!]$$

Estimons le nombre d'opérations élémentaires $(+, \times, \div)$ nécessaires pour résoudre notre système : on doit calculer $(n+1)$ déterminants puis effectuer n divisions pour calculer les x_i $(i = 1\_n)$.

Pour un déterminant $\begin{cases} n!\,n \text{ multiplications} \\ n!-1 \text{ additions} \end{cases}$

Il en résulte que la résolution de (1) par CRAMER nécessite $\begin{cases} (n+1)n!\,n \text{ multiplications} \\ (n+1)(n!-1) \text{ additions} \\ n \text{ divisions} \end{cases}$

$$\sum op = (n+1)n! - 1$$

or, dans la pratique, nous aurons à résoudre des systèmes avec $n = 50$, $n = 100$ et parfois $\hat{m}$ $n \geq 1000$ (approximation des E.D.P.)

$n = 5 \Rightarrow \sum op = 4319$ $\qquad n = 10 \Rightarrow \sum op \simeq 4 \cdot 10^8$

Il est donc impossible de résoudre de tels systèmes par CRAMER. En analyse numérique, nous pouvons résoudre un système par 2 types de méthodes : les méthodes directes et les m. itératives.

2. Méthodes directes.

*Déf : nous appelons méthode directe de résolution du système, une méthode qui permet de calculer $X = A^{-1}b$ en un nombre fini d'opérations élémentaires.

Si la matrice A du système est triangulaire supérieure

$$(2)\begin{cases} a_{11}x_1 + \dots + a_{1n}x_n = b_1 \\ \qquad \ddots \qquad\qquad \vdots \\ \qquad\quad a_{n-1n-1}x_{n-1} + a_{n-1}x_n = b_{n-1} \\ \qquad\qquad\qquad a_n x_n = b_n \end{cases}$$

sens de résolution

On peut écrire l'algorithme de résolution :

$$(2.2)\quad \begin{cases} x_n = \dfrac{b_n}{a_{nn}} \\ x_i = \dfrac{b_i - \sum_{j=i+1}^{n} a_{ij} x_j}{a_{ii}} \qquad (i = n-1, n-2, \dots, 1) \end{cases}$$

Le nombre d'opérations $\begin{cases} n \text{ divisions} \\ \frac{n(n-1)}{2} \text{ multiplications} \\ \frac{n(n-1)}{2} \text{ additions} \end{cases} \Rightarrow \Sigma op = n^2$

Si A n'est pas triangulaire, nous sommes amenés à chercher une matrice M inversible tq MA soit triangulaire. On résoud alors $MAx = Mb$ (par l'algo 2.2)

1. Méthode de GAUSS

La méthode de Gauss transforme le système $Ax=b$ en un système triangulaire équivalent à l'aide d'un algorithme d'élimination.

$$A^{(1)}x = b^{(1)} \quad \text{avec } A^{(1)} = (a_{ij}^{(1)}) = A, \quad b^{(1)} = (b_i^{(1)}) = b$$

Eventuellement, après permutation de lignes ou de colonnes dans $A^{(1)}$, on peut supposer que $a_{11}^{(1)} \neq 0$ (le 1e pivot).

Pour $i = 2\_n$, multiplions la 1e équation de notre système par $g_{i1} = \frac{a_{i1}}{a_{11}^{(1)}}$ et retranchons l'équation obtenue de la $i^{\text{ème}}$

Nous obtenons alors un système de la forme $A^{(2)}x = b^{(2)}$, $A^{(2)} = (a_{ij}^{(2)})$, $b^{(2)} = (b_i^{(2)})$

$$\begin{cases} a_{1j}^{(2)} = a_{1j}^{(1)} & j = 1\_n \\ a_{i1}^{(2)} = 0 & i = 2\_n \\ a_{ij}^{(2)} = a_{ij}^{(1)} - g_{i1} a_{1j}^{(1)} & \begin{cases} i = 2\_n \\ j = 2\_n \end{cases} \\ b_1^{(2)} = b_1^{(1)} \\ b_i^{(2)} = b_i^{(1)} - g_{i1} b_1^{(2)} & i = 2\_n \end{cases} \qquad A^{(2)} = \begin{pmatrix} a_{11}^{(1)} & \cdots & & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & \vdots & \times \;\times\;\times & \vdots \\ 0 & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} \end{pmatrix}$$

$$\tilde{A} = (A|b) = \begin{pmatrix} a_{11} & \cdots & a_{1n} & b_1 \\ \vdots & & \vdots & \vdots \\ a_{n1} & \cdots & a_{nn} & b_n \end{pmatrix} \quad b_i = a_{i\,n+1} \qquad \tilde{A}^{(2)} = (A^{(2)} | b^{(2)})$$

Plus généralement, la méthode de GAUSS consiste à construire une suite de systèmes équivalents à $A^{(k)}x = b^{(k)}$ où $A^{(k)}$ et $b^{(k)}$ sont donnés par $\tilde{A}^{(k)}$

$$\tilde{A}^{(k)} = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & & \cdots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & & \cdots & a_{2n}^{(2)} & b_2^{(2)} \\ \vdots & & \ddots & & & \vdots & \vdots \\ \vdots & 0 & 0 & a_{k-1k-1}^{(k-1)} & \cdots & a_{kn}^{(k)} & b_{k-1}^{(k-1)} \\ 0 & \cdots & 0 & a_{nk}^{(k)} & \cdots & a_{nn}^{(k)} & b_n^{(k)} \end{pmatrix}$$

On suppose que le pivot $a_{kk}^{(k)} \neq 0$. Pour $i = k+1\_n$, on multiplie la $k^{ième}$ ligne de $\tilde{A}^{(k)}$ par $g_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$ puis on retranche le résultat de la $i^{ème}$ ligne. On obtient alors $\tilde{A}^{(k+1)}$

$$\begin{cases} a_{ij}^{(k+1)} = a_{ij}^{(k)} & i = 1\_k \\ a_{ij}^{(k+1)} = a_{ij}^{(k)} - g_{ik} a_{kj}^{(k)} & \begin{cases} i = k+1\_n \\ j = k+1\_n+1 \end{cases} \\ \text{Les autres coefficients sont nuls} \end{cases}$$

à la $n^{ième}$ étape, la matrice $A^{(n)}$ est triangulaire supérieure, le système $A^{(n)}x = b^{(n)}$ est résolu par l'algo (2.2).

Remarque: Le nombre d'opérations élémentaires pour l'algorithme de Gauss est :

$$\sum op = T_G = \frac{4n^3 + 9n^2 - 7n}{6}$$

$n = 5 \quad T_G = 115$

$n = 10 \quad T_G = 805$

Analyse matricielle de Gauss

Soit $G^{(k)} = \begin{pmatrix} 1 & & & & 0 \\ & \ddots & & & \\ & & 1 & & \\ & & g_{k+1k} & 1 & \\ 0 & & \vdots & 0 & \ddots \\ & & g_{nk} & & & 1 \end{pmatrix}$ $\quad g_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$

$k^{ième}$ colonne

$\tilde{A}^{(k+1)} = G^{(k)} . \tilde{A}^{(k)} \qquad \tilde{A}^{(n)} = G^{(n-1)} . \tilde{A}^{(n-1)} = G^{(n-1)} G^{(n-2)} \tilde{A}^{(n-2)}$

$\tilde{A}^{(n)} = G^{(n-1)} . G^{(n-2)} \dots G^{(1)} \tilde{A}^{(1)} \qquad U = A^{(n)}$

$U = A^{(n)}, \; L = \left[G^{(n-1)} . G^{(n-2)} \dots G^{(1)} \right]^{-1}$

$A = LU$ Factorisation de Gauss.

$$L = \begin{pmatrix} 1 & & 0 \\ & \ddots & \\ g_{ik} & & 1 \end{pmatrix}, \quad U = A^{(n)} = \begin{pmatrix} a_{11}^{(1)} & \dots & a_{1n}^{(1)} \\ & a_{22}^{(2)} \dots & a_{2n}^{(2)} \\ 0 & \ddots & \vdots \\ & & a_{nn}^{(n)} \end{pmatrix}$$

Remarque: $A = L.U \qquad \det A = \underbrace{\det L}_{=1} . \det U = \det U \qquad \det A = \prod_{k=1}^{n} a_{kk}^{(k)}$

2. Méthode de Cholesky.

Soit A symétrique déf. >0 : $(Ax, x) = {}^t x A x > 0 \quad \forall x \neq \vec{0} \qquad A$ est S.D.P.

Théorème de Cholesky: A est S.D.P. $\Leftrightarrow A = L\,{}^tL$ où L est une matrice inversible triangulaire inférieure

Algorithme de Cholesky.

$Ax = b \Leftrightarrow L \underbrace{{}^tL}_{y} = b \quad \Leftrightarrow \quad \begin{cases} Ly = b \downarrow \\ {}^tLx = y \uparrow \end{cases}$

$L = (l_{ij}) \qquad A = L\,{}^tL \qquad a_{ij} = \sum_{k=1}^{j} l_{ik} l_{jk} \quad j \leq i \qquad L = \begin{pmatrix} l_{11} & & 0 \\ & \ddots & \\ l_{ij} & & l_{nn} \end{pmatrix}$

$$\begin{cases} a_{11} = l_{11}^2 \\ a_{i1} = l_{i1} l_{11} \end{cases} \Rightarrow \begin{cases} l_{11} = \sqrt{a_{11}} \\ l_{i1} = \frac{a_{i1}}{l_{11}} \quad i = 2\_n \end{cases}$$

ce qui permet de déterminer la $1^{ère}$ colonne de L.

Nous allons montrer qu'on peut construire L colonne par colonne.

Supposons connues les $(k-1)$ premières colonnes de L.

$$a_{kk} = \sum_{j=1}^{k} l_{kj}^2 = l_{kk}^2 + \sum_{j=1}^{k-1} l_{kj}^2 \Rightarrow \underline{l_{kk} = \sqrt{a_{kk} - \sum_{j=1}^{k-1} l_{kj}^2}}$$

$$a_{ik} = \sum_{j=1}^{k} l_{ij} l_{kj} = l_{ik} l_{kk} + \sum_{j=1}^{k-1} l_{ij} l_{kj} \Rightarrow \boxed{l_{ik} = \frac{a_{ik} - \sum_{j=1}^{k-1} l_{ij} l_{kj}}{l_{kk}}, \quad i = k+1 \_ n}$$