V. Résumé de chapitre V : Résolution numérique des systèmes d'équations linéaires

V.1. Introduction

On appelle un système linéaire de n équations et n inconnues, un système de la forme :

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2 \\ a_{31}x_1 + a_{32}x_2 + \ldots + a_{3n}x_n = b_3 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n = b_n \end{cases}$$

Sa forme matricielle: $A\vec{x} = \vec{b}$ s'écrit :

$$\begin{pmatrix} a_{11} \; a_{12} \; a_{13} \ldots a_{1n} \\ a_{21} \; a_{22} \; a_{23} \ldots a_{2n} \\ a_{31} \; a_{32} \; a_{33} \ldots a_{3n} \\ \vdots \;\; \vdots \;\; \vdots \;\; \vdots \;\; \vdots \\ a_{n1} \; a_{n2} \; a_{n3} \ldots a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{pmatrix}$$

V.2. Transposé (A')et inverse (A^{-1}) d'une matrice A

$$A' = \begin{pmatrix} a_{11} \; a_{21} \; a_{31} \ldots a_{n1} \\ a_{12} \; a_{22} \; a_{32} \ldots a_{n2} \\ a_{13} \; a_{23} \; a_{33} \ldots a_{n3} \\ \vdots \;\; \vdots \;\; \vdots \;\; \vdots \;\; \vdots \\ a_{1n} \; a_{2n} \; a_{3n} \ldots a_{nn} \end{pmatrix} \; ; \; A^{-1} = \begin{pmatrix} \Delta_{11} \; \Delta_{21} \; \Delta_{31} \ldots \Delta_{n1} \\ \Delta_{12} \; \Delta_{22} \; \Delta_{32} \ldots \Delta_{n2} \\ \Delta_{13} \; \Delta_{23} \; \Delta_{33} \ldots \Delta_{n3} \\ \vdots \;\; \vdots \;\; \vdots \;\; \vdots \;\; \vdots \\ \Delta_{1n} \; \Delta_{2n} \; \Delta_{3n} \ldots \Delta_{nn} \end{pmatrix}$$

Où $\triangle_{ij}$: sont les cofacteurs de la matrice A.

Une matrice est inversible (non singulière) si $\det(A) \neq 0$, attention : $(AB)^{-1} = B^{-1}A^{-1}$

V.3. Matrice triangulaire supérieur *U* (*Upper*), Inférieur *L*(*Lower*) et Diagonale (*D*)

$$L=\begin{pmatrix} l_{11} & 0 & 0 & \dots & 0 \\ l_{21} & l_{22} & 0 & \dots & 0 \\ l_{31} & l_{32} & l_{33} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ l_{n1} & l_{n2} & l_{n3} & \dots & l_{nn} \end{pmatrix} ;\ U=\begin{pmatrix} u_{11} & u_{12} & u_{13} & \dots & u_{1n} \\ 0 & u_{22} & u_{23} & \dots & u_{2n} \\ 0 & 0 & u_{33} & \dots & u_{3n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & u_{nn} \end{pmatrix} \text{ et } D=\begin{pmatrix} d_{11} & 0 & 0 & \dots & 0 \\ 0 & d_{22} & 0 & \dots & 0 \\ 0 & 0 & d_{33} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & d_{nn} \end{pmatrix}$$

Le déterminant des trois matrices précédentes est calculé comme suit :

$$\det(L)=\prod_{i=1}^{n} l_{ii} \quad \det(U)=\prod_{i=1}^{n} u_{ii} \text{ et } \det(D)=\prod_{i=1}^{n} d_{ii}$$

V.4. Critère de Silvestre

Matrice symétrique : la matrice A (caractérisée par $(a_{ij})_{1\le i,j\le n}$) est dite symétrique si $A=A'$ (notée également : $a_{ij}=a_{ji}$).

Matrice définie positive : il suffit que $det(A_i)>0,(i=1,2,3,\dots,n)$ où A_i : sont les sous-matrices principales, telle que : $A_i=\begin{pmatrix} a_{11} & \dots & a_{11} \\ \vdots & \vdots & \vdots \\ a_{11} & \dots & a_{11} \end{pmatrix}$

V.5. Méthodes directes

V.5.1. Méthode d'élimination de Gauss

Cette méthode de Gauss consiste à transformer le système $A\vec{x}=\vec{b}$ en $U\vec{x}=\overrightarrow{b'}$, ce dernier est résolu par remontée triangulaire.

V.5.2. Méthode de décomposition (factorisation)LU

Cette méthode transforme le système $A\vec{x}=\vec{b}$ en $LU\vec{x}=\vec{b}$ qui s'écrit également :

$$\begin{cases} L\vec{y}=\vec{b} \\ U\vec{x}=\vec{y} \end{cases}$$

On résout donc le système $L\vec{y}=\vec{b}$ par descente triangulaire puis $U\vec{x}=\vec{y}$ par remontée triangulaire

V.5.3. Remarque

L'existence et l'unicité de la factorisation LU est assurée par l'une des deux conditions suivantes :

- Si et seulement si les sous-matrices A_i sont inversibles ($det(A_i) \neq 0$).
- Si la matrice A est dite à diagonale strictement dominante.

V.5.4. Les différentes formes de LU

V.5.4.1. Factorisation de Dolittle :

U quelconque et *L* prend la forme: $L = \begin{pmatrix} 1 & 0 & 0 \ldots 0 \\ l_{21} & 1 & 0 \ldots 0 \\ l_{31} & l_{32} & 1 \ldots 0 \\ \vdots & \vdots & \vdots \; \vdots \; \vdots \\ l_{n1} & l_{n2} & l_{n3} \ldots 1 \end{pmatrix}$, elle est basée sur le principe d'élimination de Gauss.

V.5.4.2. Factorisation de Crout :

L quelconque et *U* prend la forme : $U = \begin{pmatrix} 1 & u_{12} & u_{13} \ldots u_{1n} \\ 0 & 1 & u_{23} \ldots u_{2n} \\ 0 & 0 & 1 \ldots u_{3n} \\ \vdots & \vdots & \vdots \; \vdots \; \vdots \\ 0 & 0 & 0 \ldots 1 \end{pmatrix}$

V.5.4.3. Factorisation de Cholesky :

U est la transposée de *L*, donc $LU = LL^t$

$$L = \begin{pmatrix} l_{11} & 0 & 0 \ldots 0 \\ l_{21} & l_{22} & 0 \ldots 0 \\ l_{31} & l_{32} & l_{33} \ldots 0 \\ \vdots & \vdots & \vdots \; \vdots \; \vdots \\ l_{n1} & l_{n2} & l_{n3} \ldots l_{nn} \end{pmatrix} \text{ et } U = L^t = \begin{pmatrix} l_{11} & l_{21} & l_{31} \ldots l_{n1} \\ 0 & l_{22} & l_{32} \ldots l_{n2} \\ 0 & 0 & l_{33} \ldots l_{n3} \\ \vdots & \vdots & \vdots \; \vdots \; \vdots \\ 0 & 0 & 0 \ldots l_{nn} \end{pmatrix}$$

La décomposition de *Cholesky* $\left(A = LL^t\right)$ existe si A est symétrique et définie positive

V.6. Méthodes itératives

A partir de la matrice *A*, on donne les matrices suivantes :

$$E=\begin{pmatrix} 0 & 0 & 0 & \dots 0 \\ -a_{21} & 0 & 0 & \dots 0 \\ -a_{31} & -a_{32} & 0 & \dots 0 \\ \vdots & \vdots & \vdots & \vdots \; \vdots \\ -a_{n1} & -a_{n2} & -a_{n3} & \dots 0 \end{pmatrix} \; ; \; F=\begin{pmatrix} 0 & -a_{12} & -a_{13} & \dots & -a_{1n} \\ 0 & 0 & -a_{23} & \dots & -a_{2n} \\ 0 & 0 & 0 & \dots & -a_{3n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix} \text{ et } D=\begin{pmatrix} a_{11} & 0 & 0 & \dots & 0 \\ 0 & a_{22} & 0 & \dots & 0 \\ 0 & 0 & a_{33} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & a_{nn} \end{pmatrix}$$

V.6.1. Méthode de Jacobi

Elle transforme $A\vec{x}=\vec{b}$ en un système itératif : $\vec{x}_{k+1}=D^{-1}\left(E+F\right)\vec{x}_k+D^{-1}\vec{b}$.
où $M_J=D^{-1}\left(E+F\right)$ est appelée la matrice de *Jacobi*.
On peut également écrire ce système itératif sous forme :

$$x_1^{(k)}=\frac{1}{a_{11}}\left(b_1-a_{12}x_2^{(k-1)}-a_{13}x_3^{(k-1)}-\ldots-a_{1n}x_n^{(k-1)}\right)$$

$$x_2^{(k)}=\frac{1}{a_{22}}\left(b_2-a_{21}x_1^{(k-1)}-a_{23}x_3^{(k-1)}-\ldots-a_{2n}x_n^{(k-1)}\right)$$

$$x_3^{(k)}=\frac{1}{a_{33}}\left(b_3-a_{31}x_1^{(k-1)}-a_{32}x_2^{(k-1)}-\ldots-a_{3n}x_n^{(k-1)}\right)$$

……………………………………………………......

$$x_n^{(k)}=\frac{1}{a_{nn}}\left(b_n-a_{n1}x_1^{(k-1)}-a_{n2}x_2^{(k-1)}-\ldots-a_{1n-1}x_{n-1}^{(k-1)}\right)$$

Cette méthode converge, lorsque $\det\left(D\right)\neq 0$, si et seulement si le rayon spectral: $\rho\left(D^{-1}\left(E+F\right)\right)<1$.

V.6.2. Méthode de Gauss-Seidel

La méthode de *Gauss-Seidel* considère le système itératif suivant : $\vec{x}_{k+1}=\left(D-E\right)^{-1}F\,\vec{x}_k+\left(D-E\right)^{-1}\vec{b}$, où $M_{GS}=\left(\left(D-E\right)^{-1}F\right)$, ce dernier système vient de :

$$x_1^{(k)}=\frac{1}{a_{11}}\left(b_1-a_{12}x_2^{(k-1)}-a_{13}x_3^{(k-1)}-\ldots-a_{1n}x_n^{(k-1)}\right)$$

$$x_2^{(k)}=\frac{1}{a_{22}}\left(b_2-a_{21}x_1^{(k)}-a_{23}x_3^{(k-1)}-\ldots-a_{2n}x_n^{(k-1)}\right)$$

$$x_3^{(k)}=\frac{1}{a_{33}}\left(b_3-a_{31}x_1^{(k)}-a_{32}x_2^{(k)}-\ldots-a_{3n}x_n^{(k-1)}\right)$$

…………………………………………………....

$$x_n^{(k)}=\frac{1}{a_{nn}}\left(b_n-a_{n1}x_1^{(k)}-a_{n2}x_2^{(k)}-\ldots-a_{1n-1}x_{n-1}^{(k)}\right)$$

Après vérification de $\det\left(D-E\right)\neq 0$, cette méthode converge si et seulement si $\rho\left(\left(D-E\right)^{-1}F\right)<1$.

V.6.3. *Remarques*

- Si le calcul du rayon spectral des matrices de Jacobi M_J ou de Gauss-Seidel M_{GS} est relativement long, on vérifie si A est à diagonale strictement dominante ou bien elle est symétrique et définie positive.
- La valeur du rayon spectral indique la rapidité de convergence des deux méthodes

V.7. Algorithme de Jacobi et de Gauss-Seidel

Méthode de Jacobi	Méthode de Gauss-Seidel

Données :
$A = (a_{ij})_{1\le i,j\le n}$, $b = (b_i)_{1\le i\le n}$, ItérMax, Toll ;
Initialisation :
Itér ← 0 ;
r ← |b − Ax| ;
While (r > $Toll$ et Itér < ItérMax)
    Itér ← Itér + 1 ;
    y ← x ;
    for i variant de 1 à n
        s ← 0 ;
        for j variant de 1 à i-1
            s←s+a_{ij} y_j ;
        end for
        for j variant de i+1 à n
        s←s+a_{ij} y_j ;
        end for
    x_i← (b_i−s)/a_{ii} ;
    end for
    r← ||b − Ax|| ;
end while
	Données :
	$A = (a_{ij})_{1\le i,j\le n}$, $b = (b_i)_{1\le i\le n}$, ItérMax, Toll ;
	Initialisation :
	Itér ← 0 ;
	r ← |b − Ax| ;
	While (r > $Toll$ et Itér < ItérMax)
	    Itér ← Itér + 1 ;
	    y ← x ;
	    for i variant de 1 à n
	        s ← 0 ;
	        for j variant de 1 à i-1
	            s←s+a_{ij} y_j ;
	        end for
	        for j variant de i+1 à n
	        s←s+a_{ij} x_j ;
	        end for
	    x_i← (b_i−s)/a_{ii} ;
	    end for
	    r← ||b − Ax|| ;
	end while