CI 3 : Ingénierie Numérique & Simulation

Chapitre 4 – Résolution d'un système linéaire inversible par la méthode de Gauss

Savoir

Problème discret multidimensionnel, linéaire, conduisant à la résolution d'un système linéaire inversible (ou de Cramer) par la méthode de Gauss avec recherche partielle du pivot.

Le pivot de Gauss est une méthode permettant de résoudre les systèmes linéaires. Sur l'idée du pivot de Gauss, il est alors possible de calculer l'inverse d'une matrice (lorsqu'elle est inversible), de calculer le déterminant d'une matrice $A \in \mathcal{M}_n(\mathbb{R})$ ou de calculer le rang de $A \in \mathcal{M}_{n,p}(\mathbb{R})$.

Problématique

Outre la compréhension et la mise en œuvre de l'algorithme, deux problèmes numérique peuvent se poser :
- les erreurs d'arrondis peuvent provoquer des erreurs importantes selon la méthode choisie ;
- des comparaisons d'un réel à zéro peuvent aussi engendrer des erreurs numériques.

1 Système linéaire

Remarque

Les résultats du cours sont écrits avec des coefficients réels mais restent vrais avec des coefficients complexes.

1.1 Définitions

Définition

On dit d'un système qu'il est linéaire s'il est de la forme :

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + ... + a_{1p}x_p = b_1 \\ ... \\ a_{n1}x_1 + a_{n2}x_2 + ... + a_{np}x_p = b_n \end{cases}$$

On note $n, p \in \mathbb{N}^*$ (n équations et p inconnues), $\forall i \in \{1, ..., n\}$, $\forall j \in \{1, ..., p\}$, $a_{ij} \in \mathbb{R}$. i désigne l'indice de la ligne et j l'indice de la colonne.

$b_1, ..., b_n \in \mathbb{R}$ est appelé second membre.

Définition

On dit que le système est homogène su $b_1 = b_2 = ... = b_n = 0$.

Exemple

Exemple de système linéaire (S) et son système homogène associé (S_0) :

$$(S)\begin{cases} x + y - 2z + 4t = 5 \\ 2x + 2y - 3z + t = 3 \\ 3x + 3y - 4z - 2t = 1 \end{cases} \qquad (S_0)\begin{cases} x + y - 2z + 4t = 0 \\ 2x + 2y - 3z + t = 0 \\ 3x + 3y - 4z - 2t = 0 \end{cases}$$

Exemple de système non linéaire :

$$\begin{cases} x + \cos \mathbf{y} + \mathbf{xy} = 2 \\ x - \mathbf{y^2} = 4 \end{cases}$$

Définition

Une solution d'un système linéaire est un p-uplet de réels, c'est-à-dire un élément de $(x_1, ..., x_p)$ qui vérifie les n équations.

Définition

Si un système a au moins une solution, il est dit compatible (incompatible sinon).

1.2 Opérations élémentaires

Définition

Les opérations élémentaires sont les suivantes :
- opération de transvection (*Gaussian Elimination*) :
  - $(L_i) \leftarrow (L_i) + \lambda(L_j)$ où $i, j \in \{1, ..., n\}$, $i \neq j$, $\lambda \in \mathbb{R}$;
  - $(L_i) \leftarrow \alpha(L_i)$ où $i \in \{1, ..., n\}$, $\alpha \in \mathbb{R}^*$;
- opération d'échange de lignes :
  - $(L_i) \leftrightarrow (L_j)$ où $i, j \in \{1, ..., n\}$.

Remarque

Pour éviter les fractions on peut également utiliser $(L_i) \leftarrow \alpha(L_i) + \lambda L_j$ où $i, j \in \{1, ..., n\}$, $\alpha \neq 0$, $\lambda \in \mathbb{R}$ qui est une combinaison des deux premiers items.

Propriété

L'utilisation des opérations élémentaires sur un système ne change pas l'ensemble de ses solutions. Autrement dit, elles donnent des systèmes équivalents au premier.

1.3 Notation matricielle

Remarque

Les opérations élémentaires sur les lignes du système n'opèrent que sur les coefficients.

Définition

Au système défini précédemment, on associe la matrice de ses coefficients :

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1(p-1)} & a_{1p} \\ & & a_{ij} & & \\ a_{n1} & a_{n2} & \dots & a_{n(p-1)} & a_{np} \end{pmatrix} \in \mathcal{M}_{n,p}(\mathbb{R})$$

$\mathcal{M}_{n,p}(\mathbb{R})$ désigne l'ensemble des matrices à coefficients réels à n lignes et p colonnes.

Définition

On note $B = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}$ la matrice de second membre et $(A|B)$ la matrice A augmentée de B.

Définition

Si deux matrices M et M' diffèrent d'un nombre fini d'opération sur les lignes, on dit qu'elles sont équivalentes en lignes et on note $M \underset{L}{\sim} M'$.

Remarque

Sous forme matricielle le système s'écrit $AX = B$ avec X le vecteur inconnu.

2 Pivot de Gauss

2.1 Pivot d'une ligne

Définition

On appelle pivot d'une ligne le premier nombre non nul de cette ligne.

2.2 Algorithme du pivot

Premier cas, chacun des coefficients de la première colonne est nul. En conséquence, on note $M = \begin{pmatrix} 0 & | & \\ \vdots & | & M \\ 0 & | & \end{pmatrix}$

Dans un second cas, la première colonne contient au moins un nombre non nul.

Quitte à effectuer un changement de ligne, on se ramène au cas où $a_{11} \neq 0$:

$$\left(\begin{array}{c|ccc} a_{11} & \alpha & \alpha & \alpha \\ \hline \alpha & & & \\ \alpha & & \alpha & \\ \alpha & & & \end{array}\right)$$

Les α représentent des nombres possiblement nuls.

On fait apparaître des zéros sous a_{11} avec des opérations de la forme $(L_2) \leftarrow (L_2) - \dfrac{a_{21}}{a_{11}}(L_1)$

$$\left(\begin{array}{c|ccc} a_{11} & \alpha & \alpha & \alpha \\ \hline 0 & & & \\ \vdots & & M' & \\ 0 & & & \end{array}\right)$$

On effectue le pivot à nouveau sur la matrice M'.

En généralisant, lorsqu'on en est à la ligne i, les lignes $i+1$ à n subissent la transvection suivante :

$$\forall k \in [i+1, n] \quad L_k \leftarrow L_k - \frac{a_{k,i}}{a_{i,i}} L_i$$

Remarque

Le nombre de colonnes diminue à chaque pivot.

L'algorithme se termine en un nombre fini d'étapes.

Exemple

Soit le système suivant à résoudre ainsi que la matrice augmentée qui lui est associée :

$$\begin{cases} x+y-2z+4t=5 \\ 2x+2y-3z+2t=3 \\ 3x+3y-4z-2t=1 \\ 1x+2y+3z+3t=-1 \end{cases} \qquad \left(\begin{array}{cccc|c} 1 & 1 & -2 & 4 & 5 \\ 2 & 2 & -3 & 2 & 3 \\ 3 & 3 & -4 & -2 & 1 \\ 1 & 2 & 3 & 3 & -1 \end{array}\right)$$

$$\begin{cases} (L_2) \leftarrow (L_2) - 2(L_1) \\ (L_3) \leftarrow (L_3) - 3(L_1) \\ (L_4) \leftarrow (L_4) - (L_1) \end{cases} \implies \left(\begin{array}{cccc|c} 1 & 1 & -2 & 4 & 5 \\ 0 & 0 & 1 & -7 & -6 \\ 0 & 0 & 2 & -14 & -14 \\ 0 & 1 & 5 & -1 & -6 \end{array}\right)$$

$$(L_2) \leftrightarrow (L_4) \implies \left(\begin{array}{cccc|c} 1 & 1 & -2 & 4 & 5 \\ 0 & 1 & 5 & -1 & -6 \\ 0 & 0 & 2 & -14 & -14 \\ 0 & 0 & 1 & -6 & -7 \end{array}\right)$$

Exemple

$$(L_4) \leftarrow (L_4) - \frac{1}{2}(L_3) \quad \Longrightarrow \quad \left(\begin{array}{cccc|c} 1 & 1 & -2 & 4 & 5 \\ 0 & 1 & -1 & -7 & -6 \\ 0 & 0 & 2 & -14 & -14 \\ 0 & 0 & 0 & 1 & 0 \end{array}\right)$$

Remarque

Numériquement, le codage du 0 posant des problèmes informatiques, il est difficile de s'assurer qu'un pivot est non nul. On utilisera donc la méthode du **pivot partiel**. Le pivot utilisé ne sera donc pas le premier nombre non nul d'une ligne, mais le plus grand élément d'une colonne (en valeur absolue). Si le pivot n'est pas sur la première ligne, on effectuera les échanges de lignes nécessaires. On s'affranchit ainsi de comparaisons à 0.

2.3 Matrice échelonnée

A la fin du pivot, on obtient une matrice de la **forme** :

$$\left(\begin{array}{cccc|c} \alpha & \beta & \beta & \beta & \beta \\ 0 & \alpha & \beta & \beta & \beta \\ 0 & 0 & \alpha & \beta & \beta \\ 0 & 0 & 0 & \alpha & \beta \end{array}\right) \quad \text{avec } \alpha \text{ non nul et } \beta \text{ réel quelconque.}$$

Définition

On dit d'une matrice qu'elle est échelonnée en ligne si :

1. une ligne est nulle, les suivantes le sont aussi ;
2. $L_1, ..., L_r$ désignent les lignes non nulles, $j(L_1), ..., j(L_r)$ désignent la position des pivots dans ces lignes et $j(L_1) < ... < j(L_r)$.

Exemple

Les matrices suivantes ne sont pas échelonnées :

$$\begin{pmatrix} 1 & 0 & 2 & 3 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 & 2 & 3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 & 2 & 3 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 1 & 5 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Propriété

Le pivot de Gauss permet d'obtenir une matrice échelonnée en lignes.

Définition

Le rang d'une matrice échelonnée désigne le nombre de ses pivots (ce qui correspond aussi au nombre de lignes non nulles).

2.4 Matrices échelonnées réduites

On peut poursuivre le pivot en :

1. divisant les lignes non nulles par leur pivot, les pivot deviennent alors 1 ;

2. faisant apparaître des zéros au dessus des pivots.

Propriété

Toute matrice est équivalente en ligne à une et une seule matrice échelonnée réduite.

En conséquence, on peut définir le rang d'une matrice quelconque comme le rang de sa matrice échelonnée réduite en ligne.

On peut définir (le nombre) le rang d'un système comme le rang de la matrice de ses coefficients (A).

Remarque

Deux matrices équivalentes en ligne ont le même rang.

3 Implémentation de l'algorithme

3.1 Choix d'un type de données

On verra ultérieurement que la bibliothèque numpy permet de gérer plus aisément le calcul matriciel. Dans un premier temps, le système à résoudre sera mis sous forme matricielle afin que les valeurs puissent être stockées dans un tableau. On cherchera donc à résoudre le système suivant :

$$A \cdot X = B$$

La matrice $A \in \mathscr{M}_{n,n}(\mathbb{R})$ sera stockée dans un tableau de n lignes et n colonnes. Le vecteur B, second membre de l'équation, sera stocké dans un tableau de n lignes et une colonne.

3.2 Fonctions élémentaires

3.2.1 Recherche du pivot

Dans le cadre d'une résolution numérique, on utilise un pivot partiel.

python

```
def recherche_pivot(A,i):
    n = len(A) # le nombre de lignes
    j = i # la ligne du maximum provisoire
    for k in range(i+1, n):
        if abs(A[k][i]) > abs(A[j][i]):
            j = k # un nouveau maximum provisoire
    return j
```

On considère qu'on est à la ligne i. À la ligne i, les éléments compris entre les colonnes 0 et $i-1$ sont théoriquement nuls. Initialement, le plus grand élément de la ligne est a priori en position i.

On cherche alors le plus grand élément, en valeur absolu de la ligne $i+1$à la ligne n (n exclus en Python).

On renvoie enfin l'indice du pivot.

3.2.2 Échange de lignes

L'échange de lignes peut être nécessaire pour réordonner les lignes de la matrice si le pivot de la ligne considéré est plus petit que le pivot d'une des lignes suivantes.

```python
def echange_lignes(A,i,j):
    # Li <-->Lj
    A[i][:], A[j][:]= A[j][:], A[i][:]
```

Remarque

En python, le passage des objets se faisant par référence, il n'est pas nécessaire de retourner une liste.

3.2.3 Transvection de lignes

```python
def transvection_ligne(A, i, j, mu):
    # L_i <- L_i + mu.L_j
    nc = len(A[0]) # le nombre de colonnes
    for k in range(nc):
        A[i][k] = A[i][k] + mu * A[j][k]
```

3.2.4 Phase de remontée : résolution du système

Une fois toutes les transvections réalisées, on est en présence d'une matrice échelonnée T. Le système a donc été mis sous la forme $T \cdot X = Y$. On va donc résoudre le système en partant «du bas».

```python
def remontee(A,B):
    n=len(A)
    X = [0.] * n
    for i in range(n-1, -1, -1):
        somme=0
        for j in range (i+1,n):
            somme=somme+A[i][j]*X[j]
        X[i]=(B[i][0]-somme)/A[i][i]
        print(X[i])
    return X
```

3.3 Algorithme complet

```python
def resolution (AA, BB):
    """Résolution de AA.X=BB; AA doit etre inversible"""
    A, B = AA.copy(), BB.copy()
    n = len(A)
```

```python
    assert len(A[0]) == n
    # Mise sous forme triangulaire
    for i in range(n):
        j = recherche_pivot(A, i)
        if j > i:
            echange_lignes(A, i, j)
            echange_lignes(B, i, j)
        for k in range(i+1, n):
            mu = - A[k][i] / A[i][i]
            transvection_ligne(A, k, i, mu)
            transvection_ligne(B, k, i, mu)
    # Phase de remontée
    return remontee(A,B)
```

3.4 Résolution de systèmes linéaires avec Numpy

Il est possible d'utiliser le bibliothèque Numpy pour résoudre le système $A \cdot X = B$. Pour cela, les matrice A et B peuvent être mises sous la forme d'une liste de liste.

```python
A=[[1,2,3],[4,5,6],[7,8,9]]
B=[[1],[2],[3]]
numpy.linalg.solve
```

Références

[1] Adrien Petri, *Pivot de Gauss – Systèmes linéaires*, Notes de cours de TSI 1, Lycée Rouvière, Toulon.

[2] Wack et Al., *Informatique pour tous en classes préparatoires aux grandes écoles*, Éditions Eyrolles.