Programmation et Algorithmes numériques 2
Résolution de systèmes linéaires

D. Cornu

S4-2019

Objectif

Exemple de système d'équations linéaires à résoudre

$$2x + y + 3z = 1$$
$$2x + 6y + 8z = 3$$
$$6x + 8y + 18z = 5$$

Sytème de 3 équations à 3 inconnues.

Peut être résolu à la mains dans ce cas

Cas courant : n équations à n inconnues

$$\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\
a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2 \\
&\vdots \\
a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n &= b_n
\end{aligned}$$

Représentation matricielle

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2 \\ &\vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n &= b_n \end{aligned}$$

Le cas général peux s'érire sous la forme du produit d'une matrice A par un vecteur $\vec{x}$, donnant un vecteur $\vec{b}$
soit $A\vec{x} = \vec{b}$

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

Méthode d'élimination de Gauss : Introduction

L'objectif de cette méthode est de trouver un moyen d'exprimer un système sous une **forme triangulaire** dont la solution est triviale.

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ & a_{22} & a_{23} \\ & & a_{33} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

$$x_3 = b_3/a_{33} \qquad \begin{aligned} a_{22}x_2 &= b_2 - a_{23}x_3 \\ x_2 &= (b_2 - a_{23}x_3)/a_{22} \end{aligned} \qquad \begin{aligned} a_{11}x_1 &= b_1 - a_{12}x_2 - a_{13}x_3 \\ x_1 &= (b_1 - a_{12}x_2 a_{13}x_3)/a_{11} \end{aligned}$$

La solution générale s'écrit telle que :

$$x_i = \frac{b_i - \sum_{j=i+1}^{n} a_{ij}x_j}{a_{ii}}$$

Méthode d'élimination de Gauss : principes

Pour commencer il faut changer la représentation du système
On définit une **matrice augmentée** $\tilde{A}$ telle que :

$$\tilde{A} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} & b_n \end{pmatrix}$$

Principes mathématiques - La solution reste identique si :

- On permute 2 lignes dans la matrice
- On divise une ligne par une constante non nulle
- On ajoute/retire à une ligne une constante fois une autre ligne

A l'aide de ces opération on peux triangulariser le système

Exemple de l'agorithme du pivot de Gauss

Pour la ligne en cours on sélectionne le premier nombre non nul : le **pivot**.

$$\begin{pmatrix} \color{red}{2} & 1 & 3 & 1 \\ 2 & 6 & 8 & 3 \\ 6 & 8 & 18 & 5 \end{pmatrix}$$

Exemple de l'agorithme du pivot de Gauss

Pour la ligne en cours on sélectionne le premier nombre non nul : le **pivot**.

$$\begin{pmatrix} \color{red}{2} & 1 & 3 & 1 \\ 2 & 6 & 8 & 3 \\ 6 & 8 & 18 & 5 \end{pmatrix}$$

On divise la ligne par la valeur du pivot :

$$\begin{pmatrix} \color{red}{1} & 1/2 & 3/2 & 1/2 \\ 2 & 6 & 8 & 3 \\ 6 & 8 & 18 & 5 \end{pmatrix}$$

Exemple de l'agorithme du pivot de Gauss

Pour la ligne en cours on sélectionne le premier nombre non nul : le **pivot**.

$$\begin{pmatrix} \color{red}{2} & 1 & 3 & 1 \\ 2 & 6 & 8 & 3 \\ 6 & 8 & 18 & 5 \end{pmatrix}$$

On divise la ligne par la valeur du pivot :

$$\begin{pmatrix} \color{red}{1} & 1/2 & 3/2 & 1/2 \\ 2 & 6 & 8 & 3 \\ 6 & 8 & 18 & 5 \end{pmatrix}$$

On veut maintenant obtenir des 0 sous le pivot $\rightarrow$ soustraire la première ligne autant de fois que necessaire aux lignes suivantes.

$$L2 = L2 - 2 \times L1$$

$$L3 = L3 - 6 \times L1$$

Exemple de l'agorithme du pivot de Gauss

Pour la ligne en cours on sélectionne le premier nombre non nul : le **pivot**.

$$\begin{pmatrix} \color{red}{2} & 1 & 3 & 1 \\ 2 & 6 & 8 & 3 \\ 6 & 8 & 18 & 5 \end{pmatrix}$$

On divise la ligne par la valeur du pivot :

$$\begin{pmatrix} \color{red}{1} & 1/2 & 3/2 & 1/2 \\ 2 & 6 & 8 & 3 \\ 6 & 8 & 18 & 5 \end{pmatrix}$$

On veut maintenant obtenir des 0 sous le pivot $\rightarrow$ soustraire la première ligne autant de fois que necessaire aux lignes suivantes.

$$L2 = L2 - 2 \times L1$$
$$L3 = L3 - 6 \times L1$$

$$\begin{pmatrix} \color{red}{1} & 1/2 & 3/2 & 1/2 \\ 0 & 5 & 5 & 2 \\ 0 & 5 & 9 & 2 \end{pmatrix}$$

Exemple de l'agorithme du pivot de Gauss

Pour la ligne en cours on sélectionne le premier nombre non nul : le **pivot**.

$$\begin{pmatrix} 2 & 1 & 3 & 1 \\ 2 & 6 & 8 & 3 \\ 6 & 8 & 18 & 5 \end{pmatrix}$$

On divise la ligne par la valeur du pivot :

$$\begin{pmatrix} 1 & 1/2 & 3/2 & 1/2 \\ 2 & 6 & 8 & 3 \\ 6 & 8 & 18 & 5 \end{pmatrix}$$

On veut maintenant obtenir des 0 sous le pivot $\rightarrow$ soustraire la première ligne autant de fois que necessaire aux lignes suivantes.

$$L2 = L2 - 2 \times L1$$

$$L3 = L3 - 6 \times L1$$

$$\begin{pmatrix} 1 & 1/2 & 3/2 & 1/2 \\ 0 & 5 & 5 & 2 \\ 0 & 5 & 9 & 2 \end{pmatrix}$$

On recommence ensuite sur la ligne suivante

$$\begin{pmatrix} 1 & 1/2 & 3/2 & 1/2 \\ 0 & 1 & 1 & 2/5 \\ 0 & 5 & 9 & 2 \end{pmatrix}$$

$$L3 = L3 - 5 \times L2$$

$$\begin{pmatrix} 1 & 1/2 & 3/2 & 1/2 \\ 0 & 1 & 1 & 2/5 \\ 0 & 0 & 4 & 0 \end{pmatrix}$$

Exemple de l'agorithme du pivot de Gauss

Pour la ligne en cours on sélectionne le premier nombre non nul : le **pivot**.

$$\begin{pmatrix} \color{red}{2} & 1 & 3 & 1 \\ 2 & 6 & 8 & 3 \\ 6 & 8 & 18 & 5 \end{pmatrix}$$

On divise la ligne par la valeur du pivot :

$$\begin{pmatrix} \color{red}{1} & 1/2 & 3/2 & 1/2 \\ 2 & 6 & 8 & 3 \\ 6 & 8 & 18 & 5 \end{pmatrix}$$

On veut maintenant obtenir des 0 sous le pivot $\rightarrow$ soustraire la première ligne autant de fois que necessaire aux lignes suivantes.

$$L2 = L2 - 2 \times L1$$
$$L3 = L3 - 6 \times L1$$

$$\begin{pmatrix} \color{red}{1} & 1/2 & 3/2 & 1/2 \\ 0 & 5 & 5 & 2 \\ 0 & 5 & 9 & 2 \end{pmatrix}$$

On recommence ensuite sur la ligne suivante

$$\begin{pmatrix} 1 & 1/2 & 3/2 & 1/2 \\ 0 & \color{red}{1} & 1 & 2/5 \\ 0 & 5 & 9 & 2 \end{pmatrix}$$

$$L3 = L3 - 5 \times L2$$

$$\begin{pmatrix} 1 & 1/2 & 3/2 & 1/2 \\ 0 & \color{red}{1} & 1 & 2/5 \\ 0 & 0 & 4 & 0 \end{pmatrix}$$

La matrice est maintenant triangulaire, la résolution est triviale :

$$x_1 + x_2/2 + (3/2)x_3 = 1/2$$
$$x_2 + x_3 = 2/5$$
$$4x_3 = 0$$
$$\downarrow$$
$$\boldsymbol{x_1 = 3/10}$$
$$\boldsymbol{x_2 = 2/5}$$
$$\boldsymbol{x_3 = 0}$$

Pseudo Code

For $k = 1, \ldots, n$

pivot $= a_{kk}$

If pivot$/ = 0$

For $j = k, \ldots, n+1$

$a_{kj} = a_{kj}/$pivot

For $i = k+1, \ldots, n$

$fact = a_{ik}$ **For**

$j = k, \ldots, n+1$

$a_{ij} = a_{ij} -$ fact $* a_{kj}$

Else : "Problème"

k est l'indice de la colonnne en cours, celle dans laquelle on va faire apparaitre des 0.*

On divise la ligne k par le pivot.

On parcours les lignes $i > k$

On soustrait $a_{ik} \times (k^{ieme}$ligne) à la j^{ieme} ligne

**Le choix du pivot est ici trop simple. Usuellement on selectionne le plus grand pivot (en valeur absolue) de la colonne et on échange les lignes pour le placer sur la ligne en cours.*

Calcul de la solution

On peux alors récupérer la solution du système avec :

For $i = n, \ldots, 1$

$x_i = b_i - \sum_{j=i+1}^{n} a_{ij} x_j$

Complexité algorithmique :

- La triangularisation du système est en $O(n^3)$
- Résolution du système triangulaire en $O(n^2)$

Problème : Pivot nul

Lorsque le pivot selectionné est nul, il est impossible d'utiliser directement la méthode précédente

$$
\begin{aligned}
{\color{red}0}x + y + 3z &= 1\\
2x + 6y + 8z &= 3\\
6x + 8y + 18z &= 5
\end{aligned}
$$

Problème : Pivot nul

Lorsque le pivot selectionné est nul, il est impossible d'utiliser directement la méthode précédente

$$
\begin{aligned}
{\color{red}0}x + y + 3z &= 1\\
2x + 6y + 8z &= 3\\
6x + 8y + 18z &= 5
\end{aligned}
$$

Pour régler ce problème on peux **changer de pivot en permutant les équations**. On bien directement les lignes dans la matrice A.

$$
\begin{aligned}
{\color{red}6}x + 8y + 18z &= 5\\
0x + y + 3z &= 1\\
2x + 6y + 8z &= 3
\end{aligned}
$$

En pratique on cherchera **toujours** le pivot **le plus grand possible**, avant d'effectuer les opérations suivantes.

Problème : système mal conditionné

Dans le cas ou la solution varie beaucoup pour des systèmes très proche :

Système 1

$$4.218613x_1 + 6.327917x_2 = 10.546530$$
$$3.141592x_1 + 4.712390x_2 = 7.853982$$

Solution : $x_1 = x_2 = 1$

Système 2

$$4.218611x_1 + 6.327917x_2 = 10.546530$$
$$3.141594x_1 + 4.712390x_2 = 7.853980$$

Solution : $x_1 = -5 \quad et \quad x_2 = 5$

Problème : système mal conditionné

Dans le cas ou la solution varie beaucoup pour des systèmes très proche :

Système 1

$$4.218613x_1 + 6.327917x_2 = 10.546530$$
$$3.141592x_1 + 4.712390x_2 = 7.853982$$

Solution : $x_1 = x_2 = 1$

Système 2

$$4.218611x_1 + 6.327917x_2 = 10.546530$$
$$3.141594x_1 + 4.712390x_2 = 7.853980$$

Solution : $x_1 = -5 \quad et \quad x_2 = 5$

Ce cas de figure représente la recherche du point d'intersection de deux droite presques paralleles. Une légère modification des propriété des droite donne un grand changement dans la position de l'intersection.

Méthode de décomposition LU

Cette méthode est utilisée dans une très grande partie des bibliothèque de calcul scientifique haute performances. **Principe :** Décomposer une matrice

A en deux sous parties triangulaires L (Lower) et U (Upper) tel que :

$$A = L \times U$$

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{pmatrix} \times \begin{pmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{pmatrix}$$

Intéret de la décomposition LU

Une fois A décomposée on peux écrire :

$$A\vec{x} = \vec{b}$$
$$LU\vec{x} = \vec{b}$$

Ce qui en posant $U\vec{x} = \vec{y}$ nous donne :

$$L\vec{y} = \vec{b}$$
$$U\vec{x} = \vec{y}$$

Soit 2 systèmes linéaires triangulaires faciles à résoudre.

L'intéret majeur est que si $\vec{b}$ change mais que **A reste identique**, il est innutile de refaire la décomposition et on cherche juste la solution du premier système linéaire **qui est déja triangulaire**.
Cette méthode permet d'explorer des valeurs de $\vec{b}$ de manière très efficace ce qui est utilisé dans de nombreuses méthodes de résolution numériques complexes.

Exemple de la décomposition LU

Très similaire au pivot de Gauss, mais avec enregistrement des facteurs !

$$\begin{pmatrix} 2 & 1 & 3 \\ 2 & 6 & 8 \\ 6 & 8 & 18 \end{pmatrix}$$

On définit un pivot et on soustrait les lignes suivantes :

$$\begin{pmatrix} 2 & 1 & 3 \\ 1 & 5 & 5 \\ 3 & 5 & 9 \end{pmatrix}$$

A la place des 0 on garde les facteurs a_{ik} utilisés pour réduire la ligne.

Exemple de la décomposition LU

Très similaire au pivot de Gauss, mais avec enregistrement des facteurs !

$$\begin{pmatrix} 2 & 1 & 3 \\ 2 & 6 & 8 \\ 6 & 8 & 18 \end{pmatrix}$$

On définit un pivot et on soustrait les lignes suivantes :

$$\begin{pmatrix} 2 & 1 & 3 \\ 1 & 5 & 5 \\ 3 & 5 & 9 \end{pmatrix}$$

A la place des 0 on garde les facteurs a_{ik} utilisés pour réduire la ligne.

$$\begin{pmatrix} 2 & 1 & 3 \\ 1 & 5 & 5 \\ 3 & 1 & 4 \end{pmatrix}$$

On peux alors simplement séparer la matrice en deux pour obtenir la décomposition LU.

Exemple de la décomposition LU

Très similaire au pivot de Gauss, mais avec enregistrement des facteurs !

$$\begin{pmatrix} 2 & 1 & 3 \\ 2 & 6 & 8 \\ 6 & 8 & 18 \end{pmatrix}$$

On définit un pivot et on soustrait les lignes suivantes :

$$\begin{pmatrix} 2 & 1 & 3 \\ 1 & 5 & 5 \\ 3 & 5 & 9 \end{pmatrix}$$

A la place des 0 on garde les facteurs a_{ik} utilisés pour réduire la ligne.

$$\begin{pmatrix} 2 & 1 & 3 \\ 1 & 5 & 5 \\ 3 & 1 & 4 \end{pmatrix}$$

On peux alors simplement séparer la matrice en deux pour obtenir la décomposition LU.

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

**Attention : On ne peux plus passer les pivots à 1. Le facteur gardé doit etre le rapport entre les deux lignes soustraites*