M5 E1 : Analyse Numérique (Cours magistral)

Dr. Yassine Sabbar

High school of technology
Ibn Zohr University
Agadir, Morocco
Email : yassine.sabbar@edu.uiz.ac.ma

18 février 2020

Analyse Numérique : organisation et évaluation

1. Organisation :
    1. Cours : 9 séances d'1h30.
    2. TDs et TPs : 25 h (chaque groupe).
        1. 8 séances de TDs d'1h30.
        2. 8 séances de TPs Matlab : d'1h30.

2. Evaluation :
    1. Note du TP (Compte rendu) - 1/4 note finale.
    2. 1 examen final de 1h30 - 3/4 note finale.

Plan du cours

1. Arithmétique des ordinateurs et rappels mathématiques.
2. Résolution d'un système d'équations linéaires (Partie 1) : méthodes directes.
3. Conditionnement d'une matrice pour la résolution d'un système linéaire.
4. Résolution d'un système d'équations linéaires (Partie 2) : méthodes itératives.
5. Interpolation polynomiale.

Introduction

1. Analyse numérique est une discipline des mathématiques appliquées, dont les applications sont nombreuses dans les systèmes industrielles.
2. Un algorithme est une suite finie et claire d'opérations ou d'instructions permettant de résoudre une classe de problèmes.
3. Analyse numérique est l'étude des algorithmes permettant de résoudre numériquement par discrétisation les problèmes de mathématiques continues (distinguées des mathématiques discrètes).
4. L'objectif principal est de proposer un algorithme de résolution (ou approximation) numérique des problèmes.
5. Elle s'intéresse aux manipulations des nombres par les machines pour obtenir une solution acceptable.

Chapitre 1 :
Arithmétique des ordinateurs et rappels mathématiques

La numération (1) : Le système décimal

Les nombres que nous utilisons habituellement sont ceux de la base 10 (système décimal). Nous disposons de dix chiffres différents de 0 à 9 pour écrire tous les nombres.

Exemple

Soit un nombre décimal $N = 2348$. Ce nombre est la somme de 8 unités, 4 dizaines, 3 centaines et 2 milliers.
Nous pouvons écrire

$$\begin{aligned} N &= (2x1000) + (3x100) + (4x10) + (8x1) \\ &= (2 \times 10^3) + (3 \times 10^2) + (4 \times 10^1) + (8 \times 10^0). \end{aligned}$$

10 représente la base et les puissances de 0 à 3 le rang de chaque chiffre.

La numération (2) : Le système binaire (1)

Dans les domaines de l'automatisme, de l'électronique et de l'informatique, nous utilisons la base 2. Tous les nombres s'écrivent avec deux chiffres uniquement (0 et 1) car les systèmes technologiques ont souvent deux états stables :

1. Un interrupteur est ouvert ou fermé.
2. Une diode est allumée ou éteinte.
3. Une tension est présente ou absente.
4. Un champ magnétique est orienté Nord-Sud ou Sud-Nord.

La présence d'une tension sera par exemple notée 1 et l'absence 0. Le chiffre binaire qui peut prendre ces deux états est nommé "Bit" (Binary digit).

1. Avec un bit nous pouvons coder deux états.
2. Avec deux bits nous pouvons coder quatre états.
3. Avec trois bits nous pouvons coder huit états.
4. A chaque nouveau bit, le nombre de combinaisons possibles est doublé. Ce nombre est égal à 2 puissance N (N étant le nombre de bits).

La numération (2) : Le système binaire (2)

1. Un groupe de bits est appelé un mot, un mot de huit bits est nommé un octet (byte).
2. Avec un octet, nous pouvons écrire 2 puissance 8 = 256 nombres binaires de 0 à 255
3. Exemple :

$$\begin{aligned} 1011_{(2)} &= (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) \\ &= 11_{(10)} \end{aligned}$$

4. Conversion d'un nombre binaire en décimal : Il suffit de faire la somme des poids de chaque bit à 1.
5. Conversion d'un nombre décimal en binaire : Méthode par divisions.
6. Exemple $172_{(10)} = 10101100_{(2)}$.

La numération (3) : Le système hexadécimal (1)

La manipulation des nombres écrits en binaire est difficile pour l'être humain et la conversion en décimal n'est pas simple. C'est pourquoi nous utilisons de préférence le système hexadécimal (base 16). Pour écrire les nombres en base 16 nous devons disposer de 16 chiffres, pour les dix premiers, nous utilisons les chiffres de la base 10, pour les suivant nous utiliserons des lettres de l'alphabet.

Décimale	0	1	2	3	4	5	6	7	8	9
Hexadécimale	0	1	2	3	4	5	6	7	8	9

Décimale	10	11	12	13	14	15
Hexadécimale	A	B	C	D	E	F

Les règles sont ici aussi les mêmes que pour le décimal.

$$\begin{aligned} A3F_{(16)} &= (A \times 16^2) + (3 \times 16^1) + (F \times 16^0) \\ &= (10 \times 256) + (3 \times 16) + (15 \times 1) \\ &= 2623_{(10)} \end{aligned}$$

La numération (3) : Le système hexadécimal (2)

1. Correspondance entre binaire et hexadécimal : La conversion du binaire en hexadécimal est très simple, c'est d'ailleurs la raison pour laquelle nous utilisons cette base. Il suffit de faire correspondre un mot de quatre bits (quartet) à chaque chiffre hexadécimal.
   Exemple :

$$4D7F_{(16)} = 0100 \quad 1101 \quad 0111 \quad 1111_{(2)}$$

2. Correspondance entre décimal et hexadécimal : La méthodes par divisions s'applique comme en binaire.
   Exemple :

$$2623_{(10)} = A3F_{(16)}$$

La numération (4) : Le système OCTAL

1. Le système de numération à base 8 est un moyen de représenter les nombres avec 8 symboles.
2. Octal vers décimal :

$$\begin{aligned} 4321_{(8)} &= (4 \times 8^3) + (3 \times 8^2) + (2 \times 8^2) + (1 \times 8^0) \\ &= 2257_{(10)} \end{aligned}$$

3. Décimal en Octal :

$$458_{(10)} = 712_{(8)} = 111 \;\; 001 \;\; 010_{(2)}$$

La numération (5) : binaire - octale - hexadécimale

1. Représentation binaire et octale :

Octale	0	1	2	3	4	5	6	7
Binaire	000	001	010	011	100	101	110	111

2. Représentation binaire et hexadécimale :

Hexadécimale	0	1	2	3	4	5	6
Binaire	0000	0001	0010	0011	0100	0101	0110

7	8	9	A	B	C	D	E	F
0111	1000	1001	1010	1011	1100	1101	1110	1111

Codage des nombres à virgule (1)

1. Un nombre décimal est composé d'une partie entière et d'une partie fractionnaire après la virgule.
2. En base B, ce nombre X s'écrit :

$$X_{(B)} = a_n a_{n-1} \dots a_0, b_1 \dots b_m$$

Il se convertit en décimal en :

$$X_{(10)} = a_n B^n + \dots + a_0 B^0 + b_1 B^{-1} + \dots + b_m B^{-m}.$$

Codage des nombres à virgule (2)

Exemples :

1. $$128,75 = 1 \times 10^2 + 2 \times 10^1 + 8 \times 10^0 + 7 \times 10^{-1} + 5 \times 10^{-2}$$

2. $$\begin{aligned} 101,01_{(2)} &= 1 \times 2^2 + 1 \times 2^0 + 1 \times 2^{-2} \\ &= 4 + 1 + 0.25 \\ &= 5,25 \end{aligned}$$

3. $$\begin{aligned} AE,1F_{(16)} &= 10 \times 16^1 + 14 \times 16^0 + 1 \times 16^{-1} + 15 \times 16^{-2} \\ &= 174,12109375. \end{aligned}$$

Conversion des nombres à virgule en base B (1)

1. Pour la partie entière, on fait comme pour les entiers.
2. Pour la partie décimale :
    1. On multiplie la partie entière par B.
    2. On note la partie entière obtenue.
    3. On recommence avec la partie décimale restante.
    4. On s'arrête quand la partie décimale est nulle ou quand la précision souhaitée est atteinte
3. La partie décimale est la concaténation des parties entières obtenues dans l'ordre de leur calcul.

Conversion des nombres à virgule en base B (2)

Exemple : conversion de $28,8625$ en binaire.

1. Conversion de 28 : $11100_{(2)}$.
2. Conversion de 0,8625 :

$$
\begin{aligned}
0,8625 \times 2 &= 1,725 = 1 + 0,725 \\
0,725 \times 2 &= 1,45 = 1 + 0,45 \\
0,45 \times 2 &= 0,9 = 0 + 0,9 \\
0,9 \times 2 &= 1,8 = 1 + 0,8 \\
0,8 \times 2 &= 1,6 = 1 + 0,6 \\
0,6 \times 2 &= 1,2 = 1 + 0,2 \\
0,2 \times 2 &= 0,4 = 0 + 0,4 \\
0.4 \times 2 &= 0.8 = 0 + 0.8 \ldots
\end{aligned}
$$

Donc, $28,8625$ peut être représenté par $11100,11011100\ldots_{(2)}$.

Comment les réels sont-ils représentés dans un ordinateur ?

Définition

La virgule flottante est une méthode d'écriture de nombres réels fréquemment utilisée dans les ordinateurs. Elle consiste à représenter un nombre réel par

1. un signe (égal à -1 ou 1),
2. une mantisse (aussi appelée significande).
3. et un exposant (entier relatif, généralement borné).

Théorème (Système des nombres à virgule flottante)

Soit α un entier strictement supérieur à 1. Tout nombre réel x non nul peut se représenter sous la forme

$$x = sgn(x)\alpha^e \sum_{k\geq 1} \frac{a_k}{\alpha^k},$$

où, $sgn(x) \in \{+,-\}$ est le signe de x, les a_k sont des entiers tels que $0 < a_1 \leq \alpha - 1$ et $0 \leq a_k \leq \alpha - 1$ pour $k \geq 2$, et $e \in \mathbb{Z}$.

Exemple

1. Système décimal : $\alpha = 10$ et $a_k \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
   - $0.0038 = 0.38 \times 10^{-2} = +10^{-2}\left(\frac{3}{10} + \frac{8}{10^2}\right)$.
   - $\frac{1}{7} = 0.142857... = +10^{0}\left(\frac{1}{10} + \frac{4}{10^2} + \frac{2}{10^3} + \frac{8}{10^4} + ...\right)$.
   - $-\sqrt{2} = -1.4142... = -10^{1}\left(\frac{1}{10} + \frac{4}{10^2} + \frac{3}{10^3} + \frac{4}{10^4} + ...\right)$.
2. Ordinateurs : $\alpha = 2$ (numération binaire), $\alpha = 8$ (numération octale), ou encore $\alpha = 16$ (numération hexadécimale).

Précision

Définition

On définit l'ensemble $F \subset \mathbb{R}$ par :

$$F = \left\{ y \in \mathbb{R} | y = \pm \alpha^e \left(\frac{a_1}{\alpha} + \frac{a_2}{\alpha^2} + ... + \frac{a_t}{\alpha^t} \right), \quad e_{\min} \leq e \leq e_{\max} \right\},$$

ou encore

$$F = \left\{ y \in \mathbb{R} | y = \pm \alpha^{e-t} m, \quad e_{\min} \leq e \leq e_{\max} \right\}.$$

F est un système de nombres à virgule flottante (floating point number system).
Notation $F(\alpha$(la base), t(la précision), e(l'exposant), m(la mantisse)).

Définition (Précision machine)

les nombres sont représentées sous format binaire sur la machine par un nombre fini de bites, par suite on a une valeur maximale et une autre minimale (précision machine) de représentation.

Standard IEEE 754 et epsilon machine

1. Le principe de la virgule flottante normalisé en IEEE 754 (Institute of Electrical and Electronics Engineers,1985) est de représenter les nombres en notation scientifique en utilisant, la base, les puissances, les exposants, le signe et la mantisse.
2. Dans le standard IEEE 754 utilisé par Matlab, on a $\alpha = 2$ et :
   1. en simple précision : $t = 24$, $e_{\min} = -125$, $e_{\max} = 128$,
   2. en double précision : $t = 53$, $e_{\min} = -1020$, $e_{\max} = 1024$.

Définition

On appelle epsilon machine et on note ϵ_M, la distance de 1 au nombre flottant suivant.

Proposition

Pour $F(\alpha, t, e, m)$, on a $\epsilon_M = \alpha^{1-t}$.

La normalisation IEEE 754 des nombres réels à virgule flottante en base 2

La normalisation IEEE 754 permet d'avoir des comportements communs pour des programmes identiques sur des machines différentes.

1. la mantisse va dénoter la séparation de la précision.
2. l'exposant va exprimer l'ordre de grandeur des nombres.
3. le signe va dénoter les positifs et négatifs.
4. la base va permettre ensuite l'utilisation d'algorithmes appropriés.

La représentation que nous allons suivre est :

$$Signe - exposant - mantisse.$$

Représentation IEEE 754 des nombres réels à virgule flottante en base 2

1. les nombres "simple précision" codés sur 32 bits (emplacement du codage du type INT en langage C).
2. les nombres "double précision" codés sur 64 bits (emplacement du codage du type DOUBLE en langage C).
3. les nombres "précision étendue" codés sur 80 bits (emplacement du codage du type LONG DOUBLE en langage C).

Découpage en bits des précisions IEEE 754 :

La représentation	Signe	Exposant	Mantisse
Simple précision	1 bit	8 bits	23 bits
Double précision	1 bit	11 bits	52 bits
Précision étendue	1 bit	15 bits	64 bits

TABLE: Virgule flottante – La norme IEEE754.

Exposant en codage IEEE 754

Pour permettre une meilleure dynamique de codage des nombres, nous allons disposer d'un pivot pour coder l'exposant. Le pivot va diviser l'espace de codage de l'exposant en exposant positif et négatif.

Découpage en bits des précisions IEEE 754 :

La représentation	Place en bits	Pivot en binaire	Base 10
Simple précision	8 bits	01111111	127
Double précision	11 bits	01111111111	1023
Précision étendue	15 bits	011111111111111	16383

TABLE: Valeurs de pivot en fonction des places pour l'exposant en codage IEEE754.

Mantisse en codage IEEE754

Nous sommes en notation scientifique et donc nous exprimons notre résultat en fonction de la base numérique donnée. Cette proposition d'algorithme est adaptable pour toutes les bases passées en paramètre.

Les valeurs significatives pour les bases en notation scientifique sont :

1. en base 2 : 1.
2. en base 10 : 1 à 9.
3. en base 8 : 1 à 7.

Exemple

1. Le nombre en notation scientifique retenu de $1110_{(2)}$ serait : 1.110×2^3.
2. Le nombre en notation scientifique retenu de $4321_{(10)}$ serait : 4.321×10^3.

Application

Donnez la représentation flottante, en simple précision (Norme IEEE 754), de 18.125 !

1. Signe $= 0$ (un nombre positif).
2. Conversion en binaire :
   1. Partie décimale : $0.125 = 0.001_{(2)}$ ($\times 2$).
   2. Partie entière : $18 = 10010_{(2)}$ (le reste de la division sur 2).
3. $|18.125| = 18.125 = 10010.001_{(2)}$ (conversion totale en valeur absolue).
4. $18.125 = (1.0010001)_{(2)} \times 2^4$ (valeur significative).
5. $m = 0010001...0_{(2)}$ et $e = 4$ (la mantisse et l'exposant).
6. $E = e + biais = 4 + 127 = 10000011_{(2)}$ (normalisation IEEE 754).
7. $18.125 \leftarrow 0\ 10000011\ 00100010000000000000000$ (32 bits).

Approximation

Exemple

Trouver la valeur de π ou racine de 2 !
Avec la donnée de n'importe quelle erreur, on peut s'approcher de la valeur recherchée, la différence entre le résultat exact et le résultat approché est appelée erreur.

Définition

Dans le calcul scientifique, faire attention de trois erreurs :

1. Les erreurs sur les données (dues à l'imprécision des mesures physiques ou au fait que les données proviennent elle même d'un calcul approché).
2. Les erreurs d'arrondi (dues au fait que la machine ne peut représenter les nombres réels qu'avec un nombre fini de chiffres).
3. Les erreurs d'approximation ou de discrétisation (Ce sont les erreurs qu'on commet, par exemple, lorsqu'on calcule une intégrale à l'aide d'une somme finie, une dérivée à l'aide de différences finies ou bien la somme d'une série infinie à l'aide d'un nombre fini de ses termes).

Erreur d'arrondi

Définition

Soit x un réel et $\tilde{x}$ une valeur approchée de x.

1. L'erreur absolue e est défini par $e = |x - \tilde{x}|$.
2. L'erreur relative est $|e/\tilde{x}|$.
3. Le pourcentage d'erreur est l'erreur relative multipliée par 100.

Remarque

En pratique, on ne connait en général pas la valeur exacte x mais on peut souvent avoir une idée de l'erreur maximale e que l'on a pu commettre : dans ce cas, on majore la quantité $|e/x|$.

Erreur amont et erreur aval

1. Considérons un problème que l'on résout à l'aide d'un algorithme numérique : entrée $x \to y = f(x)$.
2. En pratique, compte tenu des erreurs d'arrondis, étant donnée une entrée x, nous allons obtenir une sortie $\tilde{y} \neq y = f(x)$.
3. Erreur aval : $|\tilde{y} - y|$.
4. Erreur amont (ou erreur inverse) : plus petit δx tel que la solution algébrique $f(x + \delta x)$ correspondant à l'entrée $x + \delta x$ soit égale à $\tilde{y}$.
5. Erreur aval $\approx$ erreur amont $\times$ conditionnement.
6. Erreur amont plus intéressante (en pratique, nous ne connaissons en général qu'une valeur approchée de l'entrée).

Éléments d'analyse matricielle (1)

Définition

On appelle MATRICE $m \times n$ (ou d'ordre $m \times n$) à coefficients dans $\mathbb{K}$ tout tableau de m lignes et n colonnes d'éléments de $\mathbb{K}$. L'ensemble des matrices $m \times n$ à coefficients dans $\mathbb{K}$ est noté $\mathcal{M}_{m,n}(\mathbb{K})$. On convient de noter a_{ij} l'élément de la matrice situé sur la i-ème ligne et j-ème colonne ($1 \leq i \leq m$ et $1 \leq j \leq n$). Une matrice A est représentée entre deux parenthèses

$$A = \begin{pmatrix} a_{11} & \dots & a_{1j} & \dots & a_{1n} \\ \vdots & & \vdots & & \vdots \\ a_{i1} & \dots & a_{ij} & \dots & a_{in} \\ \vdots & & \vdots & & \vdots \\ a_{m1} & \dots & a_{mj} & \dots & a_{mn} \end{pmatrix}$$

ou encore

$$A = (a_{ij})_{\substack{1 \leq j \leq n \\ 1 \leq i \leq m}}.$$

Éléments d'analyse matricielle (2)

1. Si $m = n$, on dit qu'on a une MATRICE CARRÉE. L'ensemble des matrices carrées d'ordre n à coefficients dans $\mathbb{K}$ est noté $\mathcal{M}_n(\mathbb{K})$.
2. Une matrice $m \times 1$ est appelée VECTEUR-COLONNE et une matrice $1 \times n$ est appelée VECTEUR-LIGNE.
3. La MATRICE NULLE, notée $O_{m,n}$, est la matrice dont tous les éléments sont nuls.
4. On appelle MATRICE DIAGONALE toute matrice carrée $\mathbb{D} = (d_{ij})_{1 \leq i,j \leq n}$ telle que $i \neq j \Rightarrow d_{ij} = 0$.
5. La MATRICE IDENTITÉ d'ordre n, notée I_n, est la matrice diagonale $\text{Diag}(1, 1, ..., 1)$.
6. On dit qu'une matrice carrée $A = (a_{ij})_{1 \leq i,j \leq n}$ est
   1. TRIANGULAIRE SUPÉRIEURE si $i > j \Rightarrow a_{ij} = 0$.
   2. TRIANGULAIRE INFÉRIEURE si $i < j \Rightarrow a_{ij} = 0$.

Éléments d'analyse matricielle (3)

1. Si $A = (a_{ij})_{\substack{1\leq j\leq n\\ 1\leq i\leq m}}$ est une matrice $\mathcal{M}_{m,n}(\mathbb{R})$, on définit la matrice TRANSPOSÉE de A, notée A^T par $A^T = (a_{ji})_{\substack{1\leq i\leq m\\ 1\leq j\leq n}}$. C'est donc une matrice de $\mathcal{M}_{n,m}(\mathbb{R})$ obtenue en échangeant lignes et colonnes de la matrice initiale.
2. Une matrice A est dite SYMÉTRIQUE si $A^T = A$, i.e. si $a_{ij} = a_{ji}$ pour tout $i \neq j$.
3. Une matrice carrée $A \in \mathcal{M}_n(\mathbb{K})$ est dite INVERSIBLE (ou régulière) si elle est symétrisable pour le produit matriciel, autrement dit s'il existe une matrice $B \in \mathcal{M}_n(\mathbb{K})$ telle que $AB = BA = I_n$.
4. Une matrice carrée $A \in \mathcal{M}_n(\mathbb{K})$ est dite ORTHOGONALE si elle est inversible et $A^T A = AA^T = I_n$.

Opérations élémentaires sur les matrices

Définition (Opérations élémentaires sur les lignes d'une matrices)

Les opérations (ou manipulations) élémentaires sur les lignes d'une matrices $M \in \mathcal{M}_{m,n}(\mathbb{K})$ sont :

1. la multiplication d'une ligne L_i par un scalaire non nul α : $L_i \leftarrow \alpha L_i$,
2. l'addition d'un multiple d'une ligne αL_j à une autre ligne L_i : $L_i \leftarrow L_i + \alpha L_j$,
3. l'échange de deux lignes : $L_i \leftrightarrow L_j$.

Définition (Opérations élémentaires sur les colonnes d'une matrices)

Les opérations élémentaires sur les colonnes d'une matrice $M \in \mathcal{M}_{m,n}(\mathbb{K})$ sont :

1. la multiplication d'une colonne C_i par un scalaire non nul α : $C_i \leftarrow \alpha C_i$,
2. l'addition d'un multiple d'une colonne αC_j à une autre ligne C_i : $C_i \leftarrow C_i + \alpha C_j$,
3. l'échange de deux colonnes : $C_i \leftrightarrow C_j$.

Systèmes linéaires

1. Beaucoup de problèmes se réduisent à la résolution numérique d'un système d'équations linéaires.
2. Deux grandes classes de méthodes :
   ♣ Méthodes directes : déterminent explicitement la solution après un nombre fini d'opérations arithmétiques.
   ♣ Méthodes itératives (sur $\mathbb{R}$ ou $\mathbb{C}$) : consistent à générer une suite qui converge vers la solution du système.
3. Autres méthodes non abordées dans ce cours :
   ♣ Méthodes intermédiaires : Splitting, décomposition incomplètes.
   ♣ Méthodes probabilistes comme celle de Monte-Carlo.

Chapitre 2
Résolution d'un système d'équations linéaires : méthodes directes

Objet de l'étude

$$(S)\begin{cases} a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n = b_2 \\ . \\ . \\ . \\ a_{n1}x_1 + a_{n2}x_2 + ... + a_{nn}x_n = b_n \end{cases}$$

1. Données : les a_{ij} et $b_1, ..., b_n$ dans $\mathbb{R}$ où $\mathbb{C}$.
2. Inconnues : $x_1, ..., x_n$ dans $\mathbb{K}$ avec $\mathbb{K} = \mathbb{R}$ où $\mathbb{C}$.

Écriture matricielle

$$(S): \quad Ax = b.$$

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ a_{n1} & \dots & \dots & a_{nn} \end{pmatrix} \in \mathcal{M}_{n\times n}(\mathbb{K})$$

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{K}^n \quad \text{et} \quad b = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix} \in \mathbb{K}^n.$$

Remarque

Dans ce chapitre, A est une matrice inversible !

Motivation : Pourquoi ce problème se pose-t-il ?

1. En effet, les formules de Cramer donnent la solution : $\forall i \in \{1, ..., n\}$

$$x_i = \frac{\begin{vmatrix} a_{11} & \dots & a_{1i-1} & b_1 & \dots & a_{1i+1} & \dots & a_{1n} \\ \vdots & & & \vdots & & & & \vdots \\ a_{n1} & \dots & a_{ni-1} & b_n & \dots & a_{ni+1} & \dots & a_{nn} \end{vmatrix}}{\det A}.$$

2. Regardons le nombre d'opérations nécessaires !

Théorème

Le nombre d'opérations nécessaires pour résoudre le système à l'aide des formules de Cramer est de $(n+1)(nn! - 1)$ *opérations à virgule flottante.*

Exemple

1. Lorsque $n = 100$, nombre d'opérations de l'ordre de 9.4×10^{161}.
2. Ordi. fonctionnant à 100 megaflops, environ 3.10×10^{146} années ! !

Résolution d'un système triangulaire

1. Idée des méthodes directes : se ramener à la résolution d'un système triangulaire.
2. A triangulaire supérieure : (S) s'écrit

$$(S)\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n &= b_1 \\ a_{22}x_2 + \ldots + a_{2n}x_n &= b_2 \\ \ddots \quad \ddots \quad \ddots &= \vdots \\ a_{nn}x_n &= b_n \end{cases}$$

3. A inversible $\Rightarrow$ les a_{ii} sont non nuls.
4. Système facile à résoudre : algorithme de substitution rétrograde
5. Si A triangulaire inférieure : algorithme de substitution progressive.

Opérations et propriétés

Théorème

La résolution d'un système d'équations linéaires triangulaire se fait en n^2 opérations à virgule flottante.

Lemme

Soient $A, B \in \mathcal{M}_{n\times n}$ deux matrices triangulaires supérieures. On a alors les résultats suivants :

1. *AB est une matrice triangulaire supérieure.*
2. *Si A et B sont à diagonale unité (i.e. n'ont que des 1 sur la diagonale), alors AB est à diagonale unité.*
3. *Si A est inversible, alors A^{-1} est aussi triangulaire supérieure.*
4. *Si A sont à diagonale unité, alors A^{-1} est à diagonale unité.*

3 méthodes directes étudiées dans la suite

1. Méthode de Gauss
   ♣ Système $\rightsquigarrow (MA)x = Mb$ avec MA triang. sup. (sans calculer explicitement M).
   ♣ Associée à la factorisation $A = LU$ de la matrice A avec L triang. inf. et U triang. sup., $Ax = b \Leftrightarrow Ly = b$, $Ux = y$.
2. Méthode de Cholesky (Paragraphe supprimé)
   ♣ Associée à la factorisation de Cholesky $A = R^T R$ avec R triang. sup., $Ax = b \Leftrightarrow R^T y = b$, $Rx = y$.
   ♣ Méthode valable pour A symétrique et définie positive.
3. Méthode de Householder (Paragraphe supprimé)
   ♣ Associée à la factorisation $A = QR$ avec R triang. sup. et Q orthogonale, Q produit de $n-1$ matrices de Householder H_i.
   ♣ $Ax = b$ s'écrit alors $H_{n-1}...H_2H_1Ax = H_{n-1}...H_2H_1b$ facile à résoudre car $H_{n-1}...H_2H_1A$ triang. sup.

Méthode de Gauss : description (1)

Étape initiale

1. $(S) : Ax = b$ avec A inversible.
2. On pose $b^{(1)} = b$ et $A^{(1)} = A = (a_{ij}^{(1)}) \rightsquigarrow (S^{(1)}) = A^{(1)}x = b^{(1)}$.

Étape 1

1. A inversible $\Rightarrow$ on suppose (quitte à permuter lignes) $a_{11}^{(1)} \neq 0$. C'est le premier pivot de l'élimination de Gauss.
2. Pour $i = 2, ..., n$, on remplace L_i par $L_i - g_{i1}L_1$ où $g_{i1} = \frac{a_{i1}^{(1)}}{a_{11}^{(1)}}$.

Méthode de Gauss : description (2)

1. On obtient alors $(S^{(2)}) : A^{(2)}x = b^{(2)}$ avec :

$$\begin{cases} a_{1j}^{(2)} & = a_{1j}^{(1)}, \quad j = 1, ..., n \\ a_{i1}^{(2)} & = 0 \quad i = 2, ..., n \\ a_{ij}^{(2)} & = a_{ij}^{(1)} - g_{i1} a_{1j}^{(1)}, \quad i, j = 2, ..., n \\ b_1^{(2)} & = b_1^{(1)} \\ b_i^{(2)} & = b_i^{(1)} - g_{i1}^{(1)} b_1^{(1)}, \quad i = 2, ..., n \end{cases}$$

2. La matrice $A^{(2)}$ et le vecteur $b^{(2)}$ sont donc de la forme :

$$A^{(2)} = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} \\ 0 & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ 0 & a_{n2}^{(2)} & \dots & a_{nn}^{(2)}, \end{pmatrix} \qquad b^{(2)} = \begin{pmatrix} b_1^{(1)} \\ b_2^{(2)} \\ \vdots \\ b_n^{(2)} \end{pmatrix}$$

Méthode de Gauss : description (3)

Étape k

1. On a ramené le système à $(S^{(k)}) = A^{(k)}x = b^{(k)}$ avec

$$A^{(k)} = \begin{pmatrix} a_{11}^{(1)} & & \cdots & \cdots & a_{1k}^{(1)} & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & & & a_{2k}^{(2)} & \cdots & a_{2n}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & & a_{3k}^{(3)} & \cdots & a_{3n}^{(3)} \\ \vdots & \ddots & \ddots & \ddots & \vdots & & \vdots \\ 0 & \cdots & 0 & 0 & a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} \\ \vdots & & \vdots & 0 & a_{k+1\,k}^{(k)} & \cdots & a_{k+1\,n}^{(k)} \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 0 & \cdots & 0 & 0 & a_{nk}^{(k)} & \cdots & a_{nn}^{(k)} \end{pmatrix}$$

Méthode de Gauss : description (4)

1. A inversible $\Rightarrow$ on suppose (quitte à permuter lignes) $a_{kk}^{(k)} \neq 0$ (C'est le K-ème pivot de l'élimination de Gauss).
2. Par le même principe qu'à l'étape 1 et en utilisant $g_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$ pour $i > k$, on obtient $(S^{(k+1)}) = A^{(k+1)}x = b^{(k+1)}$ avec

$$A^{(k+1)} = \begin{pmatrix} a_{11}^{(1)} & & \cdots & \cdots & a_{1\,k+1}^{(1)} & \cdots & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & & & a_{2k}^{(2)} & \cdots & \cdots & a_{2n}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & & a_{3k}^{(3)} & \cdots & \cdots & a_{3n}^{(3)} \\ \vdots & \ddots & \ddots & \ddots & \vdots & & & \vdots \\ 0 & \cdots & 0 & 0 & a_{kk}^{(k)} & \cdots & \cdots & a_{kn}^{(k)} \\ \vdots & & \vdots & 0 & 0 & a_{k+1\,k+1}^{(k+1)} & \cdots & a_{k+1\,n}^{(k+1)} \\ \vdots & & \vdots & \vdots & \vdots & & & \vdots \\ 0 & \cdots & 0 & 0 & 0 & a_{n\,k+1}^{(k+1)} & \cdots & a_{nn}^{(k+1)} \end{pmatrix}$$

Méthode de Gauss : description (5)

Étape n-1

1. Le système $(S^{(n)}) = A^{(n)}x = b^{(n)}$ obtenu est triangulaire supérieure avec

$$A^{(n)} = \begin{pmatrix} a_{11}^{(1)} & & \cdots & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & & & a_{2n}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & & a_{3n}^{(3)} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & a_{nn}^{(n)} \end{pmatrix}$$

2. On peut le résoudre par l'algorithme de substitution rétrograde.

Méthode de Gauss : exemple (1)

$$(S) = (S^{(1)}) \begin{cases} x_1 + 2x_2 + 5x_3 &= 1, \\ 3x_1 + 2x_2 - x_3 &= \frac{1}{2}, \\ 5x_2 + 3x_3 &= 1, \end{cases}$$

1. Le premier pivot de l'élimination de Gauss est donc $a_{11}^{(1)} = 1$ et on a $g_{21}^{(1)} = 3$, $g_{31}^{(1)} = 0$. La première étape fournit donc

$$(S^{(2)}) \begin{cases} x_1 + 2x_2 + 5x_3 &= 1, \\ -4x_2 - 16x_3 &= \frac{-5}{2}, \\ 5x_2 + 3x_3 &= 1, \end{cases}$$

Méthode de Gauss : exemple (2)

1. Le second pivot de l'élimination de Gauss est donc $a_{22}^{(2)} = -4$ et on a $g_{32}^{(2)} = \frac{-5}{4}$. On obtient donc le système

$$(S^{(3)}) \begin{cases} x_1 + 2x_2 + 5x_3 &= 1, \\ -4x_2 - 16x_3 &= \frac{-5}{2}, \\ -17x_3 &= \frac{-17}{8}, \end{cases}$$

Remarque

Au cours de l'exécution de l'élimination de Gauss, si on tombe sur un pivot nul, alors on permute la ligne en question avec une ligne en dessous pour se ramener à un pivot non nul (ceci est toujours possible car A est supposée inversible).

Lien avec la factorisation LU d'une matrice (1)

Définition

On appelle factorisation LU de A une facto. $A = LU$ avec L triang. inf. et U tring. sup. (de la même taille que A).

Lemme

A l'étape k de l'élimination de Gauss, on a $A^{(k+1)} = G_k A^{(k)}$ où

$$G_k = \begin{pmatrix} 1 & & (0) & & 0 & \dots & 0 \\ & \ddots & & & \vdots & & \vdots \\ & (0) & & 1 & 0 & \dots & 0 \\ 0 & \dots & 0 & -g_{k+1\,k} & 1 & & (0) \\ \vdots & & \vdots & \vdots & & \ddots & \\ 0 & \dots & 0 & -g_{n\,k} & (0) & & 1 \end{pmatrix}, \qquad g_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$$

On a de plus $b^{(k+1)} = G_k b^{(k)}$.

Lien avec la factorisation LU d'une matrice (2)

Définition

Soit $A \in \mathcal{M}_{n\times n}(\mathbb{K})$. Les mineurs fondamentaux D_k, $k = 1, ..., n$ de A sont les déterminants des sous-matrices de A formées par les k premières lignes et les k premières colonnes de A :

$$D_k = \det((a_{ij})_{1\leq i,j\leq k}), \quad k = 1, ..., n.$$

Théorème

Soit $A \in \mathcal{M}_{n\times n}(\mathbb{K})$ une matrice carrée inversible. Les propriétés suivantes sont équivalentes :

1. *L'élimination de Gauss s'effectue sans permutation de lignes.*
2. *Il existe $L \in \mathcal{M}_{n\times n}(\mathbb{K})$ triangulaire inférieure inversible et $U \in \mathcal{M}_{n\times n}(\mathbb{K})$ triangulaire supérieure inversible telles que $A = LU$.*
3. *Tous les mineurs fondamentaux de A sont non nuls.*

Lien avec la factorisation LU d'une matrice (3)

Lemme

Avec les notations précédentes, on a

$$(G_{n-1}G_{n-2}\dots G_1)^{-1} = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ g_{21} & 1 & \ddots & & \vdots \\ g_{31} & g_{32} & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ g_{n1} & g_{n2} & \dots & g_{n\,n-1} & 1 \end{pmatrix}$$

Lien avec la factorisation LU d'une matrice (4)

Soit $A \in \mathcal{M}_{n\times n}(\mathbb{K})$ une matrice carrée inversible. Si tous les mineurs fondamentaux de A sont non nuls, alors avec les notations précédentes, l'élimination de Gauss fournit la factorisation LU de A suivante :

$$A = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ g_{21} & 1 & \ddots & & \vdots \\ g_{31} & g_{32} & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ g_{n1} & g_{n2} & \dots & g_{n\,n-1} & 1 \end{pmatrix} \begin{pmatrix} a_{11}^{(1)} & & \dots & \dots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & & & a_{2n}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & & a_{3n}^{(3)} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & a_{nn}^{(n)} \end{pmatrix}$$

Remarque

la matrice L obtenue est à diagonale unité.

Factorisation LU : exemple

Pour la matrice du système

$$(S^{(2)}) \begin{cases} x_1 + 2x_2 + 5x_3 &= 1, \\ -4x_2 - 16x_3 &= \frac{-5}{2}, \\ 5x_2 + 3x_3 &= 1, \end{cases}$$

On a :

$$\begin{pmatrix} 1 & 2 & 5 \\ 3 & 2 & -1 \\ 0 & 5 & 3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 0 & \frac{-5}{4} & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 & 5 \\ 0 & -4 & -16 \\ 0 & 0 & -17 \end{pmatrix}.$$

Lien avec la factorisation LU d'une matrice (5)

Proposition

Soit $A \in \mathcal{M}_{n\times n}(\mathbb{K})$ une matrice carrée inversible admettant une factorisation LU. Alors il existe une unique factorisation LU de A avec L à diagonale unité.

1. Lorsque A admet une factorisation LU, la résolution du système d'équations linéaires (S) : A x = b se ramène à la résolution de deux systèmes linéaires triangulaires. En effet :

$$Ax = b \Leftrightarrow LUx = b \Leftrightarrow \begin{cases} Ly = b, \\ Ux = y. \end{cases}$$

2. En pratique, on résout donc d'abord Ly= b puis connaissant y on résout Ux = y.

Chapitre 3
Conditionnement d'une matrice pour la résolution d'un système linéaire

Valeurs propres et vecteurs propres

Définition

1. Un vecteur x est un vecteur propre de la matrice A carrée de taille $n \times n$ si $Ax = \lambda x$ pour un certain réel λ.

2. Un réel λ est une valeur propre de A si il y a une solution non-triviale (autre que 0) à l'équation $Ax = \lambda x$. Une telle solution est alors appelée vecteur propre associé à la valeur propre λ.

3. L'ensemble des solutions de $(A - \lambda I)x = 0$ est appelé espace propre de A associé à la valeur propre λ.

L'équation caractéristique

Si on connaît une valeur propre λ de A. On trouve les vecteurs propres de A en résolvant

$$(A - \lambda I)x = 0.$$

Mais comment trouve-t-on les valeurs propres de A ?

1. x doit être non-nul.
2. $(A - \lambda I)x = 0$ doit avoir des solutions non-triviales.
3. $(A - \lambda I)$ n'est pas inversible.
4. $det(A - \lambda I)x = 0$ (l'équation caractéristique).

Il faut résoudre $det(A - \lambda I)x = 0$ pour trouver les valeurs propres.

Les valeurs propres

Exemple

Calculer les valeurs propres des matrices suivantes :

$$A_1 = \begin{pmatrix} -1 & 2 \\ 1 & -2 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$$

$$A_3 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{pmatrix} \quad A_4 = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 2 & 3 \\ 0 & 0 & -1 \end{pmatrix}$$

Normes vectorielles

Soit E un espace vectoriel sur $\mathbb{K} = \mathbb{R}$ ou $\mathbb{C}$.

Définition

On appelle norme sur E une application $||.|| : E \to \mathbb{R}_+$ telle que :

1. $\forall x \in E, (||x|| = 0 \to x = 0)$;
2. $\forall \lambda \in \mathbb{K}, \forall x \in E, ||\lambda x|| = |\lambda| ||x||$;
3. $\forall (x, y) \in E^2, ||x + y|| \leq ||x|| + ||x||$.

Normes classiques sur $\mathbb{R}^n$: $||.||_1$, $||.||_2$ et $||.||_\infty$ définies par :

$$||x||_1 = \sum_{i=1}^{n} |x_i|, \quad ||x||_2 = \left(\sum_{i=1}^{n} |x_i|^2 \right)^{\frac{1}{2}}, \quad ||x||_\infty = \max_{1 \leq i \leq n} |x_i|.$$

Normes matricielles et normes subordonnées

Définition

Une norme $||.||$ sur $\mathcal{M}_n(\mathbb{K})$ est une norme matricielle si elle vérifie :
$\forall (A, B) \in \mathcal{M}_n(\mathbb{K})^2$, $||AB|| \leq ||A||||B||$.

Définition

Soit $||.||$ une norme vectorielle sur $\mathbb{K}^n$. Pour toute matrice $A \in \mathcal{M}_n(\mathbb{K})$, on définit $||.||_M : \mathcal{M}_n(\mathbb{K}) \to \mathbb{R}_+$ par

$$||A||_M = \sup_{x \in \mathbb{K}^n - \{0\}} \frac{||Ax||}{||x||}.$$

Alors, $||.||_M$ est une norme matricielle. Elle est dite norme subordonnée à la norme vectorielle $||.||$.

Normes subordonnées classiques

1. Normes subordonnées associées aux normes vectorielles $||.||_1$, $||.||_2$ et $||.||_\infty$ de $\mathbb{R}^n$: $\forall A = (a_{ij})_{1\leq i,j\leq n} \in \mathcal{M}_n(\mathbb{K})$:

$$||A||_1 = \max_{1\leq j\leq n} \sum_{i=1}^{n} |a_{ij}|,$$

$$||A||_2 = \sqrt{\rho(AA^*)},$$

$$||A||_\infty = \max_{1\leq i\leq n} \sum_{j=1}^{n} |a_{ij}|,$$

où :

1. $A^* = \overline{A}^T$ désigne la matrice adjointe de A.
2. $\rho(M)$ désigne le rayon spectral d'une matrice M, c-à-d le maximum des modules des valeurs propres de M.

Conditionnement d'une matrice : exemple

Considérons le système linéaire $(S) : Ax = b$ avec

$$A = \begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix}, \quad b = \begin{pmatrix} 32 \\ 23 \\ 33 \\ 31 \end{pmatrix}$$

On remarque que :

1. A est symétrique.
2. $det(A) = 1$.
3. la solution de (S) est donnée par $x = (1, 1, 1, 1)^T$.

Premier cas : b est perturbé

1. Perturbons légèrement le second membre b et considérons

$$b^{'} = \begin{pmatrix} 32,1 \\ 23,9 \\ 33,1 \\ 31,9 \end{pmatrix}.$$

2. Si on résout le système $(S^{'})$: $Ax^{'} = b^{'}$, on trouve $x^{'} = (9.2, -12.6, 4.5, -1.1)^{T}$.
3. La petite perturbation sur le second membre b entraine donc une forte perturbation sur la solution du système.
4. D'une manière générale, pour $Ax = b$ et $A(x + \delta x) = b + \delta b$:

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

Deuxième cas : A est perturbée

1. Si on perturbe légèrement la matrice A :

$$A'' = \begin{pmatrix} 10 & 7 & 8.1 & 7.2 \\ 7.08 & 5.04 & 6 & 5 \\ 8 & 5.98 & 9.89 & 9 \\ 6.99 & 4.99 & 9 & 9.89 \end{pmatrix}.$$

2. Si on résout le système (S'') : $A''x'' = b$, on trouve $x'' = (-81, 107, -34, 22)^T$.
3. D'une manière générale, pour $Ax = b$ et $(A + \Delta A)(x + \delta x) = b$:

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

Conditionnement : définition

Définition

Soit $||.||$ une norme matricielle subordonnée et A une matrice inversible. Le nombre $Cond(A) = ||A^{-1}||||A||$ s'appelle le conditionnement de A relatif à la norme $||.||$.

1. Ce nombre mesure la sensibilité de la solution par rapport aux données du problème.

2. une matrice est :
   1. bien conditionnée si $Cond(A) \approx 1$,
   2. mal conditionnée si $Cond(A) >> 1$.

Estimation théorique de l'erreur a posteriori

1. $Ax = b$? ? erreur commise sur la solution réellement calculée !
2. x la solution exacte, y la solution obtenue, $r = Ay - b$ (résidu).

Théorème

$$||y - x|| \leq Cond(A).\frac{||r||}{||b||}.||x||.$$

3. Conditionnement est grand $\Rightarrow$ erreur relative peut être grande.
4. Difficile à utiliser car en général conditionnement inconnu !
5. C approximation de A^{-1} (Par Gauss), $R = AC - I_n$.

Théorème

$$Si \ ||R|| < 1, \quad alors \ ||y - x|| \leq \frac{||r||||C||}{1 - ||R||}.$$

Chapitre 4
Résolution d'un système d'équations linéaires : méthodes itératives

Modèle général d'un schéma itératif (1)

1. $A \in \mathcal{M}_n(\mathbb{K})$, $b \in \mathbb{K}^n$ et $(S) : Ax = b$.
2. Principe général : générer une suite de vecteurs qui converge vers la solution $A^{-1}b$.
3. Idée : écrire (S) sous une forme équivalente permettant de voir la solution comme un point fixe :

$$(S) \Leftrightarrow Bx + c = x,$$

   $B \in \mathcal{M}_n(\mathbb{K})$ et $c \in \mathbb{K}^n$ bien choisis, c-à-d $I - B$ inversible et $c = (I - B)A^{-1}b$.
4. Exemple : $A = M - N$ (M inversible), $B = M^{-1}N$ et $c = M^{-1}b$.

Modèle général d'un schéma itératif (2)

1. On se donne alors $x^{(0)} \in \mathbb{K}^n$ et on construit une suite de vecteurs $x^{(k)} \in \mathbb{K}^n$ à l'aide du schéma itératif

$$(*) \quad x^{(k+1)} = Bx^{(k)} + c, \qquad k = 1, 2, ...$$

2. Si $(x^{(k)})_{k\in\mathbb{N}}$ est convergente, alors elle converge vers la solution $A^{-1}b$ de (S).

Convergence (1)

Définition

Une méthode itérative définie par $(x^{(k)})_{k\in\mathbb{N}}$ pour résoudre un système $Ax = b$ est dite convergente si pour toute valeur initiale $(x^{(0)})_{k\in\mathbb{N}}$, on a $\lim\limits_{k\to+\infty} x^{(k)} = A^{-1}b$.

Lemme

Si la méthode itérative est convergente et si on note $x = A^{-1}b$ la solution, alors

$$x^{(k)} - x = B^k(x^{(0)} - x).$$

Remarque

$x^{(k)} - x$ erreur à la k-ème itération $\Rightarrow$ estimation de cette erreur en fonction de l'erreur initiale.

Convergence (2)

Théorème

Les assertions suivantes sont équivalentes :

1. $x^{(k+1)} = Bx^{(k)} + c$ *est convergente ;*
2. *pour tout* $y \in \mathbb{K}^n$, $\lim_{k \to +\infty} B^k y = 0$;
3. *pour toute norme matricielle* $||.||$ *sur* $\mathcal{M}_n(\mathbb{K})$, *on a*

$$\lim_{k \to +\infty} ||B^k|| = 0.$$

Théorème

Les assertions suivantes sont équivalentes :

1. $x^{(k+1)} = Bx^{(k)} + c$ *est convergente ;*
2. $\rho(B) < 1$, *où* $\rho(B)$ *désigne le rayon spectral de la matrice* B ;
3. *il existe une norme matricielle* $||.||$ *sur* $\mathcal{M}_n(\mathbb{K})$ *subordonnée à une norme vectorielle* $||.||$ *sur* $\mathbb{K}^n$ *telle que* $||B|| < 1$.

Vitesse de convergence (1)

Définition

Considérons un schéma itératif $x^{(k+1)} = Bx^{(k)} + c$ convergent. Soit $||.||$ une norme matricielle sur $\mathcal{M}_n(\mathbb{K})$ et soit k un entier tel que $||B^k|| < 1$. On appelle taux moyen de convergence associé à la norme $||.||$ pour k itérations de $x^{(k+1)} = Bx^{(k)} + c$ le nombre positif

$$R_k(B) = -\ln\left(\left(||B^k||\right)^{\frac{1}{k}}\right).$$

Définition

Considérons deux méthodes itératives convergentes

1. $x^{(k+1)} = B_1 x^{(k)} + c_1, \qquad k = 1, 2, ...,$
2. $x^{(k+1)} = B_2 x^{(k)} + c_2, \qquad k = 1, 2, ...$

Soit k un entier tel que $||B_1^k|| < 1$ et $||B_2^k|| < 1$. On dit que (1) est plus rapide que (2) relativement à la norme $||.||$ si $R_k(B_1) \geq R_k(B_1)$.

Vitesse de convergence (2)

Définition

On appelle taux asymptotique de convergence le nombre

$$R_\infty(B) = \lim_{k \to +\infty} R_k(B) = -\ln(\rho(B)).$$

Théorème

Une méthode itérative est d'autant plus rapide que son taux asymptotique de convergence est grand c-à-d que $\rho(B)$ est petit.

Les méthodes itératives classiques

1. $(S) : Ax = b$ avec A inversible.
2. Idée : déduire un schéma itératif d'une décomposition $A = M - N$, tel que M inversible.
3. En pratique, on suppose que les systèmes de matrice M sont faciles à résoudre (par ex. M diagonale, triangulaire, ...).
4. (S) s'écrit alors $Mx = Nx + b$ c-à-d $x = Bx + c$ avec $B = M^{-1}N$, $c = M^{-1}b$ et on considère le schéma itératif associé :

$$x^{(0)} \in \mathbb{K}^n, \quad Mx^{(k+1)} = Nx^{(k)} + b.$$

5. On montre alors que $I - B$ inversible et $c = (I - B)A^{-1}b$.

Trois exemples classiques (1)

1. Dans ce cours, 3 exemples classiques : les méthodes de Jacobi, Gauss-Seidel et de relaxation.

2. Point de départ : décomposition de $A = (a_{ij})_{1 \leq i,j \leq n}$ sous la forme $A = D - E - F$ avec :
   1. $D = (d_{ij})_{1 \leq i,j \leq n}$ diagonale, telle que $d_{ii} = a_{ii}$ et $d_{ij} = 0$ pour $i \neq j$;
   2. $E = (e_{ij})_{1 \leq i,j \leq n}$ triangulaire inférieure stricte telle que $e_{ij} = -a_{ij}$ si $i > j$ et $e_{ij} = 0$ si $i \leq j$;
   3. $F = (f_{ij})_{1 \leq i,j \leq n}$ triangulaire supérieure stricte telle que $f_{ij} = -a_{ij}$ si $i < j$ et $f_{ij} = 0$ si $i \geq j$.

Exemple de décomposition $A = D - E - F$

$$\underbrace{\begin{pmatrix} 2 & -1 & 1 \\ 2 & 2 & 2 \\ -1 & -1 & 2 \end{pmatrix}}_{A} = \underbrace{\begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix}}_{D} - \underbrace{\begin{pmatrix} 0 & 0 & 0 \\ -2 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}}_{E} - \underbrace{\begin{pmatrix} 0 & 1 & -1 \\ 0 & 0 & -2 \\ 0 & 0 & 0 \end{pmatrix}}_{F}$$

Trois exemples classiques (2)

On suppose D une matrice inversible.

1. Méthode de Jacobi : $M = D$, $N = E + F$.
2. Méthode de Gauss-Seidel : $M = D - E$, $N = F$.
3. Méthode de relaxation : $M = \frac{1}{\omega}(D - \omega E)$, $N = \left(\frac{1-\omega}{\omega}\right) D + F$, avec ω paramètre réel non nul.

Remarque

Gauss-Seidel est un cas particulier de relaxation pour $\omega = 1$.

Méthode de Jacobi : description

1. (S) : $Ax = b$ avec A inversible.
2. $A = M - N$ avec $M = D$ inversible et $N = E + F$.
3. Le schéma itératif s'écrit alors

$$Dx^{(k+1)} = (E + F)x^{(k)} + b \Leftrightarrow x^{(k+1)} = D^{-1}(E + F)x^{(k)} + D^{-1}b.$$

Définition

La matrice $B_j = D^{-1}(E + F)$ s'appelle la matrice de Jacobi associé à A.

Jacobi : mise en œuvre

1. on a $Dx^{(k+1)} = (E+F)x^{(k)} + b$, donc pour tout $i = 1, ..., n$, $(Dx^{(k+1)})_i = ((E+F)x^{(k)})_i + b_i$ c-à-d

$$a_{ii}x_i^{(k+1)} = -\sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij}x_j^{(k)} + b_i$$

$$\Leftrightarrow x_i^{(k+1)} = \frac{1}{a_{ii}}\left(-\sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij}x_j^{(k)} + b_i \right).$$

Jacobi : convergence et exemple

Théorème

La méthode de Jacobi converge si et seulement si $\rho(B_j) < 1$.

Exemple

Pour la matrice $A = \begin{pmatrix} 2 & -1 & 1 \\ 2 & 2 & 2 \\ -1 & -1 & 2 \end{pmatrix}$ précédente :

$$B_J = \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{2} \end{pmatrix} \begin{pmatrix} 0 & 1 & -1 \\ -2 & 0 & -2 \\ 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & \frac{1}{2} & -\frac{1}{2} \\ -1 & 0 & -1 \\ \frac{1}{2} & \frac{1}{2} & 0 \end{pmatrix}.$$

Les valeurs propres : 0 et $\frac{\pm\sqrt{5}}{2}$ donc $\rho(B_J) = \frac{\sqrt{5}}{2} > 1$ et la méthode de Jacobi diverge.

Méthode de Gauss-Seidel : description

1. (S) : $Ax = b$ avec A inversible.
2. $A = M - N$ avec $M = D - E$ inversible et $N = F$.
3. Le schéma itératif s'écrit alors

$$(D - E)x^{(k+1)} = Fx^{(k)} + b \Leftrightarrow x^{(k+1)} = (D - E)^{-1}Fx^{(k)} + (D - E)^{-1}b.$$

Définition

La matrice $B_{GS} = (D - E)^{-1}F$ s'appelle la matrice de Gauss-Seidel associée à A.

Gauss-Seidel : mise en œuvre

1. On a $(D-E)x^{(k+1)} = Fx^{(k)} + b$ dons pour tout $i = 1, ..., n$
$((D-E)x^{(k+1)})_i = (Fx^{(k)})_i + b_i$ c-à-d

$$a_{ii}x_i^{(k+1)} + \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} = -\sum_{j=i+1}^{n} a_{ij}x_j^{(k)} + b_i,$$

ce qui entraine

$$x_1^{(k+1)} = \frac{1}{a_{11}}\left(-\sum_{j=2}^{n} a_{1j}x_j^{(k)} + b_1 \right),$$

et pour $i = 2, ..., n$

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

Gauss-Seidel : convergence et exemple

Théorème

La méthode de Gauss-Seidel converge si et seulement si $\rho(B_{GS}) < 1$.

Exemple

Pour la matrice $A = \begin{pmatrix} 2 & -1 & 1 \\ 2 & 2 & 2 \\ -1 & -1 & 2 \end{pmatrix}$ précédente :

$$B_{GS} = \begin{pmatrix} 2 & 0 & 0 \\ 2 & 2 & 0 \\ -1 & -1 & 2 \end{pmatrix}^{-1} \begin{pmatrix} 0 & 1 & -1 \\ 0 & 0 & -2 \\ 0 & 0 & 0 \end{pmatrix},$$

$$B_{GS} = \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ -\frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{4} & \frac{1}{2} \end{pmatrix} \begin{pmatrix} 0 & 1 & -1 \\ 0 & 0 & -2 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & \frac{1}{2} & -\frac{1}{2} \\ 0 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & 0 & -\frac{1}{2} \end{pmatrix}.$$

Les valeurs propres : 0 et $-\frac{1}{2}$ donc $\rho(B_{GS}) = \frac{1}{2} < 1$ et Gauss-Seidel converge.

Méthode la relaxation : description

1. (S) : $Ax = b$ avec A inversible.
2. Soit ω un paramètre réel non nul. On pose $A = M - N$ avec $M = \frac{1}{\omega}(D - \omega E)$ inversible et $N = \left(\frac{1-\omega}{\omega}\right)D + F$.
3. Le schéma itératif s'écrit alors

$$\frac{1}{\omega}(D - \omega E)x^{(k+1)} = \left(\left(\frac{1-\omega}{\omega}\right)D + F\right)x^{(k)} + b,$$

$$x^{(k+1)} = (D - \omega E)^{-1}\Big((1-\omega)D + \omega F\Big)x^{(k)} + \omega(D - \omega E)^{-1}b.$$

Définition

La matrice $B_R(\omega) = (D - \omega E)^{-1}\Big((1-\omega)D + \omega F\Big)$ s'appelle la matrice de relaxation associée à A et ω est le facteur de relaxation.

Méthode de la relaxation : convergence et exemple

Théorème

La méthode de relaxation converge si et seulement si $\rho(B_R(\omega)) < 1$.

Exemple

Pour la matrice $A = \begin{pmatrix} 2 & -1 & 1 \\ 2 & 2 & 2 \\ -1 & -1 & 2 \end{pmatrix}$ précédente :

$$B_R(\omega) = \begin{pmatrix} 1-\omega & \frac{1}{2}\omega & -\frac{1}{2}\omega \\ \omega(\omega-1) & -\frac{1}{2}\omega^2 + 1 - \omega & \frac{1}{2}\omega^2 - \omega \\ \frac{1}{2}\omega(\omega-1)^2 & -\frac{1}{4}\omega^3 - \frac{1}{4}\omega^2 + \frac{1}{2}\omega & \frac{1}{4}\omega^3 - \frac{3}{4}\omega^2 + 1 - \omega \end{pmatrix}.$$

Les valeurs propres et la convergence dépendent de ω.

Cas particulier : matrice symétrique définie positive

Définition

Soit $A \in \mathcal{M}_n(\mathbb{R})$ symétrique. On dit que A est définie positive si pour tout $x \in \mathbb{R}^n$ non nul, on a $\langle Ax, x\rangle = x^T Ax > 0$.

Théorème

Soit A une matrice symétrique définie positive et écrivons $A = M - N$ avec M inversible et $M^T + N$ définie positive. Alors la méthode itérative

$$x^{(0)} \in \mathbb{K}^n, \quad x^{(k+1)} = M^{-1}Nx^{(k)} + M^{-1}b,$$

converge.

Corollaire

Soit A une matrice symétrique définie positive. Alors la méthode de Gauss-Seidel converge.

Cas particulier : matrice à diagonale strictement dominante

Définition

Une matrice $A = (a_{ij})_{1 \leq i,j \leq n}$ est dite à diagonale strictement dominante si :

$$\forall i = 1, ...n, \quad |a_{ii}| > \sum_{j=1,\ j \neq i} |a_{ij}|.$$

Théorème

Soit A une matrice à diagonale strictement dominante. Alors A est inversible et les méthodes de Jacobi et de Gauss-Seidel convergent toutes les deux.

Remarque

1. Définition équivalente : une matrice réelle symétrique A est définie positive si toutes ses valeurs propres sont positives.

Méthode alternative : le gradient conjugué (Paragraphe supprimé)

1. Solution des systèmes linéaires $(S) : Ax = b$ avec $A \in \mathcal{M}_n(\mathbb{R})$ symétrique et définie positive.
2. Un produit matrice $\times$ vecteur à chaque itération $\Leftarrow$ méthode bien adaptée aux systèmes creux et de grande taille.
3. La méthode construit une suite de $(x^{(k)})_{k=0,1,\ldots}$ telle que $x^{(m)} = A^{-1}b$ pour un indice $m \leq n \Leftarrow$ méthode exacte en principe mais considérée comme une méthode itérative à cause des erreurs numériques.
4. Dans les applications, le nombre d'itérations nécessaires est significativement plus petit que la taille du système, en particulier si on utilise des techniques de pré conditionnement.

Un problème d'optimisation

On considère le problème suivant : minimiser la fonction

$$\phi(x) = \frac{1}{2}x^T Ax - b^T x,$$

où la matrice A est symétrique et définie positive.

1. Le minimum de ϕ est atteint pour $x^* = A^{-1}b$, et cette solution est unique.
2. Minimiser $\phi(x)$ et résoudre $Ax = b$ sont deux problèmes équivalents.

Méthodes du gradient

Définition

Le gradient de ϕ en $x = (x_1, ..., x_n)^T \in \mathbb{R}^n$ est le vecteur

$$\nabla\phi(x) = \left(\frac{\partial\phi}{\partial x_1}, \frac{\partial\phi}{\partial x_2}, ..., \frac{\partial\phi}{\partial x_n}\right).$$

On a

$$\nabla\phi(x) = \frac{1}{2}Ax + \frac{1}{2}A^T x - b = Ax - b.$$

Définition

La quantité $r(x) = b - Ax = -\nabla\phi(x)$ est appelée résidu du système (S) en x.

On rappelle que $-\nabla\phi(x)$ donne la direction de plus forte pente pour $\phi(x)$ au point x.

Méthodes du gradient

A l'étape k d'une méthode du gradient :

1. on choisit une direction de descente pour $\phi(x)$, i.e. un vecteur $p^{(k)}$ tel que $p^{(k)}\nabla\phi(x^{(k)}) < 0$.
2. on minimise $\phi(x)$ sur la droite passant par $x^{(k)}$ et de vecteur directeur $p^{(k)}$

$$x^{(k+1)} = x^{(k)} + \alpha_k p^{(k)},$$

où $\alpha_k = \frac{f^{(k)T} p^{(k)}}{p^{(k)T} A p^{(k)}}$.

Proposition

A chaque itération, le résidu $r^{(k+1)}$ est orthogonal au vecteur $p^{(k)}$ utilisé à l'étape précédente : $r^{(k+1)T} p^{(k)} = 0$.

Gradient conjugué

1. Choix de la direction de descente :

$$p^{(k)} = \begin{cases} r^{(0)} & \text{si } k = 0, \\ r^{(k)} + \beta_k p^{(k-1)} & \text{si } k \geq 1, \end{cases}$$

où β_k est calculé tel que

$$p^{(k)T} A p^{(k-1)} = 0.$$

2. Les vecteurs direction $p^{(k-1)}$ et $p^{(k)}$ sont A-conjugués.
3. On a $\beta_k = -\frac{r^{(k)T} A p^{(k-1)}}{p^{(k-1)T} A p^{(k-1)}} = \frac{r^{(k)T} r^{(k)}}{r^{(k-1)T} r^{(k-1)}}$ et $\alpha_k = \frac{r^{(k)T} r^{(k)}}{p^{(k)T} A p^{(k)}}$.

Gradient conjugué

Lemme

A chaque itération, le résidu est orthogonal au résidu calculé à l'itération précédente : $r^{(k)T} r^{(k-1)} = 0$.

Théorème

Soit $\mathcal{S}_k$ *le sous-espace vectoriel de* $\mathbb{R}^n$ *engendré par les vecteurs* $p^{(0)}, ..., p^{(k-1)}$. *Alors le vecteur* $x^{(k)}$ *défini par la méthode du gradient conjugué minimise* $\phi(x)$ *sur* $\mathcal{S}_k$ *:*

$$\phi(x^{(k)}) = \min_{x \in \mathcal{S}_k} \phi(x), \qquad k \geq 1.$$

Gradient conjugué : l'algorithme

Entrée : $A \in \mathcal{M}_n(\mathbb{R})$ symétrique et définie positive, $b \in \mathbb{R}^n$ and $x^{(0)} \in R^n$.
Sortie : $x \in R^n$ tel que $Ax = b$.

1. $k = 0$;
2. $r^{(0)} = b - Ax^{(0)}$;
3. Tant que $r^{(k)} \neq 0$, faire :
   - Si $k = 0$ alors faire :
     $\beta_0 = 0$;
     $p^{(0)} = r^{(0)}$;
   - sinon faire :
     $\beta_k = \frac{r^{(k)T} r^{(k)}}{r^{(k-1)T} r^{(k-1)}}$;
     $p^{(k)} = r^{(k)} + \beta_k p^{(k-1)}$;
   - $\alpha_k = \frac{r^{(k)T} r^{(k)}}{p^{(k)T} A p^{(k)}}$;
   - $x^{(k+1)} = x^{(k)} + \alpha_k p^{(k)}$;
   - $r^{(k+1)} = r^{(k)} - \alpha_k A p^{(k)}$;
   - $k = k + 1$.
4. Retourner $x = x^{(k)}$.