

L3 Informatique

Cours Programmation Scientifique

Cyrille Bertelle et Mustapha Arfi
LITIS
UFR Sciences et Techniques Le Havre
Normandie Université - Le Havre

January 16, 2018

Contents

1	Présentation et introduction	3
2	Méthodes directes de résolution de systèmes linéaires	4
2.1	Introduction	4
2.2	Formule de Cramer	5
2.3	Méthode de Gauss	5
2.3.1	Triangulation du système	5
2.3.2	Résolution du système triangulaire	8
2.4	Méthode du pivot de Gauss	8
2.5	Place mémoire occupée	10
2.6	Méthode de factorisation LDR	11
2.6.1	Résolution du système linéaire avec la factorisation LDR	13
2.6.2	Intérêt de la méthode de factorisation	13
2.6.3	Calcul de l'inverse d'une matrice	14
2.7	Conditionnement d'une matrice	14
3	Méthodes itératives de résolution de systèmes linéaires	16
3.1	Introduction	16
3.2	Une étude abstraite générale	17
3.3	Étude de cas particuliers de décomposition	
	$A = M - N$	19
3.3.1	Méthode de Jacobi	19
3.3.2	Méthode de Gauss-Seidel	20
3.3.3	Méthode de relaxation	21
3.3.4	Exemples	22
3.3.5	Résultats sur la convergence des méthodes itératives	23
4	Interpolation polynomiale et splines cubiques	24
4.1	Introduction	24
4.2	Interpolation polynômiale	25
4.2.1	Méthode de Lagrange	26
4.2.2	Méthode de Newton	28
4.2.3	Limite de l'interpolation polynômiale	30

4.3	Méthodes de Splines Cubiques	
	Interpolation polynomiale par morceaux	33
4.3.1	Description	33
4.3.2	Algorithme	36
4.4	Interpolation bidimensionnelle	38
5	Approximation aux moindres carrés	43
5.1	Principe d'approximation et critère de moindres carrés	43
5.2	Régression linéaire	44
5.3	Généralisation aux modèles linéaires	46
5.3.1	Ecriture matricielle du problème	47
6	Dérivation et intégration numérique	48
6.1	Dérivation numérique	48
6.1.1	Approximation de la dérivée première	49
6.1.2	Approximation de la dérivée seconde	50
6.2	Intégration numérique	51
6.2.1	La méthode des trapèzes	51
6.2.2	La méthode de Simpson	53
6.2.3	Méthode de Newton-Cote	55
6.2.4	Méthode de Gauss	56
6.2.5	Intégrales multiples (introduction)	60
7	Résolution d'équations/systèmes différentiels	61
8	Annexe	
	Rappels et compléments d'algèbre linéaire	62
8.1	Quelques structures algébriques	62
8.2	Espace vectoriel	64
8.3	Applications linéaires et matrices	64
8.4	Matrice inverse d'une matrice carrée	66
8.5	Calcul du déterminant d'une matrice carrée	66
8.6	Trace d'une matrice carrée	67
8.7	Matrices transposées, orthogonales, hermitiennes, unitaires et normales	67
8.8	Valeurs propres et vecteurs propres de matrice	68
8.9	Matrices hermitiennes définies positives et matrices à diagonale dominante	69
8.10	Normes vectorielles et matricielles	70

Chapter 1

Présentation et introduction

Chapter 2

Méthodes directes de résolution de systèmes linéaires

Sommaire

2.1	Introduction	4
2.2	Formule de Cramer	5
2.3	Méthode de Gauss	5
2.3.1	Triangulation du système	5
2.3.2	Résolution du système triangulaire	8
2.4	Méthode du pivot de Gauss	8
2.5	Place mémoire occupée	10
2.6	Méthode de factorisation LDR	11
2.6.1	Résolution du système linéaire avec la factorisation LDR	13
2.6.2	Intérêt de la méthode de factorisation	13
2.6.3	Calcul de l'inverse d'une matrice	14
2.7	Conditionnement d'une matrice	14

2.1 Introduction

Soient $A \in M_{n,n}(\mathbb{R})$, $x, b \in \mathbb{R}^n$ et l'équation correspondant au système linéaire

$$Ax = b$$

On cherche le vecteur x qui vérifie ce système linéaire avec comme données à ce problème, la matrice A et le vecteur b .

Remarque 2.1. *Le système a une solution unique si et seulement si $\det A \neq 0$.*

Il existe deux familles de méthodes :

- les méthodes directes : algorithmes fournissant en un temps fini la solution exacte.
- les méthodes itératives : calcul d'une suite de vecteurs qui converge vers la solution.

2.2 Formule de Cramer

Un résultat connu en algèbre sous le nom de formule de Cramer, permet d'obtenir directement toutes les composantes du vecteur x grâce à l'expression suivante :

$$x_i = \frac{\det(A_i)}{\det(A)}$$

où A_i est la matrice obtenue à i^{e} partir de A en remplaçant la i^{e} colonne par le vecteur second membre b .

Nombre d'opérations nécessaires : on doit calculer n déterminants dont le coût/nombre d'opérations est en $O(n!)$, soit un coût total en $O(n.n!)$.

Pour $n = 15$, coût d'environ 10^{13} opérations. Sur une machine qui effectue 10^6 opérations par seconde, cela prendrait 4 mois de calcul.

En conclusion, cette formule est inutilisable sauf pour des valeurs petites de n , inférieures à 10.

2.3 Méthode de Gauss

On procède en deux étapes :

- On transforme le système en un système triangulaire supérieur équivalent,
- On résout le système triangulaire.

2.3.1 Triangulation du système

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 & (1) \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 & (2) \\ \vdots & \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n & (n) \end{cases} \quad (2.1)$$

1^{re} étape : on transforme le système en un système équivalent en effectuant des combinaisons linéaires d'équations afin d'éliminer les coefficients de x_1 de l'équation

(2) à l'équation (n).

Par exemple, pour la ligne (i), $2 \leq i \leq n$, on la remplace par la combinaison linéaire

$$(i)^{(1)} \leftarrow (i) - \frac{a_{i1}}{a_{11}}(1) \quad \text{si } a_{11} \neq 0$$

Remarque 2.2. L'indice supérieur (1) désigne les coefficients obtenus à l'issue de l'étape 1.

C'est à dire que pour chaque coefficient a_{ij} de la ligne (i) on devra le remplacer par :

$$a_{ij}^{(1)} \leftarrow a_{ij} - \frac{a_{i1}}{a_{11}}a_{1j} \quad \text{pour } 2 \leq j \leq n$$

Et pour le second membre b_i , on devra le remplacer par :

$$b_i^{(1)} \leftarrow b_i - \frac{a_{i1}}{a_{11}}b_1$$

Ces formules doivent alors être appliquées à toutes les lignes (i) pour $2 \leq i \leq n$.

2^e étape : Au début de cette étape, le système est le suivant

$$\left\{ \begin{array}{l} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \quad (1) \\ 0 + a_{22}^{(1)}x_2 + \dots + a_{2n}^{(1)}x_n = b_2^{(1)} \quad (2) \\ \vdots \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \vdots \\ 0 + a_{i2}^{(1)}x_2 + \dots + a_{in}^{(1)}x_n = b_i^{(1)} \quad (i) \\ \vdots \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \vdots \\ 0 + a_{n2}^{(1)}x_2 + \dots + a_{nn}^{(1)}x_n = b_n^{(1)} \quad (n) \end{array} \right.$$

On doit maintenant éliminer les coefficients de la deuxième colonne, c'est à dire ceux qui sont en dessous du coefficient $a_{22}^{(1)}$, en effectuant des combinaisons linéaires avec la ligne (2) et chaque ligne (i); $3 \leq i \leq n$:

$$(i)^{(2)} \leftarrow (i)^{(1)} - \frac{a_{i2}^{(1)}}{a_{22}^{(1)}}(2) \quad \text{si } a_{22}^{(1)} \neq 0$$

Remarque 2.3. L'indice supérieur (2) désigne les coefficients obtenus à l'issue de l'étape 2.

k^e étape : au début de cette étape, la matrice du système équivalent au système initial correspond à

$$\begin{pmatrix} a_{11} & a_{12} & \dots & \dots & \dots & a_{1n} \\ 0 & a_{22}^{(1)} & \dots & \dots & \dots & a_{1n}^{(1)} \\ \vdots & \ddots & \ddots & & & \vdots \\ \vdots & & 0 & a_{kk}^{(k-1)} & \dots & a_{kn}^{(k-1)} \\ \vdots & & \vdots & a_{k+1k}^{(k-1)} & \dots & a_{k+1n}^{(k-1)} \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \dots & 0 & a_{nk}^{(k-1)} & \dots & a_{nn}^{(k-1)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \\ x_{k+1} \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \\ b_{k+1} \\ \vdots \\ b_n \end{pmatrix}$$

On fait alors, à cette étape, des combinaisons avec la ligne (k) et toutes les suivantes de façon à ce qu'après la transformation, on ait supprimé tous les coefficients sous le coefficient $a_{kk}^{(k-1)}$, ce qui revient à dire que l'on veut obtenir que $a_{ik}^{(k)} = 0$ pour $k + 1 \leq i \leq n$.

Pour chacune de ces lignes (i), on doit donc faire la combinaison linéaire :

$$(i)^{(k)} \leftarrow (i)^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} (k) \quad \text{si } a_{kk}^{(k-1)} \neq 0$$

La combinaison linéaire appliquée à tous les éléments de la ligne (i) :

$$\begin{aligned} a_{ij}^{(k)} &\leftarrow a_{ij}^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} a_{kj}^{(k-1)} & k + 1 \leq j \leq n \\ b_i^{(k)} &\leftarrow b_i^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} b_k^{(k-1)} \end{aligned}$$

Algorithme :

```

for  $k = 1$  à  $\lfloor \frac{1}{2} n - 1 \rfloor$  do
  {traitement pour savoir si  $a_{kk}^{(k-1)} \neq 0$ }
  for  $i = k + 1$  à  $\lfloor \frac{1}{2} n \rfloor$  do
     $C \leftarrow a_{ik}^{(k-1)} / a_{kk}^{(k-1)}$ 
    for  $j = k + 1$  à  $\lfloor \frac{1}{2} n \rfloor$  do
       $a_{ij}^{(k)} \leftarrow a_{ij}^{(k-1)} - C * a_{kj}^{(k-1)}$ 
    end for
     $b_i^{(k)} \leftarrow b_i^{(k-1)} - C * b_k^{(k-1)}$ 
  end for
end for

```


2.3.2 Résolution du système triangulaire

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ 0 & a_{22} & \dots & a_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

On résout par remontée :

L'équation (n) donne

$$x_n = b_n/a_{nn} \quad \text{si } a_{nn} \neq 0$$

L'équation ($n - 1$) donne

$$a_{n-1 \ n-1}x_{n-1} + a_{n-1 \ n}x_n = b_{n-1}$$

C'est à dire, en isolant x_{n-1} à gauche de l'égalité :

$$x_{n-1} = (b_{n-1} - a_{n-1 \ n}x_n)/a_{n-1 \ n-1} \quad \text{si } a_{n-1 \ n-1} \neq 0$$

L'équation (i) donne

$$a_{ii}x_i + a_{i \ i+1}x_{i+1} + \dots + a_{in}x_n = b_i$$

c'est à dire

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

ce qui donne

$$x_i = \frac{b_i - \sum_{j=i+1}^n a_{ij}x_j}{a_{ii}} \quad \text{si } a_{ii} \neq 0$$

Algorithme

```

for  $i = n$  à  $\frac{1}{2}$  do
     $x_i \leftarrow \left( b_i - \sum_{j=i+1}^n a_{ij}x_j \right) / a_{ii}$ 
end for
    
```

2.4 Méthode du pivot de Gauss

Problème : l'algorithme ne fonctionne pas si l'un des dénominateurs utilisé précédemment est nul, c'est à dire si

$$a_{11} = 0 \text{ ou } a_{22}^{(1)} = 0, a_{kk}^{(k-1)} = 0, \dots$$

Si ce n'est pas le cas, alors il existe un k tel que $a_{kk}^{(k-1)} = 0$.

On recherche alors parmi les coefficients $\{a_{ik}; k+1 \leq i \leq n\}$ un coefficient non nul (voir la figure xxx). Supposons, par exemple, que ce coefficient non nul soit sur la ligne j .

Insérer ici la figure xxx.

En permutant les lignes k et j , on a résolu le problème.

Par contre, si tous les coefficients sont nuls, alors on sait que

$$\det A = a_{11} \times a_{22} \times \cdots \times a_{k-1,k-1} \times \begin{vmatrix} a_{kk} & \cdots & a_{kn} \\ \vdots & & \vdots \\ a_{nk} & \cdots & a_{nn} \end{vmatrix}$$

$$\det A = 0$$

En effet, la première colonne du dernier déterminant a des coefficients tous nuls et donc le déterminant lui-même est nul.

Le système n'a donc pas de solution et on s'arrête.

Pour éviter des singularités numériques ou des propagations d'erreurs numériques, on évitera d'utiliser un dénominateur trop petit en valeur absolue (qui pourrait d'ailleurs être en théorie un coefficient nul, aux erreurs d'arrondis près).

On recherche alors :

$$\max \{|a_{ik}|; k+1 \leq i \leq n\}$$

Soit j la ligne correspondant à $i = \frac{1}{2}$ ce coefficient, on permute les deux lignes k et j (sans oublier le second membre). Si le $\max < \varepsilon$ (considéré comme un zéro numérique), alors on considère que la matrice est singulière ($\det A = 0$).

Exemple 2.1.

$$\begin{cases} 10^{-4}x + y = 1 \\ x + y = 2 \end{cases}$$

de solution

$$x \simeq 1,00010 \dots$$

$$y \simeq 0,999909 \dots$$

On suppose que l'on travaille avec des nombres représentés en machine avec 3 digits (c'est à dire avec une mantisse de 3 chiffres).

Si on ne permute pas les lignes :

$$\begin{cases} 10^{-4}x + y = 1 \\ \left(1 - \frac{1}{10^{-4}} \times 10^{-4}\right)x + \left(1 - \frac{1}{10^{-4}} \times 1\right)y = \left(2 - \frac{1}{10^{-4}} \times 1\right) \end{cases}$$

la dernière équation avec 3 chiffres significatifs, est équivalente à

$$-10^4 y = -10^4$$

ce qui donne

$$y = 1$$

On reporte dans la première équation :

$$10^{-4}x + 1 = 1$$

C'est à dire $x = 0$! Ce qui n'est pas une solution acceptable.

Si on permute maintenant les deux lignes :

$$\begin{cases} x + y = 2 \\ 10^{-4}x + y = 1 \end{cases} \quad \begin{cases} x + y = 2 \\ \left(1 - \frac{10^{-4}}{1} \times 1\right)y = \left(1 - \frac{10^{-4}}{1} \times 2\right) \end{cases}$$

Avec 3 digits, la dernière équation équivaut à

$$y = 1$$

En reportant dans la première équation, on a

$$x + 1 = 2$$

C'est à dire $x = 1$. Ce qui est tout à fait satisfaisant, compte tenu de la précision.

2.5 Place mémoire occupée

L'algorithme de triangulation fait intervenir des coefficients de $A^{(k-1)}$ et $A^{(k)}$: on a besoin de deux matrices.

Formule centrale :

$$a_{ij}^{(k)} \leftarrow a_{ij}^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} a_{kj}^{(k-1)}, k+1 \leq i, j \leq n$$

On a besoin que d'une seule matrice. Pourquoi ? Après le calcul de $a_{ij}^{(k)}$, on ne réutilisera jamais $a_{ij}^{(k-1)}$ donc on peut remplacer.

2.6 Méthode de factorisation LDR

On factorise la matrice A du système sous la forme du produit de trois matrices :

$$A = LDR$$

où

- L est une matrice triangulaire inférieure à diagonale unité,
- D est une matrice diagonale,
- R est une matrice triangulaire supérieure à diagonale unité.

On peut montrer que la décomposition est unique.

$$\begin{aligned} A_{ij} &= \sum_{k=1}^n L_{ik}(DR)_{kj} \\ &= \sum_{k=1}^n L_{ik} \left(\sum_{l=1}^n D_{kl}R_{lj} \right) \\ &= \sum_{k=1}^n L_{ik}D_{kk}R_{kj} \end{aligned}$$

car $D_{kl} = 0$ sauf si $k = l$ puisque D est diagonale.

Si $i < j$

$$\begin{aligned} A_{ij} &= \sum_{k=1}^{i-1} L_{ik}D_{kk}R_{kj} + L_{ii}D_{ii}R_{ij} + \sum_{k=i+1}^n L_{ik}D_{kk}R_{kj} \\ &= \sum_{k=1}^{i-1} L_{ik}D_{kk}R_{kj} + L_{ii}D_{ii}R_{ij} \\ R_{ij} &= \frac{1}{D_{ii}} \left[A_{ij} - \sum_{k=1}^{i-1} L_{ik}D_{kk}R_{kj} \right] \end{aligned}$$

car $L_{ik} = 0$ quand $i < k$.

Si $i = j$

$$\begin{aligned} A_{ii} &= \sum_{k=1}^{i-1} L_{ik}D_{kk}R_{ki} + L_{ii}D_{ii}R_{ii} + \sum_{k=i+1}^n L_{ik}D_{kk}R_{ki} \\ D_{ii} &= A_{ii} - \sum_{k=1}^{i-1} L_{ik}D_{kk}R_{ki} \end{aligned}$$

car $L_{ik} = 0$ pour $i < k$, et $L_{ii} = R_{ii} = 1$.

Si $i > j$

$$A_{ij} = \sum_{k=1}^{j-1} L_{ik} D_{kk} R_{kj} + L_{ij} D_{jj} R_{jj} + \sum_{k=j+1}^n L_{ik} D_{kk} R_{kj}$$

$$L_{ij} = \frac{1}{D_{jj}} \left[A_{ij} - \sum_{k=1}^{j-1} L_{ik} D_{kk} R_{kj} \right]$$

Algorithme

```

for  $i = 1$  à  $n$  do
  for  $j = 1$  à  $i - 1$  do
     $L_{ij} \leftarrow \frac{1}{D_{jj}} \left( A_{ij} - \sum_{k=1}^{j-1} L_{ik} D_{kk} R_{kj} \right)$ 
  end for
   $D_{ii} \leftarrow A_{ii} - \sum_{k=1}^{i-1} L_{ik} D_{kk} R_{kj}$ 
  for  $j = i + 1$  à  $n$  do
     $R_{ij} \leftarrow \frac{1}{D_{ii}} \left( A_{ij} - \sum_{k=1}^{i-1} L_{ik} D_{kk} R_{kj} \right)$ 
  end for
end for

```

Est-elle calculable ? On peut regarder les indices pour s'en convaincre, pour le cas $i > j$ par exemple :

$$L_{ij} \leftarrow \underbrace{\frac{1}{D_{jj}}}_{j < i} \left(A_{ij} - \sum_{k=1}^{j-1} \underbrace{L_{ik}}_{k < j} \underbrace{D_{kk}}_{k < i} \underbrace{R_{kj}}_{k < i} \right)$$

Stockage des matrices L , D et R Dans les trois formules de calcul L_{ij} , D_{ii} et R_{ij} , seul le coefficient de A de même ordre que celui du calcul en cours est utilisé. Donc le coefficient A_{ij} une fois utilisé ne sera plus réutilisé dans la suite de la factorisation, et on peut donc l'effacer et le remplacer par le coefficient en cours de calcul.

Ainsi, une seule matrice est nécessaire pour l'algorithme, et on peut remplacer tous les L, R et D par A .

Remarque 2.4. Une méthode fréquemment rencontrée est la factorisation LU :

$$A = LU$$

où L est une matrice triangulaire inférieure à diagonale unité, et où U est une matrice triangulaire supérieure. $U = DR$ où D et R sont les matrices de la factorisation $A = LDR$.

Si A est symétrique, $A^t = A$:

$$\begin{aligned} A &= LDR \\ A^t &= (LDR)^t \\ &= R^t D^t L^t \end{aligned}$$

R^t est triangulaire inférieure à diagonale unité, L^t est triangulaire supérieure à diagonale unité et D^t est diagonale. Donc, comme $A = A^t$, $L = R^t$. Pour une matrice symétrique, on peut se passer d'un des facteurs.

En revanche, pour la méthode LU :

$$\begin{aligned} A &= LU \\ A^t &= (LU)^t \\ &= U^t L^t \end{aligned}$$

U^t est triangulaire inférieure mais pas à diagonale unité. On change alors de méthode de factorisation pour les matrices symétriques : on pose $A = LL^t$ où L est triangulaire inférieure mais pas à diagonale unité.

2.6.1 Résolution du système linéaire avec la factorisation LDR

Le système à résoudre s'écrit : $Ax = b$ avec $A = LDR$, d'où :

$$\begin{aligned} (LDR)x &= b \\ LD \underbrace{Rx}_z &= b \\ \underbrace{\quad}_y \end{aligned}$$

On résout en trois étapes :

1. On calcule y solution de $Ly = b$, avec L matrice triangulaire inférieure à diagonale unité,
2. On calcule z solution de $Dz = y$, avec D matrice diagonale,
3. On calcule x solution de $Rx = z$, avec R matrice triangulaire supérieure à diagonale unité.

Dans les TP, pour les trois appels successifs permettant de résoudre les systèmes 1,2 et 3, on transmettra la matrice A directement.

2.6.2 Intérêt de la méthode de factorisation

Dans la transformation qui correspond à la factorisation, on n'agit pas sur le second membre b , par opposition avec ce qui est fait dans la méthode de Gauss.

Si on doit résoudre plusieurs systèmes linéaires avec la même matrice et des seconds membres différents, on ne fera qu'une seule fois la factorisation LDR .

2.6.3 Calcul de l'inverse d'une matrice

Pour résoudre $AA^{-1} = \text{Id}$, on résout n systèmes linéaires :

$$\begin{aligned} Ax &= E_1 \\ Ax' &= E_2 \\ &\vdots \\ Ax^{(n-1)} &= E_n \end{aligned}$$

où E_i est la i^{e} colonne de la matrice Id . Les $x^{(i)}$ forment les colonnes de la matrice A^{-1} .

Pour la résolution de ces systèmes, la méthode de factorisation vue précédemment est bien plus efficace que la méthode de Gauss, car A est conservée.

Remarque 2.5. *Dans l'algorithme de factorisation LDR, on effectue des divisions par les coefficients de D . Que faire si l'un d'eux est nul ?*

On a un résultat qui dit que si tous les mineurs principaux de A sont non nuls, alors la matrice peut se factoriser sous la forme LDR. Le mineur d'ordre n est $\det A$, le mineur d'ordre $n-1$ est le déterminant de la matrice A privée des lignes et colonnes n, \dots . Dans la pratique, on regarde si les coefficients diagonaux de D calculés sont non nuls, et on peut dire que la méthode ne s'applique pas.

Peut-on permuter des lignes en cours de résolution, comme on le fait pour la méthode de Gauss ? Si on permute une ligne dans la matrice A , on doit mémoriser les permutations effectuées pour les appliquer ultérieurement aux seconds membres. On peut pour cela conserver un vecteur d'indices :

$$\begin{pmatrix} 1 \\ 2 \\ 3 \\ \vdots \\ n \end{pmatrix}$$

On transforme le second membre b en c , de façon à avoir $c_i \leftarrow b_{\text{permut}(i)}$. Ainsi, la méthode LDR peut s'appliquer aux mêmes matrices que la méthode de Gauss.

2.7 Conditionnement d'une matrice

Définition 2.1. *Soit $A \in M_{n,n}(K)$ inversible et $\|\cdot\|$ une norme matricielle sur $M_{n,n}(K)$. On appelle conditionnement de A relativement à une norme matricielle $\|\cdot\|$, le nombre réel positif :*

$$\text{card}(A) = \|A\| \times \|A^{-1}\|$$

Nous rappelons ci-dessous la définition de 2 normes matricielles qui pourront être utilisées pour calculer le conditionnement. On pourra se référer à l'annexe sur les rappels d'algèbre linéaire pour plus d'information sur les normes matricielles.

Définition 2.2. Norme matricielle associée à L_1 :

$$\|A\|_1 = \max_j \left(\sum_{i=1}^n |a_{ij}| \right)$$

Définition 2.3. Norme matricielle associée à L_∞ :

$$\|A\|_\infty = \max_i \left(\sum_{j=1}^m |a_{ij}| \right)$$

Exemple 2.2. Illustration de la notion de conditionnement.

Soit le système linéaire :

$$\begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} 32 \\ 23 \\ 33 \\ 31 \end{pmatrix}$$

De solution $\begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$.

On considère le système perturbé pour lequel les coefficients du second membre ont été légèrement modifiés :

$$\begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix} \begin{pmatrix} u_1 + \delta_1 \\ u_2 + \delta_2 \\ u_3 + \delta_3 \\ u_4 + \delta_4 \end{pmatrix} = \begin{pmatrix} 32,1 \\ 22,9 \\ 33,1 \\ 30,9 \end{pmatrix}$$

La solution est $\begin{pmatrix} 9,2 \\ -12,6 \\ 4,5 \\ -1,1 \end{pmatrix}$

On constate qu'une erreur relative de l'ordre de $\frac{1}{200}$ sur les données entraîne une erreur relative sur le résultat de l'ordre de 10.

On a un coefficient d'amplification entre les erreurs relatives sur les données et les erreurs relatives sur le résultat de 2000. On a le même phénomène si on introduit des perturbations sur les coefficients de la matrice.

Une matrice qui génère un tel coefficient d'amplification est dite mal conditionnée. On complètera cette définition dans le cadre d'un exercice de TD.

Chapter 3

Méthodes itératives de résolution de systèmes linéaires

Sommaire

3.1	Introduction	16
3.2	Une étude abstraite générale	17
3.3	Étude de cas particuliers de décomposition	
	$A = M - N$	19
3.3.1	Méthode de Jacobi	19
3.3.2	Méthode de Gauss-Seidel	20
3.3.3	Méthode de relaxation	21
3.3.4	Exemples	22
3.3.5	Résultats sur la convergence des méthodes itératives	23

3.1 Introduction

Résolution du système $Ax = b$ avec $A \in M_{n,n}(K)$, $b, x \in K^n$ et $K = \mathbb{R}$ ou \mathbb{C} .

Une méthode itérative consiste à :

- à partir de $x^{(0)}$, “ estimation ” de x ,
- à générer une suite de vecteurs $x^{(h)}$ par une relation de récurrence :

$$x^{(h+1)} = F_k \left(x^{(h)}, x^{(h-1)}, \dots, x^{(h-i)} \right)$$

relation multi-pas sur i termes.

Si F_k est indépendante de k , on parle d’itération stationnaire.

Remarque 3.1. La méthode est dite convergente si la suite $(x^{(k)})$ converge vers la solution x du système. D'une manière pratique, on arrête les calculs avec le test :

$$\frac{\|b - Ax^{(k)}\|}{\|b\|} \leq \varepsilon$$

pour un ε donné. Si $\varepsilon = 10^{-p}$, les p premiers chiffres significatifs de b et $Ax^{(k)}$ sont identiques. Plus précisément, le sens de l'identité dépend de la norme utilisée :

- en moyenne sur les n coefficients si $\|\cdot\| = \|\cdot\|_1$,
- pour tous les coefficients si $\|\cdot\| = \|\cdot\|_\infty$.

Une autre condition d'arrêt, moins fiable, consiste à regarder l'évolution de la suite :

$$\frac{\|x^{(k+1)} - x^{(k)}\|}{\|x^{(k)}\|} \leq \varepsilon$$

Il y a un problème si, en cours de calcul, on obtient $x^{(k)} = 0$.

3.2 Une étude abstraite générale

On écrit A sous la forme $M - N$ avec $M, N \in M_{n,n}(K)$.

$$A = M - N$$

M doit être facilement inversible, c'est à dire que $Mx = y$ doit être facile à résoudre (système diagonal ou triangulaire, par exemple).

On a le système :

$$\begin{aligned} Ax &= b \\ (M - N)x &= b \\ Mx - Nx &= b \\ Mx &= Nx + b \end{aligned}$$

C'est une formule de point fixe ($f(x) = x$). On peut en déduire un processus itératif.

$$\begin{cases} x^{(0)} \text{ donné dans } K^n \\ Mx^{(k+1)} = Nx^{(k)} + b \end{cases}$$

À chaque itération, on doit résoudre un système linéaire de matrice M que l'on peut écrire

$$x^{(k+1)} = M^{-1}Nx^{(k)} + M^{-1}b$$

Dans la pratique, on ne calcule pas M^{-1} mais on résout le système de matrice M .

Étude de convergence x solution de $Ax = b$. On pose $e^{(k)} = x^{(k)} - x$, l'erreur à l'étape k .

$$\begin{aligned} x^{(k+1)} &= M^{-1}Nx^{(k)} + M^{-1}b \\ e^{(k+1)} &= x^{(k+1)} - x \\ &= M^{-1}Nx^{(k)} + M^{-1}b - x \\ &= M^{-1}Nx^{(k)} - M^{-1}Nx \\ &= M^{-1}N(x^{(k)} - x) \\ &= M^{-1}Ne^{(k)} \end{aligned}$$

Pour passer de $e^{(k)}$ à $e^{(k+1)}$,

$$e^{(k+1)} = M^{-1}Ne^{(k)}$$

$T = M^{-1}N$ est la matrice d'itération :

$$\begin{aligned} e^{(k+1)} &= Te^{(k)} \\ &= T^2e^{(k-1)} \\ &\vdots \\ e^{(k)} &= T^ke^{(0)} \end{aligned}$$

La méthode converge si $e^{(k)} \xrightarrow[k \rightarrow \infty]{} 0$.

$$\begin{aligned} T^ke^{(0)} &\xrightarrow[k \rightarrow \infty]{} 0 \\ \lim_{k \rightarrow \infty} T^k &= 0 \end{aligned}$$

Théoreme 3.1 (admis). *Si $\|\cdot\|$ est une norme matricielle subordonnée telle que $\|T\| < 1$ alors le processus itératif converge vers la solution de $Ax = b$.*

Rappel de la définition d'une norme matricielle subordonnée (cf. l'annexe sur les rappels d'algèbre linéaire) :

Définition 3.1. *Norme matricielle associée, subordonnée et induite par une norme vectorielle.*

Soient $A \in M_{n,n}(K)$, N_m et N_n deux normes vectorielles sur K^m et K^n . On appelle norme matricielle induite, la norme :

$$\|A\| = \max_{N_n(x)=1} N_m(Ax)$$

3.3 Étude de cas particuliers de décomposition

$$A = M - N$$

Décomposition $A = D - E - F$, où :

$$\begin{bmatrix} \ddots & & -F \\ & D & \\ -E & & \ddots \end{bmatrix}$$

- D est une matrice diagonale,

$$\begin{cases} D_{ij} = 0 \text{ si } i \neq j \\ D_{ii} = A_{ii} \end{cases}$$

- E est une matrice triangulaire inférieure,

$$\begin{cases} E_{ij} = 0 \text{ si } i \leq j \\ E_{ij} = -A_{ij} \text{ si } i > j \end{cases}$$

- F est une matrice triangulaire supérieure,

$$\begin{cases} F_{ij} = 0 \text{ si } i \geq j \\ F_{ij} = -A_{ij} \text{ si } i < j \end{cases}$$

3.3.1 Méthode de Jacobi

Formulation matricielle $A = M - N = D - E - F$, avec $M = D$ et $N = E + F$. Le processus itératif :

$$\begin{cases} x^{(0)} \text{ donné} \\ Mx^{(k+1)} = Nx^{(k)} + b \end{cases}$$

À chaque itération, on doit résoudre le système diagonal :

$$Dx^{(k+1)} = (E + F)x^{(k)} + b$$

avec $D_{ii} = A_{ii} \neq 0, \forall i$.

Mise en œuvre effective

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1$$

$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2$$

$$a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = b_3$$

Pour appliquer le processus itératif, on transforme :

$$a_{11}x_1^{(k+1)} = -a_{12}x_2^{(k)} - a_{13}x_3^{(k)} + b_1$$

$$a_{22}x_2^{(k+1)} = -a_{21}x_1^{(k)} - a_{23}x_3^{(k)} + b_2$$

$$a_{33}x_3^{(k+1)} = -a_{31}x_1^{(k)} - a_{32}x_2^{(k)} + b_3$$

D'où

$$x_1^{(k+1)} = \frac{-a_{12}x_2^{(k)} - a_{13}x_3^{(k)} + b_1}{a_{11}}$$

$$x_2^{(k+1)} = \frac{-a_{21}x_1^{(k)} - a_{23}x_3^{(k)} + b_2}{a_{22}}$$

$$x_3^{(k+1)} = \frac{-a_{31}x_1^{(k)} - a_{32}x_2^{(k)} + b_3}{a_{33}}$$

3.3.2 Méthode de Gauss-Seidel

Formulation matricielle Cette fois-ci on pose $M = D - E$ et $N = F$. On doit alors résoudre à chaque itération le système diagonal :

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

$(D - E)$ est une matrice triangulaire.

Mise en œuvre effective Les coefficients de E sont à l'ordre $(k + 1)$ maintenant :

$$a_{11}x_1^{(k+1)} = -a_{12}x_2^{(k)} - a_{13}x_3^{(k)} + b_1$$

$$a_{22}x_2^{(k+1)} = -a_{21}x_1^{(k+1)} - a_{23}x_3^{(k)} + b_2$$

$$a_{33}x_3^{(k+1)} = -a_{31}x_1^{(k+1)} - a_{32}x_2^{(k+1)} + b_3$$

Soit

$$x_1^{(k+1)} = \frac{-a_{12}x_2^{(k)} - a_{13}x_3^{(k)} + b_1}{a_{11}}$$

$$x_2^{(k+1)} = \frac{-a_{21}x_1^{(k+1)} - a_{23}x_3^{(k)} + b_2}{a_{22}}$$

$$x_3^{(k+1)} = \frac{-a_{31}x_1^{(k+1)} - a_{32}x_2^{(k+1)} + b_3}{a_{33}}$$

Même si les x à l'ordre $(k + 1)$ ne sont pas tous du côté gauche des équations, si l'on résout le système de haut en bas, ils seront tous calculés avant d'être utilisés.

Mais finalement, il n'est utile de mémoriser les deux vecteurs $x^{(k)}$ et $x^{(k+1)}$ et tous les calculs peuvent être faits dans le seul vecteur x et la formule précédente se simplifie de manière équivalente en :

$$\begin{aligned} x_1 &\leftarrow \frac{-a_{12}x_2 - a_{13}x_3 + b_1}{a_{11}} \\ x_2 &\leftarrow \frac{-a_{21}x_1 - a_{23}x_3 + b_2}{a_{22}} \\ x_3 &\leftarrow \frac{-a_{31}x_1 - a_{32}x_2 + b_3}{a_{33}} \end{aligned}$$

3.3.3 Méthode de relaxation

Formulation de la méthode de Gauss-Seidel :

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

$x_i^{(k+1)} - x_i^{(k)} = \Delta_i^{(k+1)}$ est l'écart entre deux itérés successifs.

Dans la méthode de relaxation, on introduit un facteur de relaxation ω :

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

pour obtenir la formule de récurrence :

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

On peut vérifier que la formulation matricielle associée est

$$\begin{aligned} M &= \frac{1}{\omega}(D - \omega E) \\ N &= \frac{1}{\omega}((1 - \omega)D + \omega F) \end{aligned}$$

3.3.4 Exemples

Résoudre

$$\begin{cases} 10x_1 + x_2 = 11 \\ 2x_1 + 10x_2 = 12 \end{cases}$$

De solution exacte $\left(\frac{1}{1}\right)$. On part de $x^{(0)} \begin{pmatrix} 0 \\ 0 \end{pmatrix}$, et on applique les méthodes de Jacobi et de Gauss-Seidel.

Méthode de Jacobi :

$$\begin{cases} x_1^{(k+1)} = \frac{11 - x_2^{(k)}}{10} \\ x_2^{(k+1)} = \frac{12 - 2x_1^{(k)}}{10} \end{cases}$$

$$x^{(0)} \begin{pmatrix} 0 \\ 0 \end{pmatrix} x^{(1)} \begin{pmatrix} 11/10 \\ 12/10 \end{pmatrix} x^{(2)} \begin{pmatrix} 98/100 \\ 98/100 \end{pmatrix} x^{(3)} \begin{pmatrix} 1002/1000 \\ 1004/1000 \end{pmatrix} \dots$$

Méthode de Gauss-Seidel :

$$\begin{cases} x_1^{(k+1)} = \frac{11 - x_2^{(k)}}{10} \\ x_2^{(k+1)} = \frac{12 - 2x_1^{(k+1)}}{10} \end{cases}$$

$$x^{(0)} \begin{pmatrix} 0 \\ 0 \end{pmatrix} x^{(1)} \begin{pmatrix} 11/10 \\ 98/10 \end{pmatrix} x^{(2)} \begin{pmatrix} 1002/1000 \\ 9996/10000 \end{pmatrix} \dots$$

Résoudre

$$\begin{cases} x_1 + 10x_2 = 11 \\ 10x_1 + 2x_2 = 12 \end{cases}$$

de même solution que le système précédent.

Méthode de Jacobi :

$$\begin{cases} x_1^{(k+1)} = 11 - 10x_2^{(k)} \\ x_2^{(k+1)} = 6 - 5x_1^{(k)} \end{cases}$$

$$x^{(0)} \begin{pmatrix} 0 \\ 0 \end{pmatrix} x^{(1)} \begin{pmatrix} 11 \\ 6 \end{pmatrix} x^{(2)} \begin{pmatrix} -49 \\ -49 \end{pmatrix} x^{(3)} \begin{pmatrix} 501 \\ 251 \end{pmatrix} \dots$$

Méthode de Gauss-Seidel :

$$\begin{cases} x_1^{(k+1)} = 11 - 10x_2^{(k)} \\ x_2^{(k+1)} = 6 - 5x_1^{(k+1)} \end{cases}$$

$$x^{(0)} \begin{pmatrix} 0 \\ 0 \end{pmatrix} x^{(1)} \begin{pmatrix} 11 \\ -49 \end{pmatrix} x^{(2)} \begin{pmatrix} 501 \\ -2499 \end{pmatrix} \dots$$

3.3.5 Résultats sur la convergence des méthodes itératives

Proposition 3.1. *Si A est à diagonale strictement dominante, alors quelque soit $x^{(0)}$ donné :*

- *la méthode de Jacobi converge,*
- *la méthode de Gauss-Seidel converge,*
- *la méthode de relaxation converge, si $0 < \omega \leq 1$.*

Proposition 3.2. *Si A est hermitienne définie positive, alors quelque soit $x^{(0)}$:*

- *la méthode de Gauss-Seidel converge,*
- *la méthode de relaxation converge si et seulement si $0 < \omega < 2$.*

Proposition 3.3. *La méthode de relaxation converge implique que $0 < \omega < 2$.*

On retrouvera les définitions des matrices à diagonale dominante et des matrices hermitiennes définies positives dans l'annexe sur les rappels d'algèbre linéaire.

Chapter 4

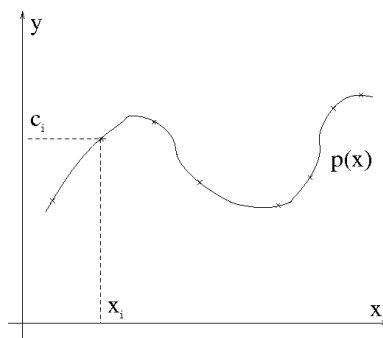
Interpolation polynomiale et splines cubiques

4.1 Introduction

Soient $(x_i, y_i)_{0 \leq i \leq n}$ des points de coordonnées réelles (qui peuvent être, par exemple, des résultats de mesures expérimentales). On veut faire correspondre à ces points, une fonction $f(x)$ de façon à estimer $f(x)$ pour $x \neq x_i$.

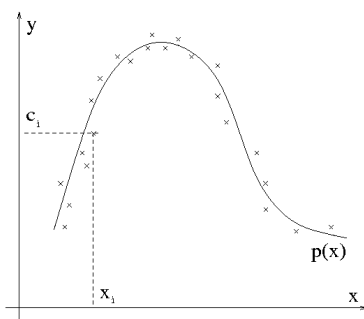
Deux types de problèmes peuvent alors se poser :

- Construire une courbe passant exactement par les points. Il s'agit alors d'un problème d'**interpolation**



- Construire une courbe qui, *dans un sens à préciser* s'approche des points. Il s'agit alors d'un problème d'**approximation**.

Dans la suite de ce chapitre, on s'intéresse au problème d'interpolation. On traitera les problèmes d'approximation au chapitre suivant.



4.2 Interpolation polynômiale

Pb : On connaît $(n+1)$ points $(x_i, y_i)_{0 \leq i \leq n}$ et on cherche le polynôme de degré n passant par ces $(n+1)$ points.

- On sait que ce polynôme est unique ;
- on peut le rechercher sous la forme :

$$P_n(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$$

c'est à dire calculer les $(n+1)$ coefficients a_i .

Pour chaque (x_j, y_j) avec $0 \leq j \leq n$, on écrit que

$P_n(x_j) = y_j \rightarrow 1$ équation linéaire par rapport aux $(n+1)$ coefficients a_i .

et donc pour $0 \leq j \leq n \rightarrow$ système de $(n+1)$ équations à $(n+1)$ inconnues.

Dans la pratique, on ne retient pas ce procédé car :

- il est coûteux : $O(n^3)$
- il est souvent mal conditionné car les coef. de la matrice sont

$$\begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{bmatrix}$$

x_j et x_j^n sont souvent d'ordre très différents sauf pour n petit.

Les méthodes auxquelles on va s'intéresser (Lagrange e Newton) ne calculent pas les coef. a_i du polynôme mais calculent directement $p_n(x)$ pour une valeur donnée de x .

4.2.1 Méthode de Lagrange

A partir des points de données x_i , $0 \leq i \leq n$, on considère les polynômes de Lagrange :

$$L_k(x) = \prod_{i=0, i \neq k}^n \frac{x - x_i}{x_k - x_i} \text{ pour } 0 \leq k \leq n$$

Ce sont des polynômes de degré n qui vérifient

$$L_k(x_j) = \delta_{kj} = \begin{cases} 0 & \text{si } j \neq k \\ 1 & \text{si } j = k \end{cases}$$

L'évaluation de $P_n(x)$ en un point donné se fait alors par la formule :

$$P_n(x) = \sum_{k=0}^n y_k L_k(x)$$

On peut facilement vérifier que

$$P_n(x_j) = y_j \quad \forall j \quad 0 \leq j \leq n$$

Calcul de la complexité

$$C(L_k(x)) \simeq (2 \text{ soust.} + 1 \text{ div.}) n + n \text{ mult.}$$

Pour tous les L_k on a une complexité en $4n^2$

Pour le calcul de $P_n(x)$ on a $(n+1)$ mult. + n add. soit au total de l'ordre de $4n^2$ c'est à dire en $O(n^2)$

En TD, on proposera un arrangement des formules un peu plus efficace surtout si l'on évalue en plusieurs x .

Exemple

Construire l'interpolation de Lagrange de la fonction $y = \sin(\pi x)$ pour les points d'abscisses $x_0 = 0$; $x_1 = \frac{1}{6}$ et $x_2 = \frac{1}{2}$.

Solution :

$$y_0 = \sin(\pi \cdot 0) = 0$$

$$y_1 = \sin\left(\frac{\pi}{6}\right) = \frac{1}{2}$$

$$y_2 = \sin\left(\frac{\pi}{2}\right) = 1$$

$$P_2(x) = 0 \cdot \frac{?}{?} + \frac{1}{2} \cdot \frac{x \left(x - \frac{1}{2} \right)}{\frac{1}{6} \cdot \left(\frac{1}{6} - \frac{1}{2} \right)} + 1 \cdot \frac{x \left(x - \frac{1}{6} \right)}{\frac{1}{2} \cdot \left(\frac{1}{2} - \frac{1}{6} \right)}$$

Après simplification, on a

$$P_2(x) = \frac{7}{2}x - 3x^2$$

Attention dans la pratique, on ne cherche pas une expression de $P_n(x)$ sous la forme d'une fonction polynômiale.

Estimation de l'erreur du calcul d'interpolation polynômiale

Soit $P_n(x)$ le polynôme de degré n qui interpole les points $(x_i, y_i)_{0 \leq i \leq n}$ que l'on suppose appartenir au graphe d'une fonction f .

Problème : quelle est l'erreur commise entre $P_n(x)$ et $f(x)$?

Remarque : c'est un problème "très théorique" car dans la pratique f n'est pas connue !

Théorème 4.1. Soit $a = \min\{x_i; 0 \leq i \leq n\}$ et $b = \max\{x_i; 0 \leq i \leq n\}$. Si $f \in C^{n+1}[a, b]$ alors $\forall x \in [a, b], \exists \xi \in [a, b]$ t.q.

$$E(x) = f(x) - P_n(x) = \frac{\gamma_{n+1}(x)}{(n+1)!} f^{(n+1)}(\xi)$$

où $\gamma_{n+1}(x) = \prod_{i=0}^n (x - x_i)$. On en déduit alors que

$$|E(x)| \leq \frac{\gamma_{n+1}(x)}{(n+1)!} M_{n+1}$$

où $M_{n+1} = \max_{x \in [a, b]} |f^{(n+1)}(x)|$

Proof. Soit \bar{x} tel que $\forall i, \bar{x} \neq x_i$, on considère la fonction :

$$u(t) = f(t) - P_n(t) - \frac{f(\bar{x}) - P_n(\bar{x})}{\gamma_{n+1}(\bar{x})} \gamma_{n+1}(t)$$

alors, $\forall i, 0 \leq i \leq n$, on peut facilement vérifier que $u(x_i) = 0$ et de plus $u(\bar{x}) = 0$. Donc u a $n+2$ racines distinctes sur $[a, b]$.

Par ailleurs, $u \in C^{n+1}[a, b]$ car composé de fonctions (f, P_n et γ_{n+1} ayant cette propriété. On applique alors le théorème de Rolle Le théorème de Rolle nous dit que si une fonction $g \in C^1[a, b]$, avec $g(a) = g(b)$, alors $\exists c \in [a, b]$ tel que $g'(c) = 0$ (voir la figure 4.2.1).

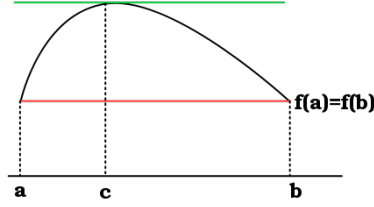


Figure 4.1: Théorème de Rolle

Donc sur chaque segment $[x_0, x_1], \dots, [x_j, \bar{x}], [\bar{x}, x_{j+1}], \dots, [x_{n-1}, x_n]$, la fonction u s'annule aux extrémités des $(n + 1)$ intervalles. Le théorème de Rolle, nous dit donc que u' a au moins $(n + 1)$ racines. En répétant ce processus sur u' , on en déduit que u'' a au moins n racines, ...

Finalement, on sait que $u^{(n+1)}$ a au moins une racine que l'on va noter ξ et on a :

$$0 = u^{(n+1)}(\xi) = f^{(n+1)}(\xi) - P_n^{(n+1)}(\xi) - \frac{f(\bar{x}) - P_n(\bar{x})}{\gamma_{n+1}(\bar{x})} \gamma_{n+1}^{(n+1)}(\xi) \quad (4.1)$$

Or P_n est un polynôme de degré n donc $P_n^{(n+1)} \equiv 0$.

$\gamma_{n+1}(x) = \prod_{i=0}^n (x - x_i)$, est de degré $(n + 1)$ et son monôme de plus haut degré est x^{n+1} . La dérivé d'ordre $n + 1$ de γ_{n+1} , c'est à dire $\gamma_{n+1}^{(n+1)}$, est donc une constante qui vaut $(n + 1)!$.

Finalement l'équation 4.1 se simplifie en :

$$0 = f^{(n+1)}(\xi) - \frac{f(\bar{x}) - P_n(\bar{x})}{\gamma_{n+1}(\bar{x})} (n + 1)!$$

D'où

$$f(\bar{x}) - P_n(\bar{x}) = \frac{\gamma_{n+1}(\bar{x})}{(n + 1)!} f^{(n+1)}(\xi)$$

□

4.2.2 Méthode de Newton

La méthode de Newton consiste à faire à nouveau une évaluation du polynôme d'interpolation en un point x donné, grâce à une autre formule que celle de la méthode de Lagrange.

Soient donc les n points $(x_i, y_i)_{0 \leq i \leq n}$, on recherche le polynôme de degré n passant par ces points. On introduit la notion de "différences divisées" que l'on définit récursivement par :

- à l'ordre 0 : $\Delta_i^0 = y_i$
- à l'ordre j : $\Delta_i^j = \frac{\Delta_{i+1}^{j-1} - \Delta_i^{j-1}}{x_{j+i} - x_i}$

Ce calcul peut se schématiser de la manière suivante :

$$\begin{array}{ccccccc}
 (x_0, \Delta_0^0) & & & & & & \\
 & \searrow & & & & & \\
 (x_1, \Delta_1^0) & \rightarrow & \Delta_0^1 & & & & \\
 & \searrow & & \searrow & & & \\
 (x_2, \Delta_2^0) & \rightarrow & \Delta_1^1 & \rightarrow & \Delta_0^2 & & \\
 & \vdots & \vdots & & \vdots & \ddots & \\
 & \vdots & \vdots & & \vdots & & \\
 (x_n, \Delta_n^0) & \rightarrow & \Delta_{n-1}^1 & \rightarrow & \Delta_{n-2}^2 & \dots & \rightarrow \Delta_0^n
 \end{array}$$

Comme pour la méthode de Lagrange, on va définir une base de décomposition, similaire aux polynômes de Lagrange. Cette base, ici, s'écrit sous la forme :

$$N_k(x) = \prod_{j=0}^{k-1} (x - x_j)$$

avec $N_0(x) = 1$

On montre alors que le polynôme d'interpolation s'écrit, dans cette base :

$$P_n(x) = \sum_{k=0}^n \Delta_0^k N_k(x)$$

La démonstration de cette formule est assez simple : le problème consiste à trouver la décomposition du polynôme d'interpolation, qui doit donc vérifier que $P_n(x_i) = y_i$, dans la base utilisée. On doit donc déterminer les coefficient a_k de

$$P_n(x) = \sum_{k=0}^n a_k N_k(x)$$

La vérification que P_n est un polynôme d'interpolation consiste donc à résoudre le système suivant

$$\begin{bmatrix} N_0(x_0) & 0 & \dots & \dots & 0 \\ N_0(x_1) & N_1(x_1) & \ddots & & \vdots \\ N_0(x_2) & N_1(x_2) & N_2(x_2) & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & 0 \\ N_0(x_n) & N_1(x_n) & N_2(x_n) & \dots & N_n(x_n) \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

La résolution simple de ce système triangulaire permet de retrouver que les coefficients a_k correspondent bien aux différences divisées, Δ_0^k , définies par la formule récursive ci-dessus.

L'intérêt de la méthode de Newton réside dans le fait que si on est amené à ajouter un $(n + 1)$ ème point de support, (x_{n+1}, y_{n+1}) , alors il n'est pas nécessaire de reprendre tout le calcul : il suffit simplement de rajouter une ligne au calcul récursif décrit dans le schéma en triangle décrit ci-dessus et de rajouter un terme à la somme (en faisant varier k jusqu'à $n + 1$) définissant le polynôme d'interpolation.

4.2.3 Limite de l'interpolation polynômiale

Les deux méthodes précédentes consistent à calculer un polynôme de degré n passant par n points de support. Ce calcul présente des inconvénients lorsque n devient grand car le polynôme d'interpolation peut présenter des oscillations entre les points de support qui ne sont pas forcément souhaitées.

Pour illustrer ce problème, on a tracé sur la figure 4.2, des calculs du polynôme d'interpolation que l'on obtient en prenant des points de support d'interpolation issus de la fonction $f(x) = |x - 5|$. On voit que assez rapidement, au-delà de 6 points que des oscillations non souhaitées commencent à apparaître. Il est à noter qu'elles correspondent bien au calcul exacte du polynôme d'interpolation et ne correspondent pas à d'éventuelles erreurs d'arrondis. De même sur la figure 4.3, on donne des exemples de tracé de polynôme d'interpolation à partir d'un nombre de points de support relativement peu élevés et on constate à nouveau l'apparition d'oscillations.

Pour aller plus loin que ces observations, on peut donner une propriétés montrées par le mathématicien Runge qui s'est intéressé au polynôme d'interpolation P_n construit à partir de points de support issus de la fonction $f(x) = \frac{1}{1+x^2}$ qui, contrairement à l'exemple de la fonction de la figure 4.2, est infiniment dérivable. Runge a montré que sur l'intervalle $[-5; 5]$, $\lim_{n \rightarrow \infty} P_n(x) = \infty$ et ceci $\forall x$ t.q. $|x| \geq 3,63$.

En conclusion, il n'est pas recommandé d'utiliser des interpolations polynômiales dès que le degré du polynôme est relativement grand (≥ 10). Si on a un

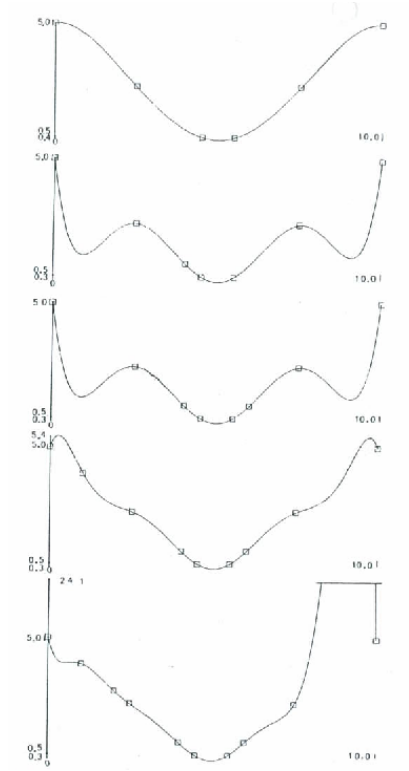


Figure 4.2: Interpolation polynômiale de points issus de la courbe $|x - 5|$ avec 6,7,8,9 et 10 points

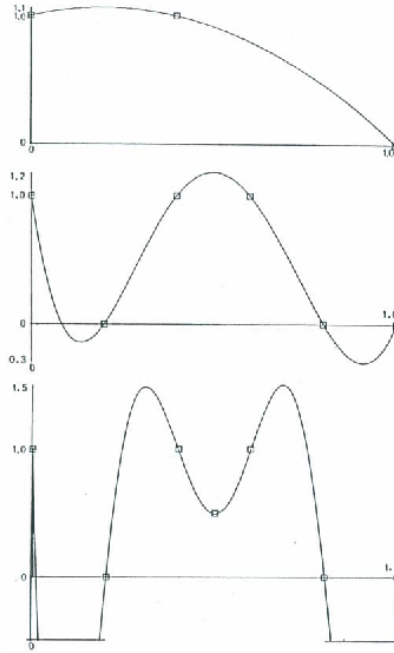


Figure 4.3: Interpolation polynômiale basée sur 3, 6 et 7 points

support de points important, comme cela peut arriver pour des designers qui veulent digitaliser des profils de voitures ou d'avions à partir d'un ensemble finis de points, il faut utiliser d'autres méthodes. La digitalisation de ces profils a permis de développer de nouvelles méthodes dans les années 1980, à l'époque où le dessin assisté par ordinateur est devenu de plus en plus présent dans le monde de l'ingénierie. Un ingénieur de Renault, Paul Bézier, a donné son nom à une famille de méthodes, appelées les courbes de Bézier, très fréquemment implémentées dans les logiciels de dessins basiques aujourd'hui. Une autre famille d'interpolation et d'approximation, nommées courbes splines, a aussi été développée, à cette époque.

Sur les figures 4.4, 4.5, 4.6, on représente une méthode d'interpolation appelée Splines Cubiques et qui fait l'objet de la section suivante. On a repris sur ces figures, des cas similaires de points de support que sur les courbes d'interpolation polynômiales et on constate que les oscillations ne sont pas présentes même sur un support contenant un nombre élevé de points.

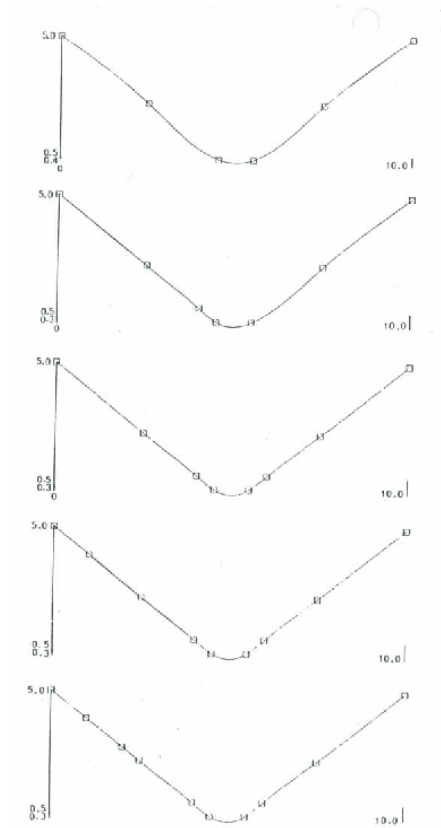


Figure 4.4: Interpolation par Spline cubique de points issus de la courbe $|x - 5|$ avec 6,7,8,9 et 10 points

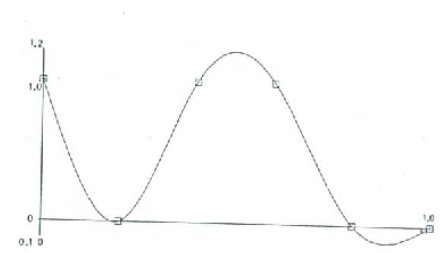


Figure 4.5: Interpolation par Spline cubique sur 6 points

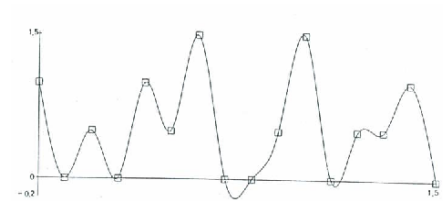


Figure 4.6: Interpolation par Spline cubique sur 16 points

4.3 Méthodes de Splines Cubiques

Interpolation polynomiale par morceaux

4.3.1 Description

Les interpolations polynômiales décrites au paragraphe précédent ne sont utilisables que si on a un nombre de points assez réduits. En effet, si celui-ci est trop élevé, on obtient des variations/oscillations importantes du polynôme d'interpolation entre certain points successifs du support d'interpolation. On peut observer ce phénomène sur les figures ...

Une fonction d'interpolation plus "douce" peut être obtenue en utilisant une analogie mathématique correspondant à l'usage de règle flexible fréquemment manipulée par les dessinateurs industriels. Une interprétation physique de la déformation de ces règles consiste à penser que le dessinateur cherche à minimiser leur déformation. Cela peut alors se traduire sous la forme mathématique suivante.

Soient $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$, les $(n + 1)$ points de support donnés pour le calcul d'interpolation. On considère la fonction d'interpolation $g(x)$ qui passe par ces points de telle manière qu'entre deux points successifs, d'abscisse x_j et x_{j+1} , elle corresponde à un polynôme de degré 3 :

$$g(x) = a_0^j + a_1^j x + a_2^j x^2 + a_3^j x^3$$

Les indices supérieurs ne sont pas des puissances mais sont simplement là pour rappeler que les coefficients ne sont valables que dans l'intervalle $[x_j; x_{j+1}]$.

Pour déterminer les 4 coefficients a_0^j, a_1^j, a_2^j et a_3^j , on impose que la fonction passe bien par les deux extrémités de l'intervalle (puisque que c'est une fonction d'interpolation), à savoir que

$$g(x_j) = y_j \text{ et } g(x_{j+1}) = y_{j+1}$$

et de plus on impose que $\frac{dg}{dx}(x)$ et $\frac{d^2g}{dx^2}(x)$ soient continues en x_j , de sorte que les cubiques définies sur chaque intervalle $[x_j; x_{j+1}]$ se raccordent sans point anguleux.

Or $g''(x) = \frac{d^2g}{dx^2}(x) = 2a_2^j + 6a_3^j x$ est linéaire,

d'où pour $x_j \leq x \leq x_{j+1}$, les rapports suivants sont égaux et correspondent au coefficient directeur de la droite représentative de $g''(x)$:

$$\frac{g''(x) - g''(x_j)}{x - x_j} = \frac{g''(x_{j+1}) - g''(x_j)}{x_{j+1} - x_j}$$

C'est à dire :

$$g''(x) = g''(x_j) + \frac{x - x_j}{x_{j+1} - x_j} (g''(x_{j+1}) - g''(x_j))$$

Pour trouver $g(x)$, on intègre 2 fois cette équation et alors on fait apparaître 2 constantes d'intégration, A et B :

$$g'(x) = (x - x_j)g''(x_j) + \frac{(x - x_j)^2}{2(x_{j+1} - x_j)} (g''(x_{j+1}) - g''(x_j)) + A$$

puis

$$g(x) = \frac{(x - x_j)^2}{2} g''(x_j) + \frac{(x - x_j)^3}{6(x_{j+1} - x_j)} (g''(x_{j+1}) - g''(x_j)) + Ax + B$$

en imposant que $g(x_j) = y_j$ et $g(x_{j+1}) = y_{j+1}$, on détermine les constantes d'intégration, A et B , et on obtient finalement, tous calculs effectués :

$$\begin{aligned} g(x) &= \frac{g''(x_j)}{6} \left(\frac{(x_{j+1} - x)^3}{x_{j+1} - x_j} - (x_{j+1} - x_j)(x_{j+1} - x) \right) \\ &+ \frac{g''(x_{j+1})}{6} \left(\frac{(x - x_j)^3}{x_{j+1} - x_j} - (x_{j+1} - x_j)(x - x_j) \right) \\ &+ y_j \left(\frac{x_{j+1} - x}{x_{j+1} - x_j} \right) + y_{j+1} \left(\frac{x - x_j}{x_{j+1} - x_j} \right) \end{aligned} \quad (4.2)$$

On peut alors vérifier facilement que $g(x_j) = y_j$ et $g(x_{j+1}) = y_{j+1}$.

Cette dernière équation n'est pas utilisable car on ne connaît pas explicitement $g''(x_j)$ et $g''(x_{j+1})$. Mais il reste à utiliser la dernière hypothèse qui nous dit que $g'(x)$ est une fonction continue.

Dérivons la dernière équation. Pour $x_j \leq x \leq x_{j+1}$, on a :

$$\begin{aligned} g'(x) &= \frac{g''(x_j)}{6} \left(-\frac{3(x_{j+1} - x)^2}{x_{j+1} - x_j} + (x_{j+1} - x_j) \right) \\ &+ \frac{g''(x_{j+1})}{6} \left(\frac{3(x - x_j)^2}{x_{j+1} - x_j} - (x_{j+1} - x_j) \right) \\ &+ \frac{y_{j+1} - y_j}{x_{j+1} - x_j} \end{aligned}$$

Mais pour l'intervalle voisin $x_{j-1} \leq x \leq x_j$, on peut obtenir une formule analogue, en décalant tous les indices de -1, par exemple j en $j - 1$:

$$\begin{aligned}
 g'(x) &= \frac{g''(x_{j-1})}{6} \left(-\frac{3(x_j - x)^2}{x_j - x_{j-1}} + (x_j - x_{j-1}) \right) \\
 &+ \frac{g''(x_j)}{6} \left(\frac{3(x - x_{j-1})^2}{x_j - x_{j-1}} - (x_j - x_{j-1}) \right) \\
 &+ \frac{y_j - y_{j-1}}{x_j - x_{j-1}}
 \end{aligned}$$

La continuité de $g'(x)$ est donc assurée en écrivant l'égalité des 2 expressions de $g'(x_j)$ données par les 2 expressions précédentes. Tous calculs effectués, on a

$$\begin{aligned}
 (x_j - x_{j-1})g''(x_{j-1}) + 2(x_{j+1} - x_{j-1})g''(x_j) + (x_{j+1} - x_j)g''(x_{j+1}) & \quad (4.3) \\
 = 6 \left(\frac{y_{j+1} - y_j}{x_{j+1} - x_j} - \frac{y_j - y_{j-1}}{x_j - x_{j-1}} \right)
 \end{aligned}$$

et ceci est valable pour $j = 1, 2, \dots, n - 1$.

En posant $g''(x_0) = g''(x_n) = 0$, on obtient finalement un système linéaire symétrique et tridiagonal, de dimension $n - 1$, dont on peut vérifier facilement qu'il est à diagonale dominante.

4.3.2 Algorithme

Les formules précédentes permettent donc de décomposer le calcul d'interpolation en 2 étapes :

- On résout le système tridiagonal (4.3), ce qui permet de déterminer toutes les dérivées secondes $g''(x_j)$ pour $1 \leq j \leq n - 1$, les dérivées secondes en x_0 et x_n sont pris comme étant arbitrairement nulle ;
- Une fois ces dérivées secondes calculées, on peut appliquer la formule (4.2) et ainsi obtenir une valeur pour $g(x)$ pour $x \in [a, b]$. Pour cela, on doit déjà calculer à quel intervalle $I_j = [x_j, x_{j+1}]$, x appartient. On utilisera alors la formule (4.2) sur cet intervalle I_j .

L'algorithme s'écrit en 2 parties.

Partie 1

On construit le système tridiagonal qui permet de calculer les dérivées secondes de la fonction spline. Il est de la forme

$$\begin{pmatrix} b_1 & d_1 & 0 & \dots & 0 \\ a_2 & b_2 & d_2 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & d_{n-2} \\ 0 & \dots & 0 & a_{n-1} & b_{n-1} \end{pmatrix} \begin{pmatrix} g_1 \\ g_2 \\ \vdots \\ \vdots \\ g_{n-1} \end{pmatrix} = \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ \vdots \\ e_{n-1} \end{pmatrix}$$

puis on le résout en utilisant la méthode de Thomas vue en TD.

Algorithme 1: Splines cubiques - partie 1 : calcul des dérivées secondes

Données :

- $(x_i, y_i); 0 \leq i \leq n$: points de support d'interpolation
- n : nombre de points de support (à une unité près)

Sorties :

- $g_j : 0 \leq j \leq n$: dérivées secondes de la fonction spline d'interpolation

Début

$g_0 \leftarrow 0$;

$g_n \leftarrow 0$;

$b_1 \leftarrow 2 * (x_2 - x_0)$;

$d_1 \leftarrow x_2 - x_1$;

$e_1 \leftarrow 6 * ((y_2 - y_1)/(x_2 - x_1) - (c_1 - c_0)/(x_1 - x_0))$;

for $j \leftarrow 2$ à $n - 2$ **do**

$a_j \leftarrow d_{j-1}$ /*symétrie du système*/;

$b_j \leftarrow 2 * (x_{j+1} - x_{j-1})$;

$d_j \leftarrow x_{j+1} - x_j$;

$e_j \leftarrow 6 * ((y_{j+1} - y_j)/d_j - (y_j - y_{j-1})/a_j)$;

$a_{n-1} \leftarrow d_{n-2}$;

— to be continued — Fin

4.4 Interpolation bidimensionnelle

On n'aborde dans cette section qu'une introduction à l'interpolation bidimensionnelle, en dérivant les méthodes de calcul les plus usuelles.

Il s'agit ici de construire des surfaces d'interpolation à partir d'un support de points $\{(x_i, y_i, z_i); 0 \leq i \leq n\}$ tels que la troisième coordonnée, correspondant ici aux valeurs de z_i , soit l'image d'une fonction de deux variables : $z = f(x, y)$, comme représentée par la figure 4.7.

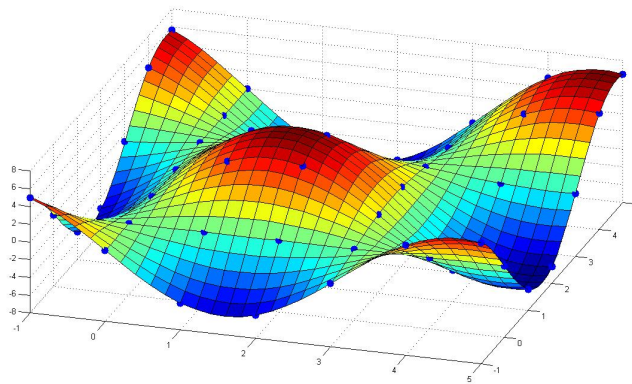


Figure 4.7: Interpolation régulière bidimensionnelle

Connaissant les points de support par lesquels cette fonction d'interpolation f doit passer, on cherche à évaluer $f(x, y)$ pour des valeurs (x, y) différentes de celles de l'ensemble $\{(x_i, y_i); 0 \leq i \leq n\}$.

Ce calcul d'interpolation dépend de la manière dont le support de points $\{(x_i, y_i); 0 \leq i \leq n\}$ est construit. Nous nous intéresserons, plus particulièrement, à 3 cas particuliers :

1. Ces points $\{(x_i, y_i); 0 \leq i \leq n\}$ sont les points d'une grille régulière comme décrite dans la figure 4.8.
2. Ces points $\{(x_i, y_i); 0 \leq i \leq n\}$ sont ceux d'un maillage irrégulier formé de quadrangles, comme décrit dans la figure 4.9.
3. Ces points $\{(x_i, y_i); 0 \leq i \leq n\}$ sont ceux obtenus à partir d'une triangulation comme décrite dans la figure 4.10. Cette dernière description est la plus générale car elle peut se construire à partir d'un ensemble de points quelconque, grâce à la méthode de triangulation de Delaunay (largement référencée dans la littérature scientifique).

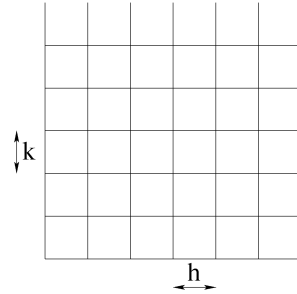


Figure 4.8: grille régulière bidimensionnelle

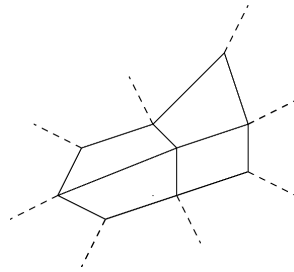


Figure 4.9: maillage irrégulier de quadrangles

Dans le cas 1 ou 2, on recherche une fonction d'un type donné sur chaque quadrangle ou rectangle.

Par exemple, sur le quadrangle de la figure 4.11, les valeurs de la fonction d'interpolation cherchée, notée f , sont connues aux 4 points de support, à savoir, $f(A)$, $f(B)$, $f(C)$ et $f(D)$ et on va rechercher la fonction f sous la forme particulière suivante :

$$f(x, y) = \alpha + \beta x + \gamma y + \delta xy$$

Cette fonction est celle qui a été utilisée sur la figure 4.7, à partir de la grille régulière qui a été projetée et dessinée sur la surface. Une telle fonction est appelée *bilinéaire* parce que pour x (resp. y) constant, elle est linéaire par rapport à l'autre variable y (resp. x).

Dans le cas général d'un maillage en quadrangle, c'est à dire le cas 2, on doit résoudre le système linéaire

$$\begin{cases} f(A) = \alpha + \beta x_A + \gamma y_A + \delta x_A y_A \\ f(B) = \alpha + \beta x_B + \gamma y_B + \delta x_B y_B \\ f(C) = \alpha + \beta x_C + \gamma y_C + \delta x_C y_C \\ f(D) = \alpha + \beta x_D + \gamma y_D + \delta x_D y_D \end{cases}$$

soit sous forme matricielle

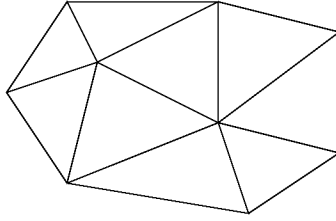


Figure 4.10: Triangulation

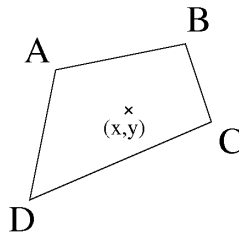


Figure 4.11: Un quadrangle

$$\begin{pmatrix} 1 & x_A & y_A & x_A y_A \\ 1 & x_B & y_B & x_B y_B \\ 1 & x_C & y_C & x_C y_C \\ 1 & x_D & y_D & x_D y_D \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \\ \gamma \\ \delta \end{pmatrix} = \begin{pmatrix} f(A) \\ f(B) \\ f(C) \\ f(D) \end{pmatrix}$$

que l'on peut résoudre assez facilement ...

Si on s'intéresse par exemple au cas 1 (comme un cas particulier du cas 2), les quadrangles sont alors des rectangles comme décrits dans la figure 4.12. On note ici $f_i; 1 \leq i \leq 4$ les valeurs respectives de f aux points A, B, C et D .

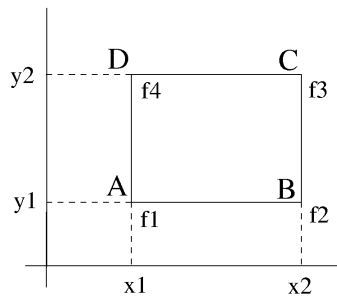


Figure 4.12: Rectangle d'interpolation

On peut exprimer la fonction d'interpolation f bilinéaire sous la forme :

$$f(x, y) = \sum_{i=1}^4 f_i p_i(x, y)$$

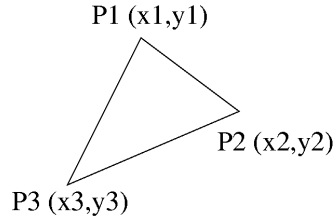


Figure 4.13: Triangle d'interpolation

avec

$$\begin{aligned}
 p_1(x, y) &= \frac{(x - x_2)(x - y_2)}{(x_1 - x_2)(y_1 - y_2)} \\
 p_2(x, y) &= \frac{(x - x_1)(x - y_2)}{(x_2 - x_1)(y_1 - y_2)} \\
 p_3(x, y) &= \frac{(x - x_1)(x - y_1)}{(x_2 - x_1)(y_2 - y_1)} \\
 p_4(x, y) &= \frac{(x - x_2)(x - y_1)}{(x_1 - x_2)(y_2 - y_1)}
 \end{aligned}$$

Pour le cas 3, on dispose d'une triangulation construite à partir d'un ensemble de points $\{(x_i, y_i); 0 \leq i \leq n\}$ pour lesquels on connaît des valeurs associées, notées z_i , par lesquelles doit passer la fonction à interpoler.

Soit le triangle décrit sur la figure 4.13 et défini par ses trois sommets et la valeur associée à interpoler, (x_1, y_1, z_1) , (x_2, y_2, z_2) et (x_3, y_3, z_3) .

On va rechercher ici, la fonction d'interpolation sous la forme d'une forme linéaire qui s'écrit

$$z = f(x, y) = \alpha x + \beta y + \gamma$$

En général, on ne fera ce calcul que pour des points (x, y) situés à l'intérieur du triangle (voir la remarque ci-après qui permet de caractériser ces points).

Pour simplifier les calculs, on réécrit facilement l'équation précédente sous la forme

$$ax + by + cz + d = 0$$

ce qui revient à poser

$$\begin{aligned}
 \alpha &= -a/c \\
 \beta &= -b/c \\
 \gamma &= -d/c
 \end{aligned}$$

CHAPTER 4. INTERPOLATION POLYNOMIALE ET SPLINES CUBIQUES

On peut alors montrer que les coefficients recherchés a , b , c et d se calculent par les formules suivantes

$$\begin{aligned} a &= y_1(z_2 - z_3) + y_2(z_3 - z_1) + y_3(z_1 - z_2) \\ b &= z_1(x_2 - x_3) + z_2(x_3 - x_1) + z_3(x_1 - x_2) \\ c &= x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2) \\ d &= -ax_1 - by_1 - cz_1 \end{aligned}$$

Remarque : Pour caractériser les points à l'intérieur d'un triangle défini par ses 3 sommets, P_1 , P_2 et P_3 , on peut l'exprimer par une formule de barycentre :

$$\overrightarrow{OP} = \sum_{i=1}^3 \omega_i \overrightarrow{OP_i}$$

Le point P est alors à l'intérieur du triangle $\{P_1, P_2, P_3\}$ si et seulement si $\forall i, \omega_i \geq 0$ et si

$$\sum_{i=1}^3 \omega_i = 1$$

Chapter 5

Approximation aux moindres carrés

5.1 Principe d'approximation et critère de moindres carrés

On considère les $n+1$ points (x_i, c_i) , pour $i = 0, \dots, n$, de \mathbb{R}^2 . On cherche à construire une courbe qui, dans un sens à préciser, s'approche des points (x_i, c_i) , ces points pouvant provenir de mesures, par exemple, et donc contenir des erreurs expérimentales.

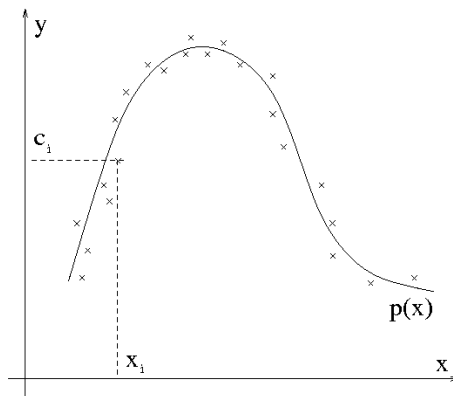


Figure 5.1: Approximation de données par une fonction $p(x)$

On recherche donc une fonction d'approximation, encore appelée *modèle* et notée $p(x)$ qui va répondre à cette question. Ce modèle dépend de paramètres : par exemple, s'il correspond à une fonction polynôme d'un degré donné, les paramètres sont les coefficients du polynôme.

Le problème que l'on se pose consiste donc à rechercher le meilleur jeu de paramètres

de façon à ce que la courbe représentative du modèle passe “au plus prêt” des points de données.

La notion de meilleure proximité retenue ici est le **critère des moindres carrés**. Il consiste à rechercher le minimum de la fonction :

$$\sum_{i=0}^n (p(x_i) - c_i)^2$$

Le minimum est réalisé pour des valeurs particulières des paramètres qui correspondent donc à l’ajustement du modèle par rapport aux points de données.

5.2 Régression linéaire

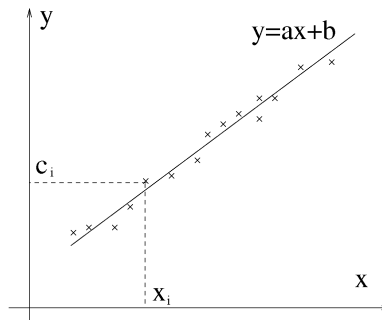


Figure 5.2: Régression linéaire

On recherche la droite d’équation $y = ax + b$ la plus proche de l’ensemble des points (x_i, c_i) , $i = 0, \dots, n$, au sens des moindres carrés. Si on note $e_i = c_i - (ax_i + b)$ l’écart entre la droite de régression et le point considéré, on cherche le minimum de

$$Q(a, b) = \sum_{i=0}^n e_i^2 = \sum_{i=0}^n (c_i - ax_i - b)^2$$

afin d’obtenir les meilleures valeurs des paramètres a et b .

Le minimum est obtenu si

$$\frac{\partial Q}{\partial a} = 0 \quad \text{et} \quad \frac{\partial Q}{\partial b} = 0$$

Il faut donc résoudre le système linéaire suivant d’ordre 2 :

$$\begin{cases} \frac{\partial Q}{\partial a} = -2 \sum_{i=0}^n x_i (c_i - ax_i - b) = 0 \\ \frac{\partial Q}{\partial b} = -2 \sum_{i=0}^n (c_i - ax_i - b) = 0 \end{cases}$$

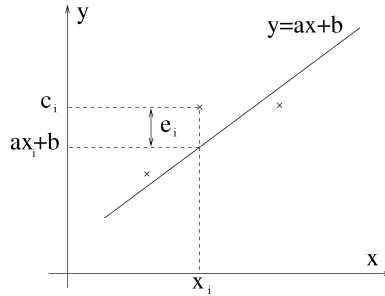


Figure 5.3: Ecart entre un point et une droite de régression

C'est à dire

$$\begin{cases} a \sum_{i=0}^n x_i^2 + b \sum_{i=0}^n x_i = \sum_{i=0}^n x_i c_i \\ a \sum_{i=0}^n x_i + (n+1)b = \sum_{i=0}^n c_i \end{cases}$$

de la dernière équation, on en déduit que

$$b = \bar{c} - a\bar{x}$$

où l'on note \bar{x} , la moyenne des x_i , à savoir $\frac{1}{n+1} \sum_{i=0}^n x_i$.

En substituant dans la première équation, on obtient

$$a \sum_{i=0}^n x_i^2 + (\bar{c} - a\bar{x}) \sum_{i=0}^n x_i = \sum_{i=0}^n x_i c_i$$

En utilisant la variance

$$V_x = \frac{1}{n+1} \left(\sum_{i=0}^n (x_i - \bar{x})^2 \right) = \frac{1}{n+1} \sum_{i=0}^n x_i^2 - \bar{x}^2$$

on obtient :

$$a(n+1)V_x = \sum_{i=0}^n x_i c_i - (n+1)\bar{x}\bar{c}$$

Et donc la valeur de a vaut :

$$a = \frac{\sum_{i=0}^n x_i c_i - (n+1)\bar{x}\bar{c}}{(n+1)V_x}$$

Il reste alors à déterminer la pertinence du calcul de régression linéaire par rapport aux données. Pour cela, on définit le *coefficient de corrélation linéaire* :

$$R = \frac{\sum_{i=0}^n x_i c_i - (n+1)\bar{x}\bar{c}}{(n+1)\sqrt{V_x V_c}}$$

- On montre que $-1 \leq R \leq 1$
- Si $|R|$ est voisin de 0 alors il n'y a pas de liaison linéaire entre les 2 coordonnées des points de données et le calcul de régression linéaire n'est pas représentatif de ces données ;
- Si $|R|$ est voisin de 1 alors les points de données sont proches d'un alignement et le calcul de régression se justifie.

5.3 Généralisation aux modèles linéaires

Nous allons étendre le calcul de régression linéaire à un modèle plus général du type :

$$p(x) = \sum_{j=0}^m \alpha_j \phi_j(x)$$

à partir des données $(x_i, c_i), 0 \leq i \leq n$

Les fonction ϕ_j sont connues alors que les coefficients α_j sont les paramètres à ajuster par rapport aux données.

Un tel modèle est appelé *linéaire* en raison de son expression qui est effectivement linéaire par rapport aux paramètres α_i .

On retrouve le modèle de régression linéaire en posant $m = 1$, $\phi_0(x) = 1$ et $\phi_1(x) = x$. L'expression du modèle est donc $p(x) = c_0 + c_1 x$.

Comme pour le calcul de régression linéaire, on applique le critère des moindres carrés qui consiste à rechercher le minimum de la fonction

$$\begin{aligned} S(\alpha_0, \alpha_1, \dots, \alpha_m) &= \sum_{i=0}^n (c_i - p(x_i))^2 \\ &= \sum_{i=0}^n \left(c_i - \sum_{j=0}^m \alpha_j \phi_j(x_i) \right)^2 \end{aligned}$$

Ce minimum est atteint lorsque les dérivées partielles de S par rapport à chaque coefficient $\alpha_k, k = 0 \dots m$, sont nulles :

$$\begin{aligned} \frac{\partial S(\alpha_0, \dots, \alpha_m)}{\partial \alpha_k} &= 0 \\ 2 \sum_{i=0}^n \left(c_i - \sum_{j=0}^m \alpha_j \phi_j(x_i) \right) (-\phi_k(x_i)) &= 0 \\ \sum_{i=0}^n \sum_{j=0}^m \alpha_j \phi_j(x_i) \phi_k(x_i) &= \sum_{i=0}^n c_i \phi_k(x_i) \\ \sum_{j=0}^m \alpha_j \sum_{i=0}^n \phi_k(x_i) \phi_j(x_i) &= \sum_{i=0}^n \phi_k(x_i) c_i \end{aligned}$$

C'est une équation linéaire à $m + 1$ inconnues : $\alpha_j, j = 0, \dots, m$ qui se décline pour chaque valeur de $k, k = 0, \dots, m$. On doit donc résoudre un système d'ordre $m + 1$.

5.3.1 Ecriture matricielle du problème

On pose F matrice d'ordre $(n + 1, m + 1)$ dont les coefficients valent :

$$F_{ij} = \phi_j(x_i)$$

et on pose Y le vecteur de composantes (c_0, \dots, c_n) .

On peut montrer que le système précédent s'écrit :

$$(F^t F) X = (F^t Y)$$

La solution X correspond aux $(m + 1)$ coefficients α_i cherchés. En effet

$$\begin{aligned} (F^t Y)_k &= \sum_{i=0}^n (F^t)_{ki} Y_i = \sum_{i=0}^n \phi_k(x_i) c_i \\ ((F^t F) X)_k &= \sum_{j=0}^m (F^t F)_{kj} X_j = \sum_{j=0}^m \left(\sum_{i=0}^n \phi_k(x_i) \phi_j(x_i) \right) \alpha_j \end{aligned}$$

On a donc bien, pour $0 \leq k \leq m$:

$$((F^t F) X)_k = (F^t Y)_k$$

Chapter 6

Dérivation et intégration numérique

Soit une fonction f définie sur $[a, b]$ et à valeur dans \mathbb{R} , “assez régulière” (c’est à dire au moins de classe C^1 , mais on supposera dans la suite qu’elle pourra être de classe C^k , pour $k > 1$). On s’intéresse au calcul des expressions suivantes :

$$I(f)_{a,b} = \int_a^b f(x)dx$$

$$D(f)_x = f'(x)$$

On utilise des méthodes numériques approchées pour calculer ces intégrales ou dérivées, lorsque f n’est pas connue de manière analytique ; elle peut être le résultat d’une procédure et donc difficile à intégrer ou dériver ; f peut n’être connue qu’à un nombre fini de points $\{x_i; 0 \leq i \leq n\}$.

Dans le chapitre 2, on a vu des calculs d’interpolations polynômiales, décomposant la fonction f en la somme d’un polynôme et d’un “reste” :

$$f(x) = P_n(x) + E_n(x)$$

On peut alors approcher l’intégrale ou la dérivée de f en la substituant à $P_n(x)$.

6.1 Dérivation numérique

On suppose ici que l’on connaît la fonction f en $(n + 1)$ points équidistants : $\{x_i; 0 \leq i \leq n\}$, avec $x_{i+1} - x_i = h$, pour h une constante donnée.

6.1.1 Approximation de la dérivée première

On utilise alors le développement en série de Taylor de $f(x+h)$:

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \dots$$

D'où, en isolant $f'(x)$ de la dernière équation :

$$f'(x) = \frac{f(x+h) - f(x)}{h} - \frac{h}{2}f''(x) + \dots$$

et donc

$$f'(x) = \frac{f(x+h) - f(x)}{h} + O(h)$$

Soit en $x = x_i$

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{h} + O(h)$$

Ce qui donne l'approximation d'ordre 1 (car le reste est en $O(h^1)$)

$$f'(x_i) \simeq \frac{f(x_{i+1}) - f(x_i)}{h}$$

C'est une formule décentrée à droite.

Cherchons à établir une formule *centrée*. On commence par écrire les deux développements limités "standards" de $f(x+h)$ et $f(x-h)$:

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{3!}f^{(3)}(x) + \dots \quad (6.1)$$

$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2}f''(x) - \frac{h^3}{3!}f^{(3)}(x) + \dots \quad (6.2)$$

En faisant (6.1)-(6.2), on obtient :

$$f(x+h) - f(x-h) = 2hf'(x) + 2\frac{h^3}{3!}f^{(3)}(\xi)$$

pour $\xi \in [a, b]$, en application du théorème des valeurs intermédiaires.

D'où

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} - \frac{h^2}{3!}f^{(3)}(\xi)$$

Soit une formule d'approximation d'ordre 2 :

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + O(h^2)$$

C'est à dire, reporter sur un ensemble de points équidistants :

$$f'(x_i) \simeq \frac{f(x_{i+1}) - f(x_{i-1}))}{2h}$$

On peut ainsi construire différentes formules en utilisant différentes formes de développements limités, par exemple :

- une formule décentrée à 3 points, d'ordre 2 :
- une formule centrée à 5 points d'ordre 4 :

La plus utilisées dans le pratique étant la formule centrée d'ordre 2, lorsque la précision qu'elle apporte est suffisante.

6.1.2 Approximation de la dérivée seconde

On reprend les formules des développements limités de $f(x + h)$ et $f(x - h)$, en y ajoutant un terme de plus :

$$f(x + h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{3!}f^{(3)}(x) + \frac{h^4}{4!}f^{(4)}(x) + \dots (6.3)$$

$$f(x - h) = f(x) - hf'(x) + \frac{h^2}{2}f''(x) - \frac{h^3}{3!}f^{(3)}(x) + \frac{h^4}{4!}f^{(4)}(x) + \dots (6.4)$$

On fait ici (6.3)+(6.4), on obtient :

$$f(x + h) + f(x - h) = 2f(x) + h^2f''(x) + \frac{h^4}{12}f^{(4)}(\xi)$$

pour $\xi \in [a, b]$, en application du théorème des valeurs intermédiaires.

D'où

$$f''(x) = \frac{f(x + h) - 2f(x) + f(x - h)}{h^2} - \frac{h^2}{12}f^{(4)}(\xi)$$

Soit une formule d'approximation d'ordre 2 :

$$f''(x) = \frac{f(x + h) - 2f(x) + f(x - h)}{h^2} + O(h^2)$$

C'est à dire, reporter sur un ensemble de points équidistants :

$$f''(x_i) \simeq \frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1}))}{h^2}$$

On peut aussi trouver des formules décentrées, comme par exemple :

6.2 Intégration numérique

Soit une fonction f définie sur $[a, b]$ et à valeur dans \mathbb{R} , “assez régulière” (c’est à dire au moins de classe C^1 , mais on supposera dans la suite qu’elle pourra être de classe C^k , pour $k > 1$). On s’intéresse au calcul de :

$$I(f)_{a,b} = \int_a^b f(x)dx$$

On découpe l’intervalle $[a, b]$ en un certain nombre de points $\{x_i; 0 \leq i \leq n\}$ tels que $a = x_0$ et $b = x_n$ et que l’on suppose régulièrement espacés : c’est à dire qu’il existe $h \in \mathbb{R}$, tel que $\forall i, x_{i+1} - x_i = h$. On suppose alors que f est connue en chacun de ces points x_i .

Le fait de prendre les points x_i régulièrement espacés permet de simplifier les formules. On peut facilement les généraliser si ces points ne sont pas régulièrement espacés.

On propose donc de calculer une approximation de l’intégrale de f sur l’intervalle $[x_0, x_n]$:

$$I = \int_{x_0}^{x_n} f(x)dx$$

figure 1

Rappelons que I s’interprète géométriquement comme l’aire située entre la courbe et l’axe des abscisses.

6.2.1 La méthode des trapèzes

Formule de calcul

On remplace la courbe de la fonction f que l’on ne connaît qu’aux points $\{x_i; 0 \leq i \leq n\}$, par une ligne brisée joignant tous ces points;

figure 2

L’intégrale est alors approchée par la somme des aires des n trapèzes de largeur h . On calcule alors l’aire d’un de ces trapèzes, le premier par exemple, par la formule suivante :

figure 3

$$\int_{x_0}^{x_1} f(x)dx \simeq h \frac{f_0 + f_1}{2}$$

En notant $f_i = f(x_i)$.

L'aire totale de I est donc égale à :

$$\begin{aligned} I = \int_{x_0}^{x_n} f(x)dx &= \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} f(x)dx \\ &\simeq \sum_{i=0}^{n-1} h \frac{f_i + f_{i+1}}{2} = \frac{h}{2} \left[\sum_{i=0}^{n-1} f_i + \sum_{i=0}^{n-1} f_{i+1} \right] \\ &= \frac{h}{2} \left[f_0 + \sum_{i=1}^{n-1} f_i + \sum_{i=0}^{n-2} f_{i+1} + f_n \right] \\ &= \frac{h}{2} \left[f_0 + \sum_{i=1}^{n-1} f_i + \sum_{i=1}^{n-1} f_i + f_n \right] \end{aligned}$$

D'où la fomule finale de calcul d'approximation de l'intégrame par la méthode des trapèzes :

$$I \simeq \frac{h}{2} \left[f_0 + 2 \sum_{i=1}^{n-1} f_i + f_n \right]$$

Erreur de la méthode des trapèzes

On pose

$$R(h) = \int_{x_0}^{x_n} f(x)dx - \frac{h}{2} \left[f_0 + 2 \sum_{i=1}^{n-1} f_i + f_n \right]$$

Si f est dérivable à l'ordre 2, de dérivée continue alors on a le résultat suivant (admis ici) :

$$R(h) = -\frac{(b-a)}{12} h^2 f''(\theta)$$

avec $\theta \in [a, b]$.

Cette formule est difficilement utilisable sous cette forme puisqu'on ne sait pas construire θ . On l'utilisera plutôt sous la forme suivante qui suppose que l'on sait calculer le sup de la dérivée seconde de f sur l'intervalle $[a, b]$:

$$R(h) \leq -\frac{(b-a)}{12} h^2 \sup_{\theta \in [a, b]} f''(\theta)$$

Remarques :

La formule reste toujours peu utilisable car souvent on ne connaît ni f , ni f'' de manière continue. Par contre, cette formule fournit des renseignements utiles :

- C'est une méthode d'ordre 1 (on l'appelle ainsi car l'erreur est en h^2)
- L'erreur dépend de f'' , c'est à dire de la variation de la dérivée de f . En effet si celle-ci varie beaucoup et notamment entre deux points successifs alors le calcul approximatif est forcément mauvais car il ne suit pas l'évolution de la courbe.

figure 4

6.2.2 La méthode de Simpson

Formule de calcul

On cherche à améliorer la méthode précédente en regroupant 2 à 2 les subdivisions (donc n doit être pair) et en faisant passer une parabole par les 3 points qui constituent les extrémités de 2 subdivisions successives.

Effectuons ainsi le calcul approché de $I_1 = \int_{x_0}^{x_2} f(x) dx$

figure 5

L'intervalle étant centré en x_1 , on recherche la parabole passant par les 3 points $(x_1 - h, f_0)$, (x_1, f_1) et $(x_1 + h, f_2)$, sous la forme

$$p(x) = a(x - x_1)^2 + b(x - x_1) + c$$

La parabole est caractérisée par les 3 coefficients a , b et c que l'on détermine après. Nous commençons par évaluer I_1 , à l'aide de ces coefficients :

$$\begin{aligned} I_1 &\simeq \int_{x_1-h}^{x_1+h} f(x) dx = \int_{x_1-h}^{x_1+h} (a(x - x_1)^2 + b(x - x_1) + c) dx \\ &= \left[\frac{a}{3}(x - x_1)^3 + \frac{b}{2}(x - x_1)^2 + c(x - x_1) \right]_{x_1-h}^{x_1+h} \\ &= \left(\frac{a}{3}h^3 + \frac{b}{2}h^2 + ch \right) - \left(\frac{a}{3}(-h)^3 + \frac{b}{2}(-h)^2 + c(-h) \right) \\ &= \frac{2a}{3}h^3 + 2ch \end{aligned}$$

Déterminons maintenant a et c . On a :

$$\begin{cases} p(x_1 - h) = f_0 \\ p(x_1) = f_1 \\ p(x_1 + h) = f_2 \end{cases}$$

Ce qui donne

$$\begin{cases} ah^2 - bh + c = f_0 \\ c = f_1 \\ ah^2 + bh + c = f_2 \end{cases}$$

La deuxième équation donne directement c .

En additionnant la première et la troisième équation, on obtient :

$$2ah^2 + 2c = f_0 + f_2$$

soit

$$a = \frac{f_0 - 2c + f_2}{2h^2} = \frac{f_0 - 2f_1 + f_2}{2h^2}$$

D'où

$$\begin{aligned} I_1 &\simeq \frac{2h^3}{3} \left(\frac{f_0 - 2f_1 + f_2}{2h^2} \right) + 2hf_1 \\ &= \frac{h}{3} (f_0 + 4f_1 + f_2) \end{aligned}$$

En cumulant sur tous les intervalles, on a

$$I = \int_{x_0}^{x_n} f(x)dx \simeq \frac{h}{3} [(f_0 + 4f_1 + f_2) + (f_2 + 4f_3 + f_4) + \dots + (f_{n-2} + 4f_{n-1} + f_n)]$$

On pose $m = \frac{n}{2}$ et on a finalement la formule de Simpson :

$$I \simeq \left(f_0 + f_{2m} + 4 \sum_{i=0}^{m-1} f_{2i+1} + 2 \sum_{i=1}^{m-1} f_{2i} \right)$$

Erreur de la méthode de Simpson

Si f est dérivable à l'ordre 4, de dérivée continue alors on a le résultat suivant (admis ici) :

$$R(h) = -\frac{(b-a)}{180} h^4 f^{(4)}(\theta)$$

avec $\theta \in [a, b]$.

Remarque :

- La méthode de Simpson est plus précise que la méthode des trapèzes car l'erreur est en h^4 (au lieu d'être en h^2 dans la méthode des trapèzes) ... h est supposé petit !
- L'erreur dépend aussi des variations des dérivées successives.

Remarque sur la précision des méthodes des trapèzes et Simpson

Deux types d'erreurs vont intervenir dans les calculs d'intégrales :

- Une erreur due à la méthode et à l'approximation effectuée de la fonction à intégrer (segments de droite ou de paraboles pour le méthode des trapèzes ou de Simpson, respectivement) ;
- Une erreur d'arrondi due à la représentation des nombres réels en machine avec un nombre fini de décimales : plus on effectue de calcul, plus les erreurs s'accumulent.

A priori, on peut penser que si on diminue h , alors on suivra de plus près la courbe (ou les variations de sa dérivées) et donc l'erreur due à la méthode va diminuer. Par contre, on augmente le nombre de calculs pour un même intervalle $[a, b]$ et par conséquent le cumul des erreurs d'arrondis va augmenter.

figure 6

Exercice d'application

6.2.3 Méthode de Newton-Cote

6.2.4 Méthode de Gauss

Dans les méthodes précédentes, on partait des points d'abscisse x_i , que l'on pouvait supposés être régulièrement espacés. Ici, dans la méthode de Gauss, on adopte une approche très différente. Les abscisses des points x_i seront calculés "au mieux" pour obtenir des formules - appelées *formules de quadrature* - respectant une précision donnée.

Ces formules de quadrature sont recherchées sous la forme

$$\int_a^b f(x)dx \simeq \sum_{k=1}^n A_k f(x_k)$$

ici les n coefficients A_k et les n points x_k sont à déterminer.

Exemple

On recherche la formule de quadrature suivante

$$\int_{-1}^1 f(x)dx \simeq af(x_1) + bf(x_2)$$

On recherche les valeurs que peuvent prendre a , b , x_1 et x_2 pour que la formule soit vraie pour tout polynôme de degré 3.

Pour cela, on applique la formule de quadrature pour les 4 fonctions de base des polynômes de degré 3, à savoir $f(x) = x^3$, $f(x) = x^2$, $f(x) = x$ et $f(x) = 1$ que l'on sait intégrer sur l'intervalle considéré. Dans la suite, on effectue d'abord ce calcul d'intégrale puis on l'identifie avec la formule de quadrature :

$$\text{pour } f(x) = x^3, \text{ on a } \int_{-1}^1 x^3 dx = 0 = ax_1^3 + bx_2^3 \quad (6.5)$$

$$\text{pour } f(x) = x^2, \text{ on a } \int_{-1}^1 x^2 dx = \frac{2}{3} = ax_1^2 + bx_2^2 \quad (6.6)$$

$$\text{pour } f(x) = x, \text{ on a } \int_{-1}^1 x dx = 0 = ax_1 + bx_2 \quad (6.7)$$

$$\text{pour } f(x) = 1, \text{ on a } \int_{-1}^1 1 dx = 2 = a + b \quad (6.8)$$

Si l'on effectue maintenant la combinaison linéaire $(1) - (3) \cdot x_1^2$, on obtient :

$$\begin{aligned} bx_2^3 - bx_2x_1^2 &= 0 \\ b(x_2)(x_2^2 - x_1^2) &= 0 \\ b(x_2)(x_2 - x_1)(x_2 + x_1) &= 0 \end{aligned}$$

Ce produit est donc nul si

$$\text{soit } b = 0$$

$$\text{soit } x_2 = 0$$

$$\text{soit } x_1 = x_2$$

$$\text{soit } x_1 = -x_2$$

On vérifie facilement que les 3 premières conditions conduisent à des incompatibilités, une fois reportées dans les équations (1) à (4).

Seul la dernière condition est réalisable et on obtient alors, en reportant dans les équations (1) à (4), tous calculs effectués :

$$a = b = 1$$

$$x_1 = -x_2 = -\sqrt{\frac{1}{3}} = 0.5773$$

Cas général

On recherche donc une formule de quadrature sous la forme générale :

$$\int_a^b f(x)dx \simeq \sum_{k=1}^n A_k f(x_k)$$

On montre d'abord, qu'en utilisant un changement de variables, on peut se ramener à la seule étude de l'intégrale $f(x)$ sur l'intervalle $[-1; 1]$:

$$x = \frac{(b-a)t + (b+a)}{2}$$

$$dx = \left(\frac{b-a}{2}\right) dt$$

$$\int_a^b f(x)dx = \frac{b-a}{2} \int_{-1}^1 f\left(\frac{(b-a)t + (b+a)}{2}\right) dt$$

le problème se ramène alors à rechercher la formule de quadrature

$$\int_{-1}^1 f(x)dx \simeq \sum_{k=1}^n A_k f(x_k)$$

dans laquelle on recherche n points x_k et n coefficients A_k .

Pour pouvoir les déterminer, on devra alors rechercher une formule exacte pour des polynômes de degré $2n - 1$ au moins, sinon le nombre de conditions exprimées

n'est pas suffisant pour obtenir les inconnues recherchées.

En intégrant toutes les fonctions polynômes : $f(x) = x^j$, comme dans l'exemple précédent, on obtient alors un système de $2n$ équations **non linéaires** qui sont exactement :

$$A_1 x_1^j + \dots + A_n x_n^j = \begin{cases} 0 & \text{pour } j \text{ impair} \\ \frac{2}{j+1} & \text{pour } j \text{ pair} \end{cases}$$

Il est alors possible de trouver les $2n$ inconnues : $A_1, \dots, A_n, x_1, \dots, x_n$ mais pas de manière simple puisque le système n'est pas linéaire.

On utilise alors une autre méthode de calcul qui est la suivante. On peut montrer que les x_i cherchés sont les racines du polynôme de Legendre de degré n .

Les polynômes de Legendre sont définis récursivement par :

$$\begin{aligned} L_0(x) &= 1 \\ L_1(x) &= x \\ (n+1)L_{n+1}(x) - (2n+1)xL_n(x) + nL_{n-1}(x) &= 0 \end{aligned}$$

Les 6 premiers polynômes de Legendre sont représentés sur la figure 6.1.

Remarque : Les polynômes de Legendre ont des propriétés remarquables, notamment ils sont orthogonaux au sens suivant :

$$\int_{-1}^1 L_n(x)L_m(x)dx = \begin{cases} 0 & \text{si } n \neq m \\ > 0 & \text{si } n = m \end{cases}$$

Une fois connues les racines de ces polynômes, on peut, grâce au système précédent, calculer les valeurs des A_k associées.

Dans la pratique, on utilise des tables des racines et des poids A_k associés puisque nous nous sommes ramenés à une unique formule de quadrature par changement de variable :

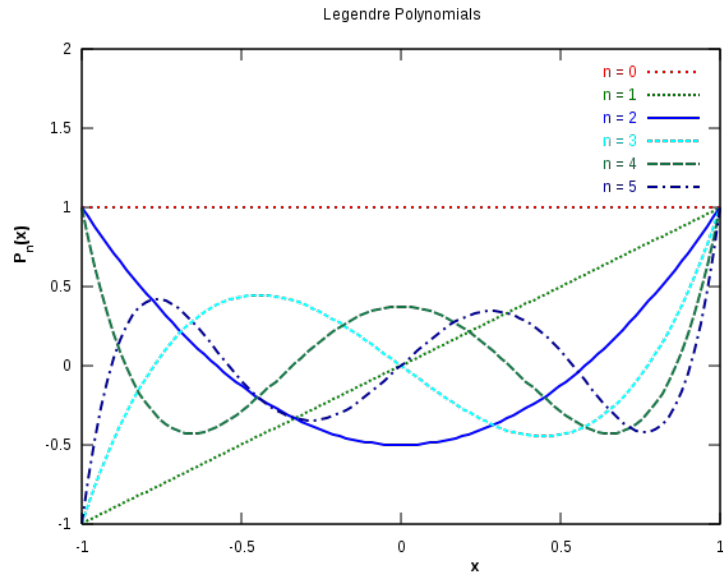


Figure 6.1: Tracé des 6 premiers polynômes de Legendre, de L_0 à L_5 .

Degré du polynôme	racines	poids associé
2	-0.5773	1
	0.5773	1
3	-0.7746	0.555
	0	0.889
	0.7746	0.555
4	-0.8611	0.3479
	-0.3400	0.6521
	0.3400	0.6521
	0.8611	0.3479

Exemple de calcul

Soit à calculer

$$I = \int_{0.2}^{1.5} e^{-x^2} dx$$

On effectue le changement de variable, en appliquant les formules précédentes :

$$x = \frac{(1.5 - 0.2)t + (1.5 + 0.2)}{2} = 0.65t + 0.85$$

On a alors, en prenant $n = 3$ et en appliquant les valeurs du tableau précédent :

$$\begin{aligned}
 I &= \frac{1.5 - 0.2}{2} \int_{-1}^1 e^{-(0.65t+0.85)^2} dt \\
 &= 0.65 \left[0.555e^{-(0.65(-0.7746)+0.85)^2} + 0.889e^{-(0.65(0)+0.85)^2} + 0.555e^{-(0.65(0.7746)+0.85)^2} \right] \\
 &= 0.6586
 \end{aligned}$$

La valeur exacte est 0.6588.

Autres méthodes

- Méthode de Romberg : voir TD
- Méthode adaptative : voir TP ... numéro 12

6.2.5 Intégrales multiples (introduction)

On ne donne ici qu'un exemple simple permettant de comprendre comment on peut généraliser les méthodes numériques à 1 dimension pour les appliquer à des calculs pour plusieurs dimensions.

On généralise ici la méthode de Simpson en dimension 2.

On connaît un ensemble de points $\{z_{ij} = f(x_i, y_j), 0 \leq i \leq n, 0 \leq j \leq m\}$ que l'on suppose régulièrement définis sur chacune des deux dimensions. Il existe donc h et k constants tels que $h = x_{i+1} - x_i$ et $k = y_{j+1} - y_j$. On recherche l'intégrale sur un "élément simple" :

$$A = \int_{x_0}^{x_2} \int_{y_0}^{y_2} f(x, y) dx dy$$

On applique alors la formule de Simpson successivement sur les 2 dimensions, c'est à dire :

$$\begin{aligned}
 A &= \int_{x_0}^{x_2} \left(\frac{k}{3} (f(x, y_0) + 4f(x, y_1) + f(x, y_2)) \right) dx \\
 &= \frac{k}{3} \left(\int_{x_0}^{x_2} f(x, y_0) dx + 4 \int_{x_0}^{x_2} f(x, y_1) dx + \int_{x_0}^{x_2} f(x, y_2) dx \right) \\
 &= \frac{k}{3} \left(\frac{h}{3} (f(x_0, y_0) + 4f(x_1, y_0) + f(x_2, y_0)) + \right. \\
 &\quad \left. 4 \frac{h}{3} (f(x_0, y_1) + 4f(x_1, y_1) + f(x_2, y_1)) + \right. \\
 &\quad \left. \frac{h}{3} (f(x_0, y_2) + 4f(x_1, y_2) + f(x_2, y_2)) \right) \\
 &= \frac{hk}{9} (z_{00} + z_{20} + z_{02} + z_{22} + 4(z_{10} + z_{01} + z_{21} + z_{12}) + 16z_{11})
 \end{aligned}$$

Chapter 7

Résolution d'équations/systemes différentiels

Chapter 8

Annexe

Rappels et compléments d'algèbre linéaire

Sommaire

8.1	Quelques structures algébriques	62
8.2	Espace vectoriel	64
8.3	Applications linéaires et matrices	64
8.4	Matrice inverse d'une matrice carrée	66
8.5	Calcul du déterminant d'une matrice carrée	66
8.6	Trace d'une matrice carrée	67
8.7	Matrices transposées, orthogonales, hermitiennes, unitaires et normales	67
8.8	Valeurs propres et vecteurs propres de matrice	68
8.9	Matrices hermitiennes définies positives et matrices à diag- onale dominante	69
8.10	Normes vectorielles et matricielles	70

8.1 Quelques structures algébriques

Définition 8.1 (Groupe). G muni d'une loi de composition interne $+$ est un groupe si

- $+$ est associative
- il existe un élément neutre dans $(G, +)$
- tout élément de G a un symétrique pour $+$

Rappels :

- La loi $+$ est une LCI sur $G : \forall a, b \in G \quad a + b \in G$
- La loi $+$ est associative sur $G : \forall a, b, c \in G \quad (a + b) + c = a + (b + c)$
- e est élément neutre de $+$: $\forall a \in G \quad a + e = e + a = a$
- a' est symétrique de a pour $+$: $a + a' = a' + a = e$

Définition 8.2 (Anneau). A muni de deux LCI, notée $+$ et $*$, est un anneau si

- $(A, +)$ est un groupe commutatif
- $*$ est associative
- $*$ est distributive sur $+$

Rappel et complément :

- La loi $*$ est distributive sur la loi $+$: $\forall a, b, c \in A \quad a * (b + c) = a * b + a * c$
- Un anneau est commutatif si $*$ est commutative
- Un anneau est unitaire si $*$ possède un élément neutre

Définition 8.3 (Corps). K muni de deux LCI, notée $+$ et $*$, est un corps si

- $(K, +, *)$ est un anneau
- $(K^*, *)$ est un groupe, où $K^* = K - \{0\}$, 0 élément neutre de $+$

Complément :

- Le corps est commutatif si $*$ est commutative

Définition 8.4 (Espace Vectoriel). E est un espace vectoriel sur un corps commutatif $(K, +, *)$, d'éléments neutres 0 et 1 si

- E possède une LCI notée $+$ telle que $(E, +)$ soit un groupe commutatif
- E possède une loi de composition externe, notée \cdot , définie de $(K, E) \rightarrow E$ et qui vérifie $\forall x, y \in E \quad \forall \lambda, \mu \in K$
 - $\lambda \cdot (\mu \cdot x) = (\lambda * \mu) \cdot x$
 - $1 \cdot x = x$
 - $(\lambda + \mu) \cdot x = \lambda \cdot x + \mu \cdot x$
 - $\lambda \cdot (x + y) = \lambda \cdot x + \lambda \cdot y$

8.2 Espace vectoriel

Un espace vectoriel E sur un corps commutatif K (en général \mathbb{R} ou \mathbb{C}) possède

- une loi de composition interne (LCI), notée $+$, $: (E \times E \rightarrow E)$
- une loi de composition externe (LCE), notée $.$, $: (K \times E \rightarrow E)$

Elles ont les propriétés rappelées dans le polycopié de rappels.

Définition 8.5. Une base B d'un espace vectoriel E sur un corps K est un ensemble d'éléments de E , $B = (e_1, \dots, e_n)$, tels que

$$\forall x \in E, \exists!^1(x_1, \dots, x_n) \in K^n, x = \sum_{i=1}^n x_i e_i$$

E est dit de dimension n , et on note $E = K^n$.

x se note $\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = (x_1, \dots, x_n)^t$.

Opérations sur les vecteurs :

- On définit le produit scalaire de deux vecteurs u et v de K^n par

$$u.v = \sum_{i=1}^n u_i v_i \in K$$

avec $u = \begin{pmatrix} u_1 \\ \vdots \\ u_n \end{pmatrix}$ et $v = \begin{pmatrix} v_1 \\ \vdots \\ v_n \end{pmatrix}$.

- On définit le produit vectoriel de deux vecteurs u et v de K^3 par

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} \wedge \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} u_2 v_3 - u_3 v_2 \\ u_3 v_1 - u_1 v_3 \\ u_1 v_2 - u_2 v_1 \end{pmatrix} \in K^3$$

- On définit le produit mixte de trois vecteurs $u, v, w \in K^3$ par

$$(u, v, w) = (u \wedge v).w \in K$$

8.3 Applications linéaires et matrices

Définition 8.6. Soit $f : K^n \rightarrow K^m$ est une application linéaire si $\forall \alpha, \beta \in K, \forall x, y \in K^n$,

$$f(\alpha x + \beta y) = \alpha f(x) + \beta f(y)$$

¹Il existe de manière unique.

Soit $B = \{e_j; 1 \leq j \leq n\}$ une base de K^n et
soit $C = \{f_i; 1 \leq i \leq m\}$ une base de K^m .
 $\forall j$ tel que $1 \leq j \leq n$,

$$f(e_j) = \sum_{i=1}^m a_{i,j} f_i$$

On peut ranger tous les $a_{i,j}$ dans un tableau :

$$A = \begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ a_{2,1} & \cdots & a_{2,n} \\ \vdots & & \vdots \\ a_{m,1} & \cdots & a_{m,n} \end{bmatrix} \in M_{m,n}(K)$$

où $M_{m,n}(K)$ est l'ensemble des matrices à m lignes et n colonnes à éléments dans K . Ainsi, pour tout élément x de K^n , on a $x = \sum_{j=1}^n x_j e_j$, et

$$f(x) = f\left(\sum_{j=1}^n x_j e_j\right) = \sum_{j=1}^n x_j f(e_j)$$

La matrice A caractérise complètement l'application linéaire f , et elle permet de calculer les transformations par f de tous les vecteurs de K^n . En effet, pour $x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \in K^n$ et $y = \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix} \in K^m$, on a

$$\begin{aligned} y &= f(x) \\ y_i &= \sum_{j=1}^n a_{i,j} x_j \\ y &= Ax \end{aligned}$$

Le produit $Ax = y$ étant :

$$\begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ a_{2,1} & \cdots & a_{2,n} \\ \vdots & \cdots & \vdots \\ a_{m,1} & \cdots & a_{m,n} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} a_{1,1}x_1 + \cdots + a_{1,n}x_n \\ \vdots \\ a_{m,1}x_1 + \cdots + a_{m,n}x_n \end{bmatrix}$$

Opérations élémentaires sur les matrices :

- L'addition : $A + B = C$
si $A, B \in M_{n,m}(K)$, $C_{i,j} = A_{i,j} + B_{i,j}$, $\forall i, j$
- La multiplication par un scalaire de K : $\alpha A = C$,
 $C_{i,j} = \alpha A_{i,j}$, $\forall i, j$.

- Le produit : $AB = C$,
si $A \in M_{m,n}(K)$, $B \in M_{n,p}(K)$, $C_{i,j} = \sum_{k=1}^n a_{i,k}b_{k,j}$.
Si A est associée à l'application linéaire f , et B associée à l'application linéaire g , alors AB est associée à $f \circ g$. Donc, si $z = f(g(x)) = (f \circ g)(x)$, alors $z = (AB)x = A(Bx)$.

Remarque 8.1.

- La matrice $\begin{pmatrix} 0 & \dots & 0 \\ \vdots & & \vdots \\ 0 & \dots & 0 \end{pmatrix}$ est élément neutre pour l'addition des matrices.
- La matrice $\text{Id} = \begin{pmatrix} 1 & \dots & 0 \\ \vdots & & \vdots \\ 0 & \dots & 1 \end{pmatrix}$ est élément neutre de la multiplication.

8.4 Matrice inverse d'une matrice carrée

Définition 8.7. $A \in M_{n,n}(K)$ est inversible (ou régulière) si $\exists B \in M_{n,n}(K)$ telle que

$$AB = BA = \text{Id}$$

Le produit matriciel n'étant pas commutatif, il faut vérifier les deux produits AB et BA . B est notée A^{-1} et associée à l'application linéaire inverse de celle associée à A .

8.5 Calcul du déterminant d'une matrice carrée

Par développement de ligne ou de colonne :

$$\begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,j} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,j} & \dots & a_{2,n} \\ \vdots & & \dots & & \dots & \vdots \\ a_{n,1} & a_{n,2} & \dots & a_{n,j} & \dots & a_{n,n} \end{pmatrix} \\ = (-1)^{1+j} a_{1,j} \times \det(A_{[-1;-j]}) + \dots + (-1)^{n+j} a_{n,j} \times \det(A_{[-n;-j]})$$

où $A_{[-i;-j]}$ est la matrice A sans la ligne i et sans la colonne j . Pour calculer un déterminant d'ordre n , $C(n) = nC(n-1) + a$ où a est une constante. D'où, $C(n) \geq n!$.

Propriété 8.1.

- $\det(\text{Id}) = 1$
- $\det(AB) = \det(A) \times \det(B)$
- $\det(\lambda A) = \lambda^n \det(A)$

- A est régulière (invertible) si et seulement si $\det(A) \neq 0$.
- Dans \mathbb{R}^3 , $(u, v, w) = \det(A)$ où A est la matrice dont les trois colonnes sont u, v et w .
- Si A est triangulaire, alors $\det A = \prod_{i=1}^n A_{i,i}$.

8.6 Trace d'une matrice carrée

Définition 8.8. La trace est une application

$\text{tr} : M_{n,n}(K) \rightarrow K$ définie par

$$A \mapsto \text{tr}(A) = \sum_{i=1}^n A_{i,i}.$$

Autrement dit, la trace d'une matrice carrée A d'ordre n est la somme de ses n coefficients diagonaux.

Propriété 8.2.

- $\text{tr}(\text{Id}) = n$ pour une matrice d'ordre n
- $\text{tr}(A + B) = \text{tr}(A) + \text{tr}(B)$
- $\text{tr}(\lambda A) = \lambda \text{tr}(A)$

8.7 Matrices transposées, orthogonales, hermitiennes, unitaires et normales

Définition 8.9. Soit $A \in M_{m,n}(K)$, on appelle transposée de A , notée $A^t \in M_{n,m}(K)$, définie par $(A^t)_{i,j} = A_{j,i}, \forall i, j$.

Remarque 8.2. On assimile $M_{n,1}(K)$ à K^n .

Définition 8.10. On dit que A est symétrique si $A^t = A$ (et donc $m = n$).

Propriété 8.3.

- $\det(A^t) = \det(A)$
- $(A + B)^t = A^t + B^t$
- $(AB)^t = B^t A^t$
- Si A est régulière (invertible), alors $(A^t)^{-1} = (A^{-1})^t$, et donc A^t est également invertible.

Définition 8.11. On dit que A est orthogonale si et seulement si $A^{-1} = A^t$.

Dans la suite de la section, on se place dans $K = \mathbb{C}$, l'ensemble des nombres complexes.

Définition 8.12. On appelle A^* , adjointe de A :

$$(A^*)_{i,j} = \overline{A_{j,i}}$$

Propriété 8.4.

- $\det(A^*) = \overline{\det(A)}$
- $(\lambda A + \mu B)^* = \overline{\lambda}A^* + \overline{\mu}B^*$, $\lambda, \mu \in \mathbb{C}$
- $(AB)^* = B^*A^*$
- Si A est régulière, A^* l'est aussi et

$$(A^*)^{-1} = (A^{-1})^*$$

Définition 8.13. $A \in M_{n,n}(\mathbb{C})$ est hermitienne si $A^* = A$. En conséquence, les éléments diagonaux d'une matrice hermitienne sont des réels, et les éléments symétriques par rapport à la diagonale sont des conjugués.

Définition 8.14. On dit que $A \in M_{n,n}(\mathbb{C})$ est unitaire si $A^{-1} = A^*$.

Définition 8.15. On dit que $A \in M_{n,n}(\mathbb{C})$ est normale si elle commute avec son adjointe A^* : $A^*A = AA^*$.

Remarque 8.3. Une matrice hermitienne ou unitaire est normale.

Théorème 8.1 (de factorisation d'une matrice normale). A est normale si et seulement s'il existe une matrice U unitaire telle que :

$$A = UDU^*$$

où D est la matrice diagonale des valeurs propres complexes de A .

8.8 Valeurs propres et vecteurs propres de matrice

Soit $A \in M_{n,n}(K)$.

Définition 8.16. $\lambda \in K$ est appelée valeur propre de A si :

$$\exists x \in K^n, x \neq 0, Ax = \lambda x$$

x est appelé le vecteur propre associé à la valeur propre λ .

Propriété 8.5 (caractéristique). λ est une valeur propre de A si et seulement si $\det(A - \lambda \text{Id}) = 0$.

On note $\det(A - \lambda \text{Id}) = P(\lambda)$ le polynôme caractéristique de A de degré n .

Remarque 8.4.

- Si $K = \mathbb{C}$, $P(\lambda) = 0$ est donc une équation de degré n , et possède donc n racines complexes (pas nécessairement disjointes), d'où n valeurs propres complexes.
- Si $K = \mathbb{R}$, on n'a pas forcément n valeurs propres réelles.

Définition 8.17. On appelle spectre de A , noté $sp(A)$, l'ensemble des valeurs propres de A .

On appelle rayon spectral $\rho(A) = \max_{\lambda_i \in sp(A)} |\lambda_i|$.

Rappel 8.1. Si $z = x + iy$, le module de z est $|z| = \sqrt{x^2 + y^2}$.

Propriété 8.6.

- $\det(A) = \prod_{i=1}^n \lambda_i$
- $\text{tr}(A) = \sum_{i=1}^n \lambda_i$
- $sp(A^t) = sp(A)$
- $\rho(A) \leq \max_{1 \leq i \leq n} \left(\sum_{j=1}^n |a_{i,j}| \right)$, et $\rho(A) \leq \max_{1 \leq j \leq n} \left(\sum_{i=1}^n |a_{i,j}| \right)$, qui sont deux normes matricielles.

Définition 8.18. $A \in M_{n,n}(K)$ est diagonalisable s'il existe $P \in M_{n,n}(K)$ inversible telle que :

$$\underbrace{PAP^{-1}}_{\text{chgt de base}} = D$$

où D est la matrice diagonale des valeurs propres de A , et où P est formé des vecteurs propres de A qui constituent ainsi une nouvelle base.

Si A est diagonalisable, ses vecteurs propres sont linéairement indépendants.

8.9 Matrices hermitiennes définies positives et matrices à diagonale dominante

Définition 8.19. Une matrice carrée $A \in M_{n,n}(\mathbb{C})$ hermitienne est dite positive si :

$$\forall x \in \mathbb{C}^n, x \neq 0 \text{ alors } x^*Ax > 0$$

Remarque 8.5. $x^*Ax \in \mathbb{C}$, et

$$\begin{aligned} (x^*Ax)^* &= x^*A^*x^* \\ &= x^*Ax \end{aligned}$$

donc $x^*Ax \in \mathbb{R}$, comparable à 0.

Définition 8.20. Une matrice $A \in M_{n,n}(K)$ est dite à diagonale dominante (resp. strictement dominante) si :

$$|a_{i,i}| \geq \sum_{\substack{j=1 \\ j \neq i}}^n |a_{i,j}|$$

resp.

$$|a_{i,i}| > \sum_{\substack{j=1 \\ j \neq i}}^n |a_{i,j}|$$

Propriété 8.7. Une matrice à diagonale strictement dominante est inversible.

8.10 Normes vectorielles et matricielles

Définition 8.21 (norme vectorielle). Une norme $N : \mathbb{C}^n \rightarrow \mathbb{R}^+$ vérifie les propriétés : $\forall x, y \in \mathbb{C}^n, \forall \lambda \in \mathbb{C}$,

1. $N(x) \geq 0$
2. $N(x) = 0 \iff x = 0$
3. $N(x + y) \leq N(x) + N(y)$ (inégalité triangulaire)
4. $N(\lambda x) = |\lambda|N(x)$

Définition 8.22. La norme de la convergence en moyenne N_1 est définie par :

$$N_1(x) = \|x\|_1 = \sum_{i=1}^n |x_i|$$

L'espace vectoriel normé par N_1 est appelé L_1 .

Définition 8.23. La norme euclidienne N_2 est définie par :

$$N_2(x) = \|x\|_2 = \left(\sum_{i=1}^n |x_i|^2 \right)^{1/2}$$

L'espace vectoriel normé par N_2 est appelé L_2 .

Définition 8.24. La norme de la convergence absolue N_∞ est définie par :

$$N_\infty(x) = \|x\|_\infty = \max_{1 \leq i \leq n} |x_i|$$

L'espace vectoriel normé par N_∞ est appelé L_∞ .

Définition 8.25. De manière générale, l'espace vectoriel L_p , $p \in \mathbb{N}$, est normé par N_p définie par :

$$N_p(x) = \left(\sum_{i=1}^n |x_i|^p \right)^{1/p}$$

Remarque 8.6. Dans \mathbb{R}^2 ou \mathbb{R}^3 :

$$u.v = \|u\|_2 \|v\|_2 \cos(\widehat{u,v})$$

$u.v$ est nul si $u = 0$, ou $v = 0$, ou si $\cos(\widehat{u,v}) = 0$, et u et v sont dits orthogonaux.
 $(\widehat{u,v}) = \pi/2 \pmod{\pi}$

Dans \mathbb{R}^3 ,

$$\|u \wedge v\|_2 = \|u\|_2 \|v\|_2 |\sin(\widehat{u,v})|$$

C'est l'aire du parallélogramme défini par u et v . $u \wedge v$ est perpendiculaire à u et à v car

$$(u \wedge v).u = (u \wedge v).v = (u, v, u) = (u, v, v) = 0$$

La définition précédente d'une norme vectorielle s'applique à tout espace vectoriel, donc à l'espace vectoriel des matrices.

Définition 8.26. Norme de Schurr sur $M_{n,n}(K)$

$$\|A\|_s = \left(\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2 \right)^{\frac{1}{2}}$$

On peut vérifier facilement que c'est bien une norme au sens de la définition 8.21.

Définition 8.27. Norme matricielle associée, subordonnée et induite par une norme vectorielle.

Soient $A \in M_{n,n}(K)$, N_m et N_n deux normes vectorielles sur K^m et K^n . On appelle norme matricielle induite, la norme :

$$\|A\| = \max_{N_n(x)=1} N_m(Ax)$$

Propriété 8.8.

1. Ce sont des normes au sens de la définition 8.21,
2. $\|A\| = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|}$
3. $\|Ax\| \leq \|A\| \cdot \|x\|, \forall x \in K^n$
4. $\|AB\| \leq \|A\| \cdot \|B\|$
5. $\|I\| = 1$

Définition 8.28. Norme matricielle associée à L_1 :

$$\|A\|_1 = \max_j \left(\sum_{i=1}^n |a_{ij}| \right)$$

Définition 8.29. Norme matricielle associée à L_2 :

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

$\rho(M)$: rayon spectral de la matrice M .

Définition 8.30. Norme matricielle associée à L_∞ :

$$\|A\|_\infty = \max_i \left(\sum_{j=1}^m |a_{ij}| \right)$$

Remarque 8.7. La norme de Shurr n'est pas une norme matricielle induite car

$$\|I\|_s = \sqrt{n}.$$

Si A est hermitienne, on peut montrer $\|A\|_2 = \rho(A)$.

... à compléter