

Analyse Numérique

Thomas Cluzeau

Maître de Conférences

École Nationale Supérieure d'Ingénieurs de Limoges
Parc ester technopole, 16 rue d'atlantis 87068 Limoges Cedex

thomas.cluzeau@unilim.fr

http://www.unilim.fr/pages_perso/thomas.cluzeau



- **Harmonisation** en fonction du test de la rentrée
 - Analyse
 - Algèbre linéaire
- **Tronc Commun (TC) - 1^{ière} année**
 - Mathématiques pour l'ingénieur (TC1 - S1)
 - Analyse numérique (TC1 - S2)

- **Organisation :**
 - Cours : 7 séances d'1h30
 - TDs et TPs : 12h
 - 4 séances de TDs d'1h30
 - 3 séances de TPs `MATLAB` : 1 de 3h et 2 d'1h30.
- **Évaluation :**
 - Note du TP de 3h (Compte rendu) - 1/4 note finale
 - 1 examen final de 1h30 avec documents - 3/4 note finale

- 1 Arithmétique des ordinateurs et analyse d'erreurs
- 2 Résolution d'un système d'équations linéaires (Partie 1) : méthodes directes
- 3 Conditionnement d'une matrice pour la résolution d'un système linéaire
- 4 Résolution d'un système d'équations linéaires (Partie 2) : méthodes itératives
- 5 Interpolation polynomiale
- 6 Intégration numérique
- 7 Résolution d'équations et de systèmes d'équations non linéaires

Chapitre 1

Arithmétique des ordinateurs et analyse d'erreurs

- Comment les réels sont-ils représentés dans un ordinateur ?

Théorème (Système des nombres à virgule flottante)

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

$$x = \text{sgn}(x) \beta^e \sum_{k \geq 1} \frac{d_k}{\beta^k},$$

où $\text{sgn}(x) \in \{+, -\}$ est le signe de x , les d_k sont des entiers tels que $0 < d_1 \leq \beta - 1$ et $0 \leq d_k \leq \beta - 1$ pour $k \geq 2$, et $e \in \mathbb{Z}$. De plus, cette écriture est unique (sauf pour les décimaux : $2,5 = 2,499999\dots$).

Exemples

- **Système décimal** : $\beta = 10$ et $d_k \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
 - $0,0038 = 0,38 \cdot 10^{-2} = +10^{-2} \left(\frac{3}{10} + \frac{8}{10^2} \right)$
 - $\frac{1}{7} = 0,142857\dots = +10^0 \left(\frac{1}{10} + \frac{4}{10^2} + \frac{2}{10^3} + \frac{8}{10^4} + \dots \right)$.
Développement décimal d'un nombre rationnel est périodique :
 $\frac{1}{7} = 0, \mathbf{142857}142857142857\dots$
 - $-\sqrt{2} = -1,4142\dots = -10^1 \left(\frac{1}{10} + \frac{4}{10^2} + \frac{1}{10^3} + \frac{4}{10^4} + \dots \right)$
 - $\pi = 3,14159\dots = +10^1 \left(\frac{3}{10} + \frac{1}{10^2} + \frac{4}{10^3} + \frac{1}{10^4} + \dots \right)$
- Historiquement, $\beta = 10$ car nous avons 10 doigts !
- **Ordinateurs** : $\beta = 2$ (numération binaire), $\beta = 8$ (num. octale),
ou encore $\beta = 16$ (num. hexadécimale)
- Unicité basée sur $d_1 \neq 0$:

$$\begin{aligned} 0,0038 &= 0,38 \cdot 10^{-2} = +10^{-2} \left(\frac{3}{10} + \frac{8}{10^2} \right) \\ &= 0,038 \cdot 10^{-3} = +10^{-1} \left(\frac{0}{10} + \frac{3}{10^2} + \frac{8}{10^3} \right) \end{aligned}$$

Le système F (1)

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

$$F = \left\{ y \in \mathbb{R} \mid y = \pm \beta^e \left(\frac{d_1}{\beta} + \frac{d_2}{\beta^2} + \cdots + \frac{d_t}{\beta^t} \right), e_{\min} \leq e \leq e_{\max} \right\}$$

ou encore

$$F = \{ y \in \mathbb{R} \mid y = \pm m \beta^{e-t}, e_{\min} \leq e \leq e_{\max} \}$$

Ceci correspond aux deux écritures :

- $0,0038 = +10^{-2} \left(\frac{3}{10} + \frac{8}{10^2} \right)$
- $0,0038 = +38 \cdot 10^{-4}$

avec $\beta = 10$, $e = -2$, $t = 2$, $e - t = -4$

- m s'appelle **la mantisse**. Notation : $m = \overline{d_1 d_2 \dots d_t}^\beta$
- Notons que $0 \notin F$.

Le système F (2)

Pour $y \neq 0$, on a

$$m \beta^{e-t} = \beta^e \left(\frac{d_1}{\beta} + \frac{d_2}{\beta^2} + \dots + \frac{d_t}{\beta^t} \right) \geq \beta^e \frac{1}{\beta} \implies m \geq \beta^{t-1}$$

$$m = \overline{d_1 d_2 \dots d_t} \beta = d_1 \beta^{t-1} + \dots + d_{t-k} \beta^k + \dots + d_{t-1} \beta + d_t < \beta^t$$

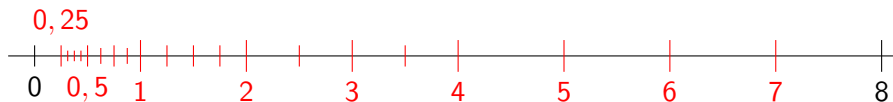
On a donc montré que $\beta^{t-1} \leq m < \beta^t$.

- F est un **système de nombres à virgule flottante** (floating point number system). Notation : $F(\beta, t, e_{\min}, e_{\max})$.
- Il dépend de quatre paramètres :
 - ① la base β (chiffres utilisés $0, 1, \dots, \beta - 1$),
 - ② la précision t (# chiffres utilisés pour représenter la mantisse),
 - ③ e_{\min} et e_{\max} qui définissent le domaine des exposants.

Exemple : $F(2, 3, -1, 3)$

- Un réel $y \in F(2, 3, -1, 3)$ s'écrit :

$$y = 2^e \left(\frac{1}{2} + \frac{d_2}{4} + \frac{d_3}{8} \right), \quad -1 \leq e \leq 3, \quad d_2, d_3 \in \{0, 1\}$$



- Écart entre deux nombres consécutifs $\times 2$ à chaque puissance de 2

Standard IEEE 754 et epsilon machine

- Dans le standard IEEE 754 utilisé par MATLAB, on a $\beta = 2$ et :
 - en simple précision : $t = 24$, $e_{\min} = -125$, $e_{\max} = 128$,
 - en double précision : $t = 53$, $e_{\min} = -1021$, $e_{\max} = 1024$.

Définition

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

- Par exemple, pour $F(2, 3, -1, 3)$, on a $\epsilon_M = 0,25$
- Dans MATLAB, c'est la variable eps.

Écart entre deux nombres consécutifs

Proposition

Pour $F(\beta, t, e_{\min}, e_{\max})$, on a $\epsilon_M = \beta^{1-t}$.

Proof.

On a $1 = \frac{1}{\beta} \beta = \overline{10 \dots 0}^\beta \beta$.

Nombre suivant : $\overline{10 \dots 1}^\beta \beta = \left(\frac{1}{\beta} + \frac{1}{\beta^t}\right) \beta = 1 + \beta^{1-t}$. □

Lemme

Dans le système de nombres à virgule flottante $F(\beta, t, e_{\min}, e_{\max})$, l'écart $|y - x|$ entre un nombre flottant x (non nul) et un nombre flottant y (non nul) adjacent vérifie $\beta^{-1} \epsilon_M |x| \leq |y - x| \leq \epsilon_M |x|$.

- **Représentation physique :**
 - simple précision 32 bits (bit = binary digit), 8 bits sont réservés à l'exposant et 24 bits (dont 1 pour le signe) à la mantisse.
 - double précision 64 bits, 11 bits sont réservés à l'exposant et 53 bits (dont 1 pour le signe) à la mantisse.
- **Arrondi :**
 - 1 par troncature : par exemple avec 3 chiffres, $0,8573\dots$ devient $0,857$.
 - 2 au plus près : $0,8573\dots$ devient $0,857$.
 - 3 au représentant le plus proche dont la dernière décimale est paire (rounding to even) : $0,8573\dots$ devient $0,858$.

Définition

Soit $G = G(\beta, t) = \{y \in \mathbb{R} \mid y = \pm m \beta^{e-t}\}$ sans conditions sur l'exposant e . L'application $\text{fl} : \mathbb{R} \rightarrow G, x \mapsto \text{fl}(x)$ est appelée opération d'arrondi.

• Étant donné un domaine $F(\beta, t, e_{\min}, e_{\max})$, il y a alors **dépassement de capacité** si :

- 1 $|\text{fl}(x)| > \max\{|y| \mid y \in F\}$. On parle d'**overflow**
- 2 $|\text{fl}(x)| < \min\{|y| \mid y \in F\}$. On parle d'**underflow**

Sinon, x est dans le domaine de F .

Définition

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

L'**erreur absolue** e est défini par $e = |x - \bar{x}|$.

L'**erreur relative** est $|\frac{e}{x}|$.

Le **pourcentage d'erreur** est l'erreur relative multipliée par 100.

- **En pratique**, on ne connaît en général pas la valeur exacte x mais on peut souvent avoir une idée de l'erreur maximale e que l'on a pu commettre : dans ce cas, **on majore la quantité** $|\frac{e}{x}|$

Théorème

Soit x un réel. Si x est dans le domaine $F(\beta, t, e_{\min}, e_{\max})$, alors il existe $\delta \in \mathbb{R}$ avec $|\delta| < u = \frac{1}{2} \beta^{1-t} = \frac{1}{2} \epsilon_M$ tel que $\text{fl}(x) = x(1 + \delta)$.

- L'erreur relative sur l'arrondi est égale à $|\delta| < u$: le nombre u s'appelle **unité d'erreur d'arrondi**
- Exemple : standard IEEE 754 utilisé par MATLAB, on a
 $u = 2^{-24} \approx 5,96 \cdot 10^{-8}$ en simple précision
 $u = 2^{-53} \approx 1,11 \cdot 10^{-16}$ en double précision.

Modèle de l'arithmétique flottante

- **Modèle Standard** (utilisé par le standard IEEE) :

Soit $x, y \in F(\beta, t, e_{\min}, e_{\max})$. Pour $op \in \{+, -, \times, \div, \sqrt{\cdot}\}$, on définit

$$x \boxed{op} y = fl(x op y) = (x op y) (1 + \delta), \quad |\delta| < u = \frac{1}{2} \beta^{1-t} = \frac{1}{2} \epsilon_M$$

- Nous allons maintenant nous intéresser aux **erreurs faites par** \boxed{op}

Analyse d'erreurs : non-associativité

- Contrairement à op , l'opération $\boxed{\text{op}}$ n'est pas associative:

$$(x \boxed{\text{op}} y) \boxed{\text{op}} z \neq x \boxed{\text{op}} (y \boxed{\text{op}} z)$$

- Ceci est dû aux erreurs d'arrondi !

Par exemple, supposons que les réels soient calculés avec 3 chiffres significatifs et arrondis à la décimale la plus proche et cherchons à calculer la somme $x \boxed{+} y \boxed{+} z$ avec $x = 8,22$, $y = 0,00317$ et $z = 0,00432$.

- $x \boxed{+} y = 8,22$ donc $(x \boxed{+} y) \boxed{+} z = 8,22$
- $y \boxed{+} z = 0,01$ donc $x \boxed{+} (y \boxed{+} z) = 8,23$

Analyse d'erreurs : erreurs d'arrondi sur une somme

- Calculer $S = u_1 + u_2 + \dots + u_n$ dans $F(\beta, t, e_{\min}, e_{\max})$
- On calcule alors les **sommes partielles** S_j par la récurrence $S_0 = 0$, $S_j = S_{j-1} + u_j$
- Si u_j connus exactement, alors les **erreurs d'arrondi** ΔS_j commises sur le calcul des sommes partielles S_j vérifient

$$\Delta S_j \leq \Delta S_{j-1} + \delta(S_{j-1} + u_j) = \Delta S_{j-1} + \delta S_j, \quad |\delta| < u$$

- L'**erreur globale sur** $S = S_n$ vérifie donc $\Delta S \leq \delta(S_2 + \dots + S_n)$,

$$\Delta S \leq \delta(u_n + 2u_{n-1} + 3u_{n-2} + \dots + (n-1)u_2 + (n-1)u_1).$$

↪ Erreur minimale en sommant d'abord les termes les plus petits

Analyse d'erreurs : erreurs d'arrondi sur un produit

- Calculer $P = u_1 u_2 \dots u_n$ dans $F(\beta, t, e_{\min}, e_{\max})$
- On calcule alors les produits P_i par la récurrence $P_0 = 1$,
 $P_i = P_{i-1} u_i$
- Si u_i connus exactement, alors les erreurs d'arrondi ΔP_i commises sur le calcul des P_i vérifient

$$\Delta P_i \leq (\Delta P_{i-1}) u_i + \delta (P_{i-1} u_i) = \Delta P_{i-1} u_i + \delta P_i, \quad |\delta| < u$$

- L'erreur globale sur $P = P_n$ vérifie donc

$$\Delta P \leq (k - 1) \delta P_n.$$

↪ Contrairement au cas de l'addition, la majoration de l'erreur ne dépend pas de l'ordre des facteurs.

Phénomènes de compensation (1)

- Phénomènes qui se produisent lorsque l'on tente de soustraire des nombres très proches

- Exemple 1 : $E = \sqrt{x+1} - \sqrt{x}$ avec $x > 0$

Sous MATLAB, on obtient :

- pour $x = 10^9$, $E = 1,5811.10^{-5}$
- pour $x = 10^{16}$, $E = 0 !$

Si l'on remarque que $E = \frac{1}{\sqrt{x+1} + \sqrt{x}}$, alors, en utilisant cette nouvelle formule, on trouvera :

- pour $x = 10^9$, $E = 1,5811.10^{-5}$
- pour $x = 10^{16}$, $E = 5,000.10^{-9} !$

Phénomènes de compensation (2)

- Phénomènes qui se produisent lorsque l'on tente de soustraire des nombres très proches

- Exemple 2 : équation du second degré $x^2 - 1634x + 2 = 0$.

Supposons que les calculs soient effectués avec 10 chiffres significatifs. Les formules habituelles donnent

$$\Delta' = \left(\frac{1634}{2}\right)^2 - 2 = 667487, \quad \sqrt{\Delta'} = 816,9987760$$

$$x_1 = \frac{1634}{2} + \sqrt{\Delta'} = 817 + 816,9987760 = 1633,998776,$$

$$x_2 = \frac{1634}{2} - \sqrt{\Delta'} = 817 - 816,9987760 = 0,0012240.$$

↪ perte de 5 chiffres significatifs sur x_2 !

Pour y remédier, on peut utiliser la relation $x_1 x_2 = 2$ et calculer

$$x_2 = \frac{2}{x_1} = \frac{2}{1633,998776} = 0,001223991125.$$

Phénomènes d'instabilité numérique (1)

- Phénomènes d'amplification d'erreur d'arrondi : se produisent pour des calculs récurrents ou itératifs

Exemple 1 : calcul de $I_n = \int_0^1 \frac{x^n}{10+x} dx$, $n \in \mathbb{N}$

- Calcul direct :

$$I_0 = \ln\left(\frac{11}{10}\right), \quad I_n = \frac{1}{n} - 10 I_{n-1}$$

↪ calcul de I_n par récurrence

- Numériquement, résultats très mauvais !
- Explication : erreur d'arrondi ΔI_n vérifie $\Delta I_n \approx 10 \Delta I_{n-1}$ et croît exponentiellement: l'erreur sur I_0 est multipliée par 10^n sur I_n .

↪ Cette formule de récurrence ne peut pas nous permettre de calculer la valeur de I_{36} par exemple

Phénomènes d'instabilité numérique (2)

- Phénomènes d'amplification d'erreur d'arrondi : se produisent pour des calculs récurrents ou itératifs

Exemple 1 : calcul de $I_n = \int_0^1 \frac{x^n}{10+x} dx$, $n \in \mathbb{N}$

- Pour remédier à ce problème, on peut renverser la récurrence :
 $I_{n-1} = \frac{1}{10} \left(\frac{1}{n} - I_n \right)$.
- on obtient alors $\Delta I_{n-1} \approx \frac{1}{10} \Delta I_n$.

$$10 \leq 10 + x \leq 11 \implies \frac{1}{11(n+1)} \leq I_n \leq \frac{1}{10(n+1)}$$

- Approximation $I_n \approx \frac{1}{11(n+1)} \rightsquigarrow$ valeur de départ pour notre récurrence renversée. Exemple, $I_{46} \approx \frac{1}{11(46+1)}$, on obtient pour I_{36} une erreur relative meilleure que 10^{-10} .

- Importance du coefficient d'amplification d'erreur

Phénomènes d'instabilité numérique (3)

- Phénomènes d'amplification d'erreur d'arrondi : se produisent pour des calculs récurrents ou itératifs

Exemple 2 : On considère la suite définie par (J.-M. Muller) :

$$\begin{cases} u_0 = 2, \\ u_1 = -4, \\ u_n = 111 - \frac{1130}{u_{n-1}} + \frac{3000}{u_{n-1} u_{n-2}}, \end{cases}$$

- Limite théorique 6 mais en pratique 100 !

Phénomènes d'instabilité numérique (4)

- **Explication** : solution générale de $u_n = 111 - \frac{1130}{u_{n-1}} + \frac{3000}{u_{n-1} u_{n-2}}$:

$$u_n = \frac{\alpha 100^{n+1} + \beta 6^{n+1} + \gamma 5^{n+1}}{\alpha 100^n + \beta 6^n + \gamma 5^n},$$

où α , β et γ dépendent des valeurs initiales u_0 et u_1

- $\alpha \neq 0 \rightsquigarrow$ convergence vers 100, sinon convergence vers 6 ($\beta \neq 0$)
- Dans notre exemple ($u_0 = 2$, $u_1 = -4$) : $\alpha = 0$, $\beta = -3$ et $\gamma = 4$
 \rightsquigarrow **À cause des erreurs d'arrondi**, même les premiers termes calculés seront différents des termes exacts et donc la **valeur de α correspondant à ces termes calculés sera très petite mais non-nulle** ce qui suffira à faire en sorte que la suite converge vers 100 au lieu de 6.

Erreur amont et erreur aval

- Considérons un problème que l'on résout à l'aide d'un algorithme numérique : entrée $x \rightsquigarrow y = f(x)$
- En pratique, compte tenu des erreurs d'arrondis, étant donnée une entrée x , nous allons obtenir une sortie $\bar{y} \neq y = f(x)$
- **Erreur aval** : $|\bar{y} - y|$
- **Erreur amont** (ou erreur inverse) : plus petit δx tel que la solution algébrique $f(x + \delta x)$ correspondant à l'entrée $x + \delta x$ soit égale à \bar{y} .
- Erreur aval \approx erreur amont \times Conditionnement.
- **Erreur amont plus intéressante** :
 - nous renseigne sur le problème qui est réellement résolu par l'algorithme numérique
 - en pratique, nous ne connaissons en général qu'une valeur approchée de l'entrée

Outils théoriques de l'analyse d'erreurs

- Formule $(x \times y) + z$ avec x, y et z dans $F(\beta, t, e_{\min}, e_{\max})$.
- On a alors :

$$\begin{aligned}\text{fl}((x \times y) + z) &= [\text{fl}(x \times y) + z] (1 + \delta_1) \\ &= [(x \times y)(1 + \delta_2) + z] (1 + \delta_1) \\ &= (x \times y)(1 + \delta_2)(1 + \delta_1) + z(1 + \delta_1),\end{aligned}$$

Lemme

Si pour tout $i = 1, \dots, k$, on a $|\delta_i| < u$ et si $ku < 1$, alors il existe θ_k tel que $|\theta_k| \leq \frac{ku}{1-ku}$ et $\prod_{i=1}^k (1 + \delta_i) \leq 1 + \theta_k$.

- Notation $\langle k \rangle = \prod_{i=1}^k (1 + \delta_i)$ avec $\langle j \rangle \cdot \langle k \rangle = \langle j+k \rangle$.

$$\begin{aligned}\text{fl}((x \times y) + z) &= (x \times y) \langle 2 \rangle + z \langle 1 \rangle \\ &\leq (x \times y) \left(1 + \frac{2u}{1-2u}\right) + z \left(1 + \frac{u}{1-u}\right).\end{aligned}$$

Chapitre 2

Résolution d'un système d'équations linéaires (Partie 1) : méthodes directes

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

$$(S) \begin{cases} a_{1,1} x_1 + a_{1,2} x_2 + \cdots + a_{1,n} x_n = b_1 \\ a_{2,1} x_1 + a_{2,2} x_2 + \cdots + a_{2,n} x_n = b_2 \\ \vdots \\ a_{n,1} x_1 + a_{n,2} x_2 + \cdots + a_{n,n} x_n = b_n \end{cases}$$

- **Données** : les a_{ij} et b_1, \dots, b_n dans \mathbb{K} avec $\mathbb{K} = \mathbb{R}$ ou \mathbb{C}
- **Inconnues** : x_1, \dots, x_n dans \mathbb{K}

$$(S) \quad Ax = b,$$

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ a_{n,1} & \dots & \dots & a_{n,n} \end{pmatrix} \in \mathbb{M}_{n \times n}(\mathbb{K})$$

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

- Dans ce chapitre, **A est inversible !**

Motivation (1)

- Pourquoi ce problème se pose-t-il ?
- En effet, les formules de Cramer donnent la solution :

$$\forall i \in \{1, \dots, n\}, \quad x_i = \frac{\begin{vmatrix} a_{1,1} & \dots & a_{1,(i-1)} & b_1 & a_{1,(i+1)} & \dots & a_{1,n} \\ \vdots & & & \vdots & & & \vdots \\ a_{n,1} & \dots & a_{n,(i-1)} & b_n & a_{n,(i+1)} & \dots & a_{n,n} \end{vmatrix}}{\det(A)}.$$

- Regardons le nombre d'opérations nécessaires !

Motivation (2)

- Regardons le nombre d'opérations nécessaires !

Lemme

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

- Lorsque $n = 100$, nombre d'opérations de l'ordre de $9,4 \cdot 10^{161}$!
↪ Ordi. fonctionnant à 100 megaflops, environ $3 \cdot 10^{146}$ années !
↪ Impossible d'utiliser Cramer pour résoudre de grands systèmes !

Résolution d'un système triangulaire

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

$$(S) \begin{cases} a_{1,1} x_1 + a_{1,2} x_2 + \cdots + a_{1,n} x_n = b_1 \\ \phantom{a_{1,1} x_1} + a_{2,2} x_2 + \cdots + a_{2,n} x_n = b_2 \\ \phantom{a_{1,1} x_1} \phantom{+ a_{2,2} x_2} + \cdots \phantom{+ a_{2,n} x_n} = \vdots \\ \phantom{a_{1,1} x_1} \phantom{+ a_{2,2} x_2} + a_{n,n} x_n = b_n. \end{cases}$$

- A inversible \Rightarrow les $a_{i,i}$ sont non nuls

\rightsquigarrow Système facile à résoudre : algorithme de substitution rétrograde

Résolution d'un système triangulaire : exemple

- On considère le système triangulaire supérieur :

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

- 3ième équation : $x_3 = \frac{1}{8}$
 - 2ième équation : $x_2 = \frac{-5/2 + 16x_3}{-4} = \frac{1}{8}$
 - 1ière équation : $x_1 = \frac{1 - 2x_2 - 5x_3}{1} = \frac{1}{8}$
- Idem si A triang. inf. : algorithme de substitution progressive

Systeme triangulaire : # opérations et propriétés

Lemme

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

Lemme (Propriétés)

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

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

3 méthodes directes étudiées dans la suite

- 1 **Méthode de Gauss** : système $\rightsquigarrow (MA)x = Mb$ avec MA triang. sup. (sans calculer explicitement M).
 - Associée à la factorisation $A = LU$ de la matrice A avec L triang. inf. et U triang. sup., $Ax = b \Leftrightarrow Ly = b, Ux = y$
- 2 **Méthode de Cholesky**
 - Associée à la factorisation de Cholesky $A = R^T R$ avec R triang. sup., $Ax = b \Leftrightarrow R^T y = b, Rx = y$
 - Méthode valable pour A symétrique et définie positive
- 3 **Méthode de Householder**
 - Associée à la factorisation $A = QR$ avec R triang. sup. et Q ortho., Q produit de $n - 1$ matrices de Householder H_i .
 - $Ax = b$ s'écrit alors $H_{n-1} \cdots H_2 H_1 Ax = H_{n-1} \cdots H_2 H_1 b$ facile à résoudre car $H_{n-1} \cdots H_2 H_1 A$ triang. sup.

Méthode de Gauss : description (1)

- $(S) : Ax = b$ avec A inversible
- On pose $b^{(1)} = b$ et $A^{(1)} = A = (a_{i,j}^{(1)}) \rightsquigarrow (S^{(1)}) : A^{(1)}x = b^{(1)}$

Étape 1

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

Méthode de Gauss : description (2)

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

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

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

$$A^{(2)} = \begin{pmatrix} a_{1,1}^{(1)} & a_{1,2}^{(1)} & \dots & a_{1,n}^{(1)} \\ 0 & a_{2,2}^{(2)} & \dots & a_{2,n}^{(2)} \\ 0 & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ 0 & a_{n,2}^{(2)} & \dots & a_{n,n}^{(2)} \end{pmatrix}, \quad b^{(2)} = \begin{pmatrix} b_1^{(1)} \\ b_2^{(2)} \\ \vdots \\ b_n^{(2)} \end{pmatrix}$$

Étape k

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

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

Méthode de Gauss : description (4)

• A inversible \Rightarrow on suppose (quitte à permuter lignes) $a_{k,k}^{(k)} \neq 0$.
C'est le **kième pivot** de l'élimination de Gauss

• Par le même principe qu'à l'étape 1 et en utilisant $g_{i,k} = \frac{a_{i,k}^{(k)}}{a_{k,k}^{(k)}}$ pour $i > k$, on obtient alors $(S^{(k+1)}) : A^{(k+1)} x = b^{(k+1)}$ avec

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

Étape $n - 1$

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

$$A^{(n)} = \begin{pmatrix} a_{1,1}^{(1)} & & \dots & \dots & a_{1,n}^{(1)} \\ 0 & a_{2,2}^{(2)} & & & a_{2,n}^{(2)} \\ 0 & 0 & a_{3,3}^{(3)} & & a_{3,n}^{(3)} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & a_{n,n}^{(n)} \end{pmatrix}$$

- On peut le résoudre par l'algorithme de substitution rétrograde

Méthode de Gauss : exemple (1)

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

- Le premier pivot de l'élimination de Gauss est donc $a_{1,1}^{(1)} = 1$ et on a $g_{2,1}^{(1)} = 3$, $g_{3,1}^{(1)} = 0$. La première étape fournit donc

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

Méthode de Gauss : exemple (2)

- Le second pivot de l'élimination de Gauss est donc $a_{2,2}^{(2)} = -4$ et on a $g_{3,2}^{(2)} = -\frac{5}{4}$. On obtient donc le système

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

- Algorithme de substitution rétrograde** $\rightsquigarrow x_1 = x_2 = x_3 = \frac{1}{8}$

Point de vue numérique : stratégies de choix du pivot (1)

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

Certains choix de pivots peuvent s'avérer plus judicieux que d'autres.

Exemple : considérons le système (S) : $Ax = b$ où

$$A = \begin{pmatrix} \alpha & 1 \\ 1 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \quad \alpha \in \mathbb{R}^*$$

- On suppose de plus $\alpha \neq 1$ de sorte que A est inversible
- Solution $x_1^* = \frac{1}{1-\alpha}$, $x_2^* = \frac{1-2\alpha}{1-\alpha}$
- Supposons maintenant que α est *très petit* ($0 \ll \alpha < 1$) et appliquons l'élimination de Gauss

Point de vue numérique : stratégies de choix du pivot (3)

- Premier pivot α , $g_{2,1} = \frac{1}{\alpha} \rightsquigarrow (S^{(2)}) : A^{(2)} x = b^{(2)}$ avec

$$A^{(2)} = \begin{pmatrix} \alpha & 1 \\ 0 & 1 - \frac{1}{\alpha} \end{pmatrix}, \quad b^{(2)} = \begin{pmatrix} 1 \\ 2 - \frac{1}{\alpha} \end{pmatrix}.$$

$\rightsquigarrow -\frac{1}{\alpha} x_2 \approx -\frac{1}{\alpha}$ d'où $x_2 \approx 1$ et $x_1 \approx 0$ ce qui est faux !

- L'erreur ne provient pas seulement du fait que α est très petit car si on multiplie la première ligne par une puissance de 10 quelconque, on va trouver la même erreur ...

Point de vue numérique : stratégies de choix du pivot (4)

- Notons $x_2 = x_2^* + \delta x_2$ où $|\delta x_2|$ est l'erreur absolue sur x_2

- On a alors

$$x_1 = \frac{1 - x_2}{\alpha} = \frac{1 - x_2^*}{\alpha} - \frac{\delta x_2}{\alpha},$$

↪ Erreur $\delta x_1 = \frac{1}{\alpha} \delta x_2$ sur x_1 très amplifiée par rapport à δx_2 .

- Cause = déséquilibre entre coeffs de x_1 et x_2 sur la ligne du pivot
- Pour y remédier, échanger les lignes et appliquer Gauss avec 1 comme pivot. On obtient alors

$$A^{(2)} = \begin{pmatrix} 1 & 1 \\ 0 & 1 - \alpha \end{pmatrix}, \quad b = \begin{pmatrix} 2 \\ 1 - 2\alpha \end{pmatrix},$$

d'où $x_2 \approx 1$ et $x_1 \approx 1$ ce qui est correct.

Élimination de Gauss à pivot partiel

- À l'étape k , on échange les lignes k et k' ($k' \geq k$) de $A^{(k)}$ de telle sorte que : $|a_{k,k}^{(k)}| = \max\{|a_{i,k}^{(k)}|, i \geq k\}$.

Exemple : pour

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

à la première étape, on permute les lignes 1 et 2 :

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

Élimination de Gauss à pivot total

- À l'étape k , on échange à la fois les lignes k et k' ($k' \geq k$) et les colonnes k et k'' ($k'' \geq k$) de telle sorte que :

$$|a_{k,k}^{(k)}| = \max\{|a_{i,j}^{(k)}|, i \geq k, j \geq k\}.$$

Attention : Si on échange des colonnes, cela modifie l'ordre des composantes de x donc il faut penser à rétablir le bon ordre à la fin.

Exemple : pour

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

à la première étape, on permute les colonnes 1 et 3 :

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

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

Définition

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

Lemme

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

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

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

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

Définition

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

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

Théorème

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

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

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

Lemme

Avec les notations précédentes, on a

$$(G_{n-1} G_{n-2} \cdots G_1)^{-1} = \begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ g_{2,1} & 1 & \ddots & & \vdots \\ g_{3,1} & g_{3,2} & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ g_{n,1} & g_{n,2} & \cdots & g_{n,n-1} & 1 \end{pmatrix}.$$

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

Corollaire

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

$$A = \begin{pmatrix} 1 & 0 & \dots & \dots & 0 \\ g_{2,1} & 1 & \ddots & & \vdots \\ g_{3,1} & g_{3,2} & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ g_{n,1} & g_{n,2} & \dots & g_{n,n-1} & 1 \end{pmatrix} \begin{pmatrix} a_{1,1}^{(1)} & \dots & \dots & a_{1,n}^{(1)} \\ 0 & a_{2,2}^{(2)} & & a_{2,n}^{(2)} \\ 0 & 0 & a_{3,3}^{(3)} & a_{3,n}^{(3)} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & a_{n,n}^{(n)} \end{pmatrix}.$$

- Remarque : la matrice L obtenue est à diagonale unité.

Factorisation LU : exemple

Pour la matrice du système

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

on a :

$$\underbrace{\begin{pmatrix} 1 & 2 & 5 \\ 3 & 2 & -1 \\ 0 & 5 & 3 \end{pmatrix}}_A = \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 0 & -\frac{5}{4} & 1 \end{pmatrix}}_L \underbrace{\begin{pmatrix} 1 & 2 & 5 \\ 0 & -4 & -16 \\ 0 & 0 & -17 \end{pmatrix}}_U$$

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

Proposition

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

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

$$Ax = b \iff L U x = b \iff \begin{cases} Ly = b, \\ Ux = y. \end{cases}$$

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

Définition

On appelle *matrice de permutation associée à une permutation* $\sigma \in \mathcal{S}_n$, la matrice $\mathcal{P}_\sigma = (\delta_{i\sigma(j)})$ où $\delta_{ij} = 1$ si $i = j$, $\delta_{ij} = 0$ sinon.

- Exemple :

$$\sigma : (1, 2, 3, 4, 5) \mapsto (3, 2, 5, 1, 4) \rightsquigarrow \mathcal{P}_\sigma = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

- Multiplier A à gauche (resp. à droite) par une matrice de permutation revient alors à permuter des lignes (resp. les colonnes)
- Les matrices de permutation sont orthogonales : $\mathcal{P}_\sigma^{-1} = \mathcal{P}_\sigma^T$.

Lien avec la factorisation LU d'une matrice (6)

- Nous avons vu une CNS pour qu'une matrice inversible admette une factorisation LU. Lorsque cette factorisation LU n'existe pas, on peut tout de même utiliser le théorème suivant :

Théorème

Soit $A \in \mathbb{M}_{n \times n}(\mathbb{K})$ une matrice carrée inversible. Il existe une matrice de permutation \mathcal{P} telle que $\mathcal{P}A$ admette une factorisation LU.

- Notons que dans ce cas, on a :

$$Ax = b \iff \mathcal{P}Ax = \mathcal{P}b \iff L U x = \mathcal{P}b \iff \begin{cases} Ly = \mathcal{P}b, \\ Ux = y. \end{cases}$$

- En pratique, on résout donc d'abord $Ly = \mathcal{P}b$ puis connaissant y on résout $Ux = y$.

Lemme

Soit $A \in \mathbb{M}_{n \times n}(\mathbb{K})$ une matrice carrée inversible. Résoudre un système linéaire $(S) : Ax = b$ via l'élimination de Gauss nécessite un nombre d'opérations à virgule flottante équivalent à $\frac{2n^3}{3}$ lorsque n tend vers l'infini. Ce coût asymptotique est aussi celui du calcul de la factorisation LU de A .

- Pour $n = 100$, cela donne $6,6 \cdot 10^5$ opérations à virgule flottante à comparer à $9,4 \cdot 10^{161}$ avec Cramer
- Avec un ordinateur fonctionnant à 100 megaflops, cela prendra moins de 7 millièmes de secondes. À comparer avec $3 \cdot 10^{146}$ années pour Cramer

Faut-il inverser une matrice ?

- Étant donnée la factorisation LU de A , le coût du calcul de l'inverse A^{-1} de A lorsque n tend vers l'infini est de $\frac{4n^3}{3}$ opérations à virgule flottante
 - Au total, lorsque n tend vers l'infini, il faut donc $2n^3$ opérations à virgule flottante pour calculer l'inverse de A
- ↪ Asymptotiquement (*i.e.*, lorsque n tend vers l'infini), il faut 3 fois plus d'opérations à virgule flottante pour calculer l'inverse de A que pour résoudre le système linéaire $Ax = b$ en utilisant l'élimination de Gauss
- ⇒ Il ne faut pas calculer l'inverse d'une matrice pour résoudre un système linéaire

Résolution de plusieurs systèmes de même matrice A

- Soit $A \in \mathbb{M}_{n \times n}(\mathbb{K})$ une matrice carrée inversible et supposons que l'on ait à résoudre K systèmes linéaires avec la même matrice A et N seconds membres $b^{[1]}, \dots, b^{[K]}$
- Gauss à chacun de ces systèmes $\rightsquigarrow K \frac{4n^3 + 9n^2 - 7n}{6}$ flops
- Facto. LU de A puis résolution successive des $2K$ systèmes triangulaires $\rightsquigarrow \left(\frac{4n^3 + 3n^2 - 7n}{6} \right) + 2Kn^2$ flops
- Calcul de l'inverse A^{-1} de A puis résolution successive des systèmes en posant $x^{[i]} = A^{-1} b^{[i]}$ $\rightsquigarrow 2n^3 + 2Kn^2$ flops

Méthode de Cholesky (1)

- Alternative à Gauss pour matrices symétriques et définies positives

Définition

Une matrice $A \in \mathbb{M}_{n \times n}(\mathbb{K})$ est dite **symétrique** si elle est égale à sa transposée, i.e., $A^T = A$.

Définition

Soit $\mathbb{K} = \mathbb{R}$ ou \mathbb{C} . **Le produit scalaire canonique sur \mathbb{K}^n** est défini comme l'application $\langle \cdot, \cdot \rangle : \mathbb{K}^n \times \mathbb{K}^n \rightarrow \mathbb{K}$, $(u, v) \mapsto \langle u, v \rangle$ qui vérifie :

- Si $\mathbb{K} = \mathbb{R}$, $\langle u, v \rangle = v^T u = \sum_{i=1}^n u_i v_i$ (produit scalaire euclidien),
- Si $\mathbb{K} = \mathbb{C}$, $\langle u, v \rangle = \bar{v}^T u = \sum_{i=1}^n u_i \bar{v}_i$ (produit scalaire hermitien).

Méthode de Cholesky (2)

Définition

Une matrice $A \in \mathbb{M}_{n \times n}(\mathbb{K})$ est dite **définie positive**, resp. **semi définie positive** si pour tout $x \in \mathbb{R}^n$ non nul, on a $\langle Ax, x \rangle > 0$, resp. $\langle Ax, x \rangle \geq 0$.

- 1 Une matrice définie positive est inversible ;
- 2 Si $A \in \mathbb{M}_{n \times n}(\mathbb{K})$ est inversible, alors $A^T A$ est symétrique et définie positive ;
- 3 Si $A = (a_{i,j}) \in \mathbb{M}_{n \times n}(\mathbb{K})$ est définie positive, alors $a_{i,i} > 0$ pour tout $i = 1, \dots, n$.

Théorème

Une matrice réelle $A \in \mathbb{M}_{n \times n}(\mathbb{R})$ est symétrique définie positive ssi il existe une matrice $L = (l_{i,j})_{1 \leq i,j \leq n} \in \mathbb{M}_{n \times n}(\mathbb{R})$ triangulaire inférieure inversible telle que $A = LL^T$. De plus, si pour tout $i = 1, \dots, n$, $l_{i,i} \geq 0$, alors L est unique.

Algorithme de Cholesky

Entrée : $A = (a_{i,j})_{1 \leq i,j \leq n} \in \mathbb{M}_{n \times n}(\mathbb{R})$ symétrique et définie positive.

Sortie : $L = (l_{i,j})_{1 \leq i,j \leq n} \in \mathbb{M}_{n \times n}(\mathbb{R})$ tel que $A = L L^T$.

- 1 $l_{1,1} = \sqrt{a_{1,1}}$;
- 2 Pour i de 2 à n par pas de 1, faire :
 - $l_{i,1} = \frac{a_{i,1}}{l_{1,1}}$;
- 3 Pour j de 2 à n par pas de 1, faire :
 - Pour i de 1 à $j-1$ par pas de 1, faire :
 $l_{i,j} = 0$;
 - $l_{j,j} = \sqrt{a_{j,j} - \sum_{k=1}^{j-1} l_{j,k}^2}$;
 - Pour i de $j+1$ à n par pas de 1, faire :
 $l_{i,j} = \frac{a_{i,j} - \sum_{k=1}^{j-1} l_{i,k} l_{j,k}}{l_{j,j}}$;
- 4 Retourner $L = (l_{i,j})_{1 \leq i,j \leq n} \in \mathbb{M}_{n \times n}(\mathbb{R})$.

Proposition

L'algorithme de Cholesky décrit ci-dessus nécessite n extractions de racines carrées et un nombre d'opérations à virgule flottante équivalent à $\frac{n^3}{3}$ lorsque n tend vers l'infini.

- Asymptotiquement, presque deux fois moins d'opérations à virgule flottante que pour LU

↪ Il est conseillé de l'utiliser lorsque A est réelle symétrique et définie positive

Matrices de Householder

- Ici $A \in \mathbb{M}_{n \times n}(\mathbb{R})$ est une matrice **réelle** inversible

Définition

On appelle **matrice (élémentaire) de Householder** une matrice H de la forme $H_u = \mathbb{I}_n - 2 u u^T$, où $u \in \mathbb{R}^n$ est un vecteur unitaire c'est-à-dire de norme 1 pour la norme associée au produit scalaire canonique sur \mathbb{R}^n définie par $\|u\| = \sqrt{\langle u, u \rangle}$.

- Exemple : pour $n = 3$, on peut considérer le vecteur $u = \frac{1}{\sqrt{6}} \begin{pmatrix} -1 & 1 & 2 \end{pmatrix}^T$ qui vérifie bien $\|u\| = 1$. On obtient alors

la matrice de Householder $H_u = \frac{1}{3} \begin{pmatrix} 2 & 1 & 2 \\ 1 & 2 & -2 \\ 2 & -2 & -1 \end{pmatrix}$.

Matrices Orthogonales

Définition

Une matrice $A \in \mathbb{M}_{n \times n}(\mathbb{K})$ est dite **orthogonale** si elle est réelle, i.e., $A \in \mathbb{M}_{n \times n}(\mathbb{R})$ et si $AA^T = A^T A = \mathbb{I}_n$.

Proposition

Toute matrice de Householder H est symétrique et orthogonale.

Proposition

Pour tout vecteur $u \in \mathbb{R}^n$ tel que $\|u\| = 1$, on a $H_u u = -u$. De plus, si $v \in \mathbb{R}^n$ est orthogonal à u , i.e., $\langle u, v \rangle = 0$, alors $H_u v = v$.

- H_u représente la symétrie orthogonale par rapport à u^\perp

Lemme

Soit x et y deux vecteurs de \mathbb{R}^n tels que $x \neq y$ et $\|x\| = \|y\|$. Alors il existe un vecteur unitaire $u \in \mathbb{R}^n$ tel que $H_u x = y$.

Principe de la méthode de Householder

- Méthode basée sur les deux propositions suivantes :

Proposition

Soit v un vecteur non nul de \mathbb{R}^n . Alors il existe une matrice de Householder H et un réel α tels que $Hv = \alpha e_1$, où $e_1 = (1, 0, \dots, 0)^T$ est le premier vecteur de la base canonique de \mathbb{R}^n .

Proposition

Soit $u = (u_i)$ un vecteur unitaire de \mathbb{R}^n tel que $u_1 = \dots = u_p = 0$ pour $p < n$. On décompose alors u en deux blocs : $u = (0 \ z)^T$ avec $z \in \mathbb{R}^{n-p}$. La matrice de Householder H_u se décompose alors par blocs de la manière suivante : $H_u = \begin{pmatrix} \mathbb{I}_p & 0 \\ 0 & H_z \end{pmatrix}$.

Principe de la méthode de Householder

$$A = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \end{pmatrix}$$

Principe de la méthode de Householder

$$A = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \end{pmatrix}$$

Principe de la méthode de Householder

$$H_1 A = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \end{pmatrix}$$

Principe de la méthode de Householder

$$H_1 A = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \end{pmatrix}$$

Principe de la méthode de Householder

$$H_2 H_1 A = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & \times & \times & \times \end{pmatrix}$$

Principe de la méthode de Householder

$$H_2 H_1 A = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & \times & \times & \times \end{pmatrix}$$

Principe de la méthode de Householder

$$H_3 H_2 H_1 A = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times & \times \end{pmatrix}$$

Principe de la méthode de Householder

$$H_3 H_2 H_1 A = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times & \times \end{pmatrix}$$

Principe de la méthode de Householder

$$H_4 H_3 H_2 H_1 A = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{pmatrix} = R$$

Donc $A = (H_4 H_3 H_2 H_1)^T R$.

Exemple (1)

$$(S) : \begin{cases} 2x_1 + x_2 + 2x_3 = 1, \\ x_1 + x_2 + 2x_3 = 1, \\ 2x_1 + x_2 + x_3 = 1. \end{cases}$$

Étape 1

• 1^{ère} colonne de A donnée par (S) : $a_1 = (2 \ 1 \ 2)^T$

• $v_1 = \frac{a_1}{\|a_1\|} - e_1 = \frac{1}{3}(-1 \ 1 \ 2)^T$

• $u_1 = \frac{v_1}{\|v_1\|} = \frac{1}{\sqrt{6}}(-1 \ 1 \ 2)^T$

• Matrice de Householder $H_{u_1} = \frac{1}{3} \begin{pmatrix} 2 & 1 & 2 \\ 1 & 2 & -2 \\ 2 & -2 & -1 \end{pmatrix}$

$$Ax = b \Leftrightarrow H_{u_1} Ax = H_{u_1} b \Leftrightarrow \begin{pmatrix} 9 & 5 & 8 \\ 0 & 1 & 4 \\ 0 & -1 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 5 \\ 1 \\ -1 \end{pmatrix}$$

Exemple (2)

Étape 2

- Dans \mathbb{R}^2 et on considère le vecteur $a_2 = (1 \quad -1)^T$
- $z'_2 = \frac{a_2}{\|a_2\|} - e'_1$ où $e'_1 = (1, 0)$
- $z_2 = \frac{z'_2}{\|z'_2\|}$ et $H_{z_2} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ -1 & -1 \end{pmatrix}$
- $u_2 = (0 \quad z_2)^T$ et $H_{u_2} = \begin{pmatrix} 1 & 0 \\ 0 & H_{z_2} \end{pmatrix}$

$$Ax = b \Leftrightarrow H_{u_2} H_{u_1} Ax = H_{u_2} H_{u_1} b \Leftrightarrow \begin{pmatrix} 9 & 5 & 8 \\ 0 & \sqrt{2} & \frac{5}{\sqrt{2}} \\ 0 & 0 & -\frac{3}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 5 \\ \sqrt{2} \\ 0 \end{pmatrix}$$

\rightsquigarrow Système triangulaire et $x = (0 \quad 1 \quad 0)^T$.

Définition

Soit $A \in \mathbb{M}_{n \times n}(\mathbb{R})$ une matrice carrée réelle inversible. On appelle **factorisation QR de A** une factorisation de la forme $A = QR$ avec $Q \in \mathbb{M}_{n \times n}(\mathbb{R})$ orthogonale et $R \in \mathbb{M}_{n \times n}(\mathbb{R})$ triangulaire supérieure.

- Généralisation de l'exemple précédent \rightsquigarrow pour A quelconque de taille n , on obtient $n - 1$ matrices de Householder H_1, \dots, H_{n-1} telles que $R = H_{n-1} H_{n-2} \cdots H_1 A$ triang. sup.
- $Q = (H_{n-1} H_{n-2} \cdots H_1)^{-1}$ (orthogonale) de sorte que $A = QR$

Théorème

Pour toute matrice réelle $A \in \mathbb{M}_{n \times n}(\mathbb{R})$, il existe une matrice orthogonale $Q \in \mathbb{M}_{n \times n}(\mathbb{R})$ produit d'au plus $(n - 1)$ matrices de Householder et une matrice triangulaire supérieure $R \in \mathbb{M}_{n \times n}(\mathbb{R})$ telles que $A = QR$.

Proposition

La méthode de Householder pour résoudre un système linéaire nécessite un nombre d'opérations à virgule flottante équivalent à $\frac{4n^3}{3}$ lorsque n tend vers l'infini.

- **Coût relativement élevé** comparé à Gauss ou Cholesky.
- **Plus stable numériquement** que Gauss ou Cholesky
- Factorisation QR existe pour des **matrices rectangulaires** : utilisé pour des problèmes de **moindres carrés** (voir TP).

Chapitre 3

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

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

Définition

On appelle **norme sur E** une application $\|\cdot\| : E \rightarrow \mathbb{R}_+$ telle que :

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

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

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

Définition

Une norme $\|\cdot\|$ sur $\mathbb{M}_{n \times n}(\mathbb{K})$ est une **norme matricielle** si elle vérifie : $\forall (A, B) \in \mathbb{M}_{n \times n}(\mathbb{K})^2, \quad \|AB\| \leq \|A\| \|B\|$.

- Exemple fondamental : normes dites **subordonnées** associées à une norme vectorielle :

Théorème et Définition

Soit $\|\cdot\|$ une norme vectorielle sur \mathbb{K}^n . Pour toute matrice $A \in \mathbb{M}_{n \times n}(\mathbb{K})$, on définit $\|\cdot\|_M : \mathbb{M}_{n \times n}(\mathbb{K}) \rightarrow \mathbb{R}_+$ par $\|A\|_M = \sup_{x \in \mathbb{K}^n \setminus \{0\}} \frac{\|Ax\|}{\|x\|}$. Alors $\|\cdot\|_M$ est une norme matricielle. Elle est dite **norme subordonnée à la norme vectorielle $\|\cdot\|$** .

Normes subordonnées classiques

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

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

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

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

où :

- $A^* = \bar{A}^T$ désigne la matrice adjointe de A
- $\rho(M)$ désigne le rayon spectral d'une matrice M cad le maximum des modules des valeurs propres de M .

Conditionnement d'une matrice : exemple

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

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

- On remarque que :
 - A est symétrique
 - $\det(A) = 1$
 - la solution de (S) est donnée par $x = (1 \ 1 \ 1 \ 1)^T$

Premier cas : b est perturbé (1)

- Perturbons légèrement le second membre b et considérons

$$b' = \begin{pmatrix} 32,1 \\ 22,9 \\ 33,1 \\ 30,9 \end{pmatrix}$$

- Si on résout le système (S') : $Ax' = b'$, on trouve $x' = (9,2 \quad -12,6 \quad 4,5 \quad -1,1)^T$

↪ La **petite perturbation sur le second membre b** entraîne donc une **forte perturbation sur la solution du système**

- D'une manière générale, pour $Ax = b$ et $A(x + \delta x) = b + \delta b$:

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

Premier cas : b est perturbé (2)

- **Majoration optimale** : il n'existe pas de borne plus petite qui soit valable pour tout système !

- $A = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1}{2} \end{pmatrix}$, $b = (1 \ 0)^T$, et $\delta b = (0 \ \frac{1}{2})^T$

\rightsquigarrow Solution de $Ax = b$: $x = (1 \ 0)^T$ et celle de $A\delta x = \delta b$ est $\delta x = (0 \ 1)^T$

$\rightsquigarrow \frac{\|\delta x\|}{\|x\|} = 1$, $\frac{\|\delta b\|}{\|b\|} = \frac{1}{2}$

\rightsquigarrow Or $\|A^{-1}\| \cdot \|A\| = 2$ donc la borne est atteinte.

Deuxième cas : A est perturbée

- Si on perturbe légèrement la matrice A :

$$A'' = \begin{pmatrix} 10 & 7 & 8,1 & 7,2 \\ 7,08 & 5,04 & 6 & 5 \\ 8 & 5,98 & 9,89 & 9 \\ 6,99 & 4,99 & 9 & 9,98 \end{pmatrix},$$

\rightsquigarrow Solution de (S'') : $A'' x'' = b$: $x'' = (-81 \quad 107 \quad -34 \quad 22)^T$.

- D'une manière générale, pour $Ax = b$ et $(A + \Delta A)(x + \delta x) = b$:

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

Définition

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

- Ce nombre mesure la **sensibilité** de la solution par rapport aux données du problème
- Une matrice est :
 - bien conditionnée si $\text{Cond}(A) \approx 1$
 - mal conditionnée si $\text{Cond}(A) \gg 1$

Exemples

- Matrices **bien conditionnées**

$$A = \begin{pmatrix} 4 & 1 & 0 & \dots & 0 \\ 1 & 4 & 1 & \ddots & \vdots \\ 0 & 1 & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 1 \\ 0 & \dots & 0 & 1 & 4 \end{pmatrix} \quad \text{Cond}_\infty(A) \leq 3$$

- Matrices **mal conditionnées** : matrices H_n et V_n

$$H_n = \left(\frac{1}{i+j-1} \right)_{1 \leq i, j \leq n}, \quad V_n = \left(\binom{j}{n}^{i-1} \right)_{1 \leq i, j \leq n}$$

n	$\text{Cond}(H_n)$	$\text{Cond}(V_n)$
2	27	8
4	$2,8 \cdot 10^4$	$5,6 \cdot 10^2$
6	$2,9 \cdot 10^7$	$3,7 \cdot 10^4$

Proposition

Soit A une matrice réelle et considérons la norme matricielle subordonnée $\|A\|_2 = \sqrt{\rho(AA^*)}$. On a

$$\text{Cond}_2(A) = \|A^{-1}\|_2 \cdot \|A\|_2 = \sqrt{\frac{\rho(AA^T)}{\sigma(AA^T)}},$$

où $\sigma(M)$ désigne le minimum des modules des valeurs propres de M .
En particulier, si A est symétrique, alors on a $\text{Cond}_2(A) = \frac{\rho(A)}{\sigma(A)}$.

Estimation théorique de l'erreur a priori (1)

Théorème (Cas b perturbé)

Soit $A \in \mathbb{M}_{n \times n}(\mathbb{R})$ inversible et $b \in \mathbb{R}^n$ tels que $Ax = b$ et $A(x + \delta x) = b + \delta b$ avec $x \neq 0$. Alors on a :

$$\frac{1}{\text{Cond}(A)} \cdot \frac{\|\delta b\|}{\|b\|} \leq \frac{\|\delta x\|}{\|x\|} \leq \text{Cond}(A) \cdot \frac{\|\delta b\|}{\|b\|},$$

Théorème (Cas A perturbée)

Soit $A \in \mathbb{M}_{n \times n}(\mathbb{R})$ inversible, $b \in \mathbb{R}^n$ et $\Delta A \in \mathbb{M}_{n \times n}(\mathbb{R})$ tels que $\|A^{-1}\| \cdot \|\Delta A\| < 1$. Alors $A + \Delta A$ est inversible. De plus si on suppose $Ax = b$ et $(A + \Delta A)(x + \delta x) = b$ avec $x \neq 0$, alors on a :

$$\frac{\|\delta x\|}{\|x\|} \leq \frac{\text{Cond}(A) \cdot \frac{\|\Delta A\|}{\|A\|}}{1 - \text{Cond}(A) \cdot \frac{\|\Delta A\|}{\|A\|}}.$$

Estimation théorique de l'erreur a priori (2)

Théorème (Cas A et b perturbés)

Soit $A \in \mathbb{M}_{n \times n}(\mathbb{R})$ inversible, $b \in \mathbb{R}^n$ et $\Delta A \in \mathbb{M}_{n \times n}(\mathbb{R})$ vérifiant $\|A^{-1}\| \cdot \|\Delta A\| < 1$. Si l'on suppose que $Ax = b$ et $(A + \Delta A)(x + \delta x) = b + \delta b$ avec $x \neq 0$, alors on a :

$$\frac{\|\delta x\|}{\|x\|} \leq \frac{\text{Cond}(A)}{1 - \text{Cond}(A) \cdot \frac{\|\Delta A\|}{\|A\|}} \left(\frac{\|\delta b\|}{\|b\|} + \frac{\|\Delta A\|}{\|A\|} \right).$$

Estimation théorique de l'erreur a posteriori

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

Théorème

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

- Conditionnement est grand \Rightarrow erreur relative peut être grande
- Difficile à utiliser car en général conditionnement inconnu !
- C approximation de A^{-1} (par Gauss), $R = AC - \mathbb{I}_n$

Théorème

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

Chapitre 4

Résolution d'un système d'équations linéaires (Partie 2) : méthodes itératives

- Les méthodes directes exigent un nombre de flops de l'ordre de n^3 lorsque n devient grand ce qui les rend lentes !

↪ Méthodes itératives deviennent indispensables dès que la taille n du système est grande, $n \geq 1000$

- De tels systèmes apparaissent par exemple dans les techniques de résolution numérique d'équations aux dérivées partielles
- Les matrices des systèmes obtenus sont en général creuses (cad qu'elles ont beaucoup de 0) et (semi) définies positives
- Les méthodes itératives s'appliquent sur \mathbb{R} ou \mathbb{C} mais pas \mathbb{F}_p !

Exemple introductif (1)

- Donnée : une fonction $f : \mathbb{R}^2 \rightarrow \mathbb{R}$
- Problème : trouver une solution approchée $\tilde{u} : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ de :

$$\begin{cases} -\Delta \tilde{u} = f, & \forall (x, y) \in \Omega =]0, 1[\times]0, 1[, \\ \tilde{u} = 0, & \forall (x, y) \in \partial\Omega, \end{cases}$$

où :

- $\Delta \tilde{u} = \frac{\partial^2 \tilde{u}}{\partial x^2} + \frac{\partial^2 \tilde{u}}{\partial y^2}$ désigne le laplacien de la fonction \tilde{u}
- $\partial\Omega$ la frontière de Ω

Exemple introductif (2)

- Discrétisation de $\Omega =]0, 1[\times]0, 1[$ de pas h
- ? fonction étagée u t.q. $u \rightarrow \tilde{u}$ lorsque $h \rightarrow 0$

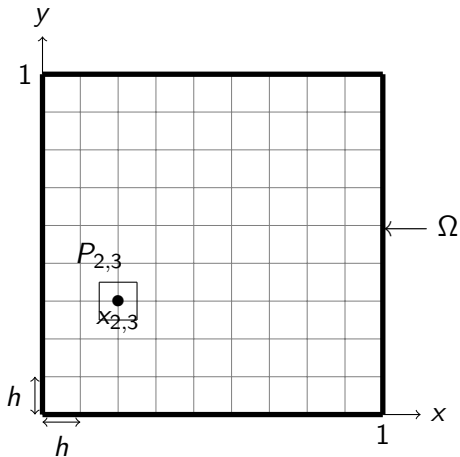
$$h = \frac{1}{n+1}, \quad u = \sum_{i,j} u_{i,j} \chi_{i,j}$$

$\chi_{i,j}$ fonction caractéristique du pavé

$$P_{i,j} =](i - \frac{1}{2})h, (i + \frac{1}{2})h[\times](j - \frac{1}{2})h, (j + \frac{1}{2})h[$$

On note alors $x_{i,j} = (ih, jh)$ les nœuds du quadrillage.

Exemple introductif (3)



Exemple introductif (4)

- Définition de la dérivée d'une fonction : h suffisamment petit :

$$\frac{\partial \tilde{u}}{\partial x} \approx \frac{u(x + \frac{h}{2}, y) - u(x - \frac{h}{2}, y)}{h},$$

$$\frac{\partial^2 \tilde{u}}{\partial x^2} \approx \frac{u(x + h, y) - 2u(x, y) + u(x - h, y)}{h^2}$$

Approximations par différences finies

$$f = \sum_{i,j} f_{i,j} \chi_{i,j} \rightsquigarrow \begin{cases} 4 u_{i,j} - u_{i-1,j} - u_{i+1,j} - u_{i,j-1} - u_{i,j+1} = h^2 f_{i,j}, \\ u_{0,j} = u_{n+1,j} = u_{i,0} = u_{i,n+1} = 0, \end{cases}$$

- schéma numérique dit **implicite**

Exemple introductif (5)

- Système linéaire associé :

$$M = \begin{pmatrix} 4 & -1 & 0 & \dots & 0 \\ -1 & 4 & -1 & \ddots & \vdots \\ 0 & -1 & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -1 \\ 0 & \dots & 0 & -1 & 4 \end{pmatrix} \in \mathbb{M}_{n \times n}(\mathbb{R})$$

- $X_j = (u_{1,j} \quad u_{2,j} \quad \dots \quad u_{n,j})^T$, $X = (X_1^T \quad X_2^T \quad \dots \quad X_n^T)^T$
- $F_j = (f_{1,j} \quad f_{2,j} \quad \dots \quad f_{n,j})^T$, $F = (F_1^T \quad F_2^T \quad \dots \quad F_n^T)^T$
- En définissant de plus $X_0 = X_{n+1} = 0$, le système précédent s'écrit:

$$-X_{j-1} + M X_j - X_{j+1} = h^2 F_j, \quad 1 \leq j \leq n$$

Exemple introductif (6)

$$\rightsquigarrow AX = h^2 F, \quad A = \begin{pmatrix} M & -\mathbb{I}_n & 0 & \dots & 0 \\ -\mathbb{I}_n & M & -\mathbb{I}_n & \ddots & \vdots \\ 0 & -\mathbb{I}_n & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -\mathbb{I}_n \\ 0 & \dots & 0 & -\mathbb{I}_n & M \end{pmatrix} \in \mathbb{M}_{n^2 \times n^2}(\mathbb{R})$$

- A est symétrique réelle, définie positive (donc en particulier inversible).

\rightsquigarrow Système linéaire tridiagonal par blocs de grande taille
($h \rightarrow 0$ équivaut à $n \rightarrow \infty$)

Modèle général d'un schéma itératif (1)

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

$$(S) \iff Bx + c = x$$

$B \in \mathbb{M}_{n \times n}(\mathbb{K})$ et $c \in \mathbb{K}^n$ bien choisis cad $\mathbb{I} - B$ inversible et $c = (\mathbb{I} - B) A^{-1} b$

- Exemple : $A = M - N$ (M inversible), $B = M^{-1} N$ et $c = M^{-1} b$

Modèle général d'un schéma itératif (2)

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

$$(*) \quad x^{(k+1)} = B x^{(k)} + c, \quad k = 1, 2, \dots$$

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

Convergence (1)

Définition

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

Lemme

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

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

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

Convergence (2)

Théorème

Les assertions suivantes sont équivalentes :

- (i) (\star) est convergente ;
- (ii) Pour tout $y \in \mathbb{K}^n$, $\lim_{k \rightarrow +\infty} B^k y = 0$;
- (iv) Pour toute norme matricielle $\|\cdot\|$ sur $\mathbb{M}_{n \times n}(\mathbb{K})$, on a $\lim_{k \rightarrow +\infty} \|B^k\| = 0$.

- En pratique, caractérisations difficiles à vérifier \rightsquigarrow

Théorème

Les assertions suivantes sont équivalentes :

- (i) (\star) est convergente ;
- (ii) $\rho(B) < 1$, où $\rho(B)$ désigne le rayon spectral de la matrice B ;
- (iii) Il existe une norme matricielle $\|\cdot\|$ sur $\mathbb{M}_{n \times n}(\mathbb{K})$ subordonnée à une norme vectorielle $\|\cdot\|$ sur \mathbb{K}^n telle que $\|B\| < 1$.

Vitesse de convergence (1)

Définition

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

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

Définition

Considérons deux méthodes itératives convergentes

$$(1) \quad x^{(k+1)} = B_1 x^{(k)} + c_1, \quad k = 1, 2, \dots,$$

$$(2) \quad x^{(k+1)} = B_2 x^{(k)} + c_2, \quad k = 1, 2, \dots$$

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

Vitesse de convergence (2)

Définition

On appelle *taux asymptotique de convergence* le nombre

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

Théorème

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

Les méthodes itératives classiques

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

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

- On montre alors que $\mathbb{I} - B$ inversible et $c = (\mathbb{I} - B)^{-1}b$

Trois exemples classiques (1)

- Dans ce cours, 3 exemples classiques : **les méthodes de Jacobi, Gauss-Seidel et de relaxation**
- Point de départ : décomposition de $A = (a_{i,j})_{1 \leq i,j \leq n}$ sous la forme $A = D - E - F$ avec :
 - $D = (d_{i,j})_{1 \leq i,j \leq n}$ diagonale, telle que $d_{i,i} = a_{i,i}$ et $d_{i,j} = 0$ pour $i \neq j$;
 - $E = (e_{i,j})_{1 \leq i,j \leq n}$ triangulaire inférieure **stricte** telle que $e_{i,j} = -a_{i,j}$ si $i > j$ et $e_{i,j} = 0$ si $i \leq j$;
 - $F = (f_{i,j})_{1 \leq i,j \leq n}$ triangulaire supérieure **stricte** telle que $f_{i,j} = -a_{i,j}$ si $i < j$ et $f_{i,j} = 0$ si $i \geq j$;

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

$$\underbrace{\begin{pmatrix} 2 & -1 & 1 \\ 2 & 2 & 2 \\ -1 & -1 & 2 \end{pmatrix}}_A = \underbrace{\begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix}}_D - \underbrace{\begin{pmatrix} 0 & 0 & 0 \\ -2 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}}_E - \underbrace{\begin{pmatrix} 0 & 1 & -1 \\ 0 & 0 & -2 \\ 0 & 0 & 0 \end{pmatrix}}_F$$

Trois exemples classiques (2)

- On suppose D inversible
 - Méthode de **Jacobi** : $M = D, N = E + F$;
 - Méthode de **Gauss-Seidel** : $M = D - E, N = F$;
 - Méthode de **relaxation** : $M = \frac{1}{\omega}(D - \omega E), N = \left(\frac{1-\omega}{\omega}\right) D + F$
avec ω paramètre réel non nul.
- Gauss-Seidel est un cas particulier de relaxation pour $\omega = 1$.

Méthode de Jacobi : description

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

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

Définition

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

Jacobi : mise en œuvre et complexité arithmétique

- ? nombre de flops pour calculer $x^{(k+1)}$ à partir de $x^{(k)}$
- On a $Dx^{(k+1)} = (E + F)x^{(k)} + b$ donc pour tout $i = 1, \dots, n$,
 $(Dx^{(k+1)})_i = ((E + F)x^{(k)})_i + b_i$ cad

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

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

\rightsquigarrow Pour K itérations, on aura besoin de $K n(2n - 1)$ flops !

- Comparaison $n = 1000$: Gauss $6,6 \cdot 10^8$ et 100 it. de Jacobi $2 \cdot 10^8$

Théorème

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

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

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

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

Méthode de Gauss-Seidel : description

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

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

Définition

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

Gauss-Seidel : mise en œuvre et complexité arithmétique (1)

- ? nombre de flops pour calculer $x^{(k+1)}$ à partir de $x^{(k)}$
- On a $(D - E)x^{(k+1)} = Fx^{(k)} + b$ donc pour tout $i = 1, \dots, n$, $((D - E)x^{(k+1)})_i = (Fx^{(k)})_i + b_i$ c'est-à-dire

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

ce qui entraîne

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

et pour $i = 2, \dots, n$,

Gauss-Seidel : mise en œuvre et complexité arithmétique (2)

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

↪ Pour K itérations, on aura besoin de $K n(2n - 1)$ flops ! (idem Jacobi)

- Gauss-Seidel **plus intéressante en ce qui concerne la gestion de la mémoire !**

On peut écraser au fur et à mesure la valeur de $x_i^{(k)}$ et ne stocker au cours des calculs qu'un seul vecteur de taille n , e.g., le vecteur $(x_1^{(k+1)} \dots x_i^{(k+1)} x_{i+1}^{(k)} \dots x_n^{(k)})^T$, au lieu de deux vecteurs pour la méthode de Jacobi.

Gauss-Seidel : convergence et exemple

Théorème

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

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

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

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

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

Méthode de la relaxation : description

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

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

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

Définition

La matrice $B_R(\omega) = (D - \omega E)^{-1} [(1 - \omega) D + \omega F]$ s'appelle **la matrice de relaxation associée à A** et ω est le **facteur de relaxation**.
Si $\omega < 1$, on parle de **sous-relaxation**, si $\omega = 1$, on retrouve la **méthode de Gauss-Seidel** et si $\omega > 1$, on parle de **sur-relaxation**.

Théorème

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

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

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

- Valeurs propres et donc convergence dépendent de ω

Cas particulier : matrice symétrique définie positive

Théorème

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

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

converge.

Corollaire

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

Cas particulier : matrice à diagonale strictement dominante

Définition

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

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

- Exemple : matrice du système linéaire obtenu par discrétisation de l'edp $-\Delta u = f$.

Théorème

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

Le gradient conjugué

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

Rappel : matrices définies positives

Définition

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

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

Théorème

Si $A \in \mathbb{M}_{n \times n}(\mathbb{R})$ symétrique est telle que

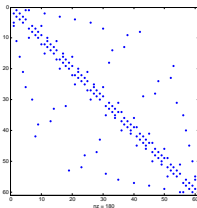
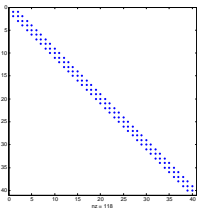
- $a_{ii} > 0$ pour $i = 1, \dots, n$,
- $|a_{ii}| \geq \sum_{j \neq i} |a_{ij}|$ pour $i = 1, \dots, n$ et il existe i_0 tel que $|a_{i_0 i_0}| > \sum_{j \neq i_0} |a_{i_0 j}|$ (diagonale dominante),

alors A est définie positive.

Exemple : $A = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 2 \end{pmatrix}$ est définie positive.

Matrices creuses

- $A \in \mathbb{M}_{n \times n}(\mathbb{R})$ est **creuse** si le nombre d'éléments non nuls est un petit multiple de n .
- Exemples : matrices tridiagonales, matrices d'adjacence, discrétisation d'équations différentielles, ...



Produit scalaire défini par A

Définition

Soit $A \in \mathbb{M}_{n \times n}(\mathbb{R})$ symétrique et définie positive. On définit la fonction $\|\cdot\|_A : \mathbb{R}^n \rightarrow \mathbb{R}_+$ comme $\|x\|_A = \sqrt{x^T A x}$.

Proposition

La fonction $\|\cdot\|_A$ est une norme vectorielle.

Définition

Soit $A \in \mathbb{M}_{n \times n}(\mathbb{R})$ symétrique définie positive. On dit que les vecteurs $u, v \in \mathbb{R}^n$ sont A -conjugués si $u^T A v = 0$.

- $u, v \in \mathbb{R}^n \mapsto u^T A v \in \mathbb{R}$ est un produit scalaire sur \mathbb{R}^n .
- $u, v \in \mathbb{R}^n$ sont A -conjugués s'ils sont orthogonaux par rapport au produit scalaire défini par A , i.e., $u^T A v = 0$.

On considère le problème suivant : **minimiser la fonction**

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

où la matrice A est symétrique et définie positive.

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

Définition

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

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

On a

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

Définition

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

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

Méthodes du gradient

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

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

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

où α_k est tel que

$$\phi(x^{(k+1)}) = \min_{\alpha \in \mathbb{R}} \phi(x^{(k)} + \alpha p^{(k)}),$$

$$\text{d'où } \alpha_k = \frac{r^{(k)T} p^{(k)}}{p^{(k)T} A p^{(k)}}.$$

Proposition

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

Méthode de la plus forte pente

- On choisit $p^{(k)} = r^{(k)} = -\nabla\phi(x^{(k)})$
 \rightsquigarrow direction de plus forte pente pour $\phi(x)$.
- Chaque direction de descente est orthogonale à la précédente, i.e., $p^{(k+1)T}p^{(k)} = 0$.

Théorème

Pour la méthode de la plus forte pente on a, à l'itération k :

$$\|x^* - x^{(k)}\|_A \leq \left(\frac{\text{Cond}_2(A) - 1}{\text{Cond}_2(A) + 1} \right)^k \|x^* - x^{(0)}\|_A.$$

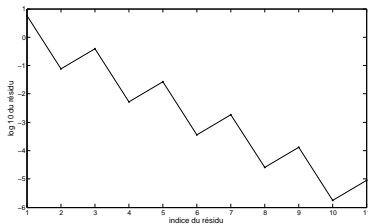
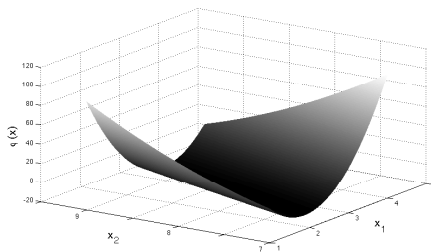
Méthode de la plus forte pente

- Le choix $p^{(k)} = -\nabla\phi(x^{(k)})$ peut paraître intuitivement assez efficace pour minimiser $\phi(x)$, vu qu'on se déplace au long de la direction de plus forte décroissance de la fonction.
- Mais dans certains cas la convergence de cette méthode pourrait être lente, notamment si la matrice A est mal conditionnée.

Exemple :

- A est une matrice symétrique avec valeurs propres $\{e^{-2}, e^4\}$,
- $b = (1 \quad 1)^T$,
- $\text{Cond}_2(A) = e^6 \approx 403,43$.

Méthode de la plus forte pente



- Choix de la direction de descente:

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

où β_k est tel que

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

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

Lemme

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

Théorème

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

$$\phi(x^{(k)}) = \min_{x \in S_k} \phi(x), \quad k \geq 1.$$

Gradient conjugué

Théorème

Soit $r^{(0)} \neq 0$ et $h \geq 1$ tels que $r^{(k)} \neq 0$ pour tout $k \leq h$. Alors pour $k, j = 0, \dots, h$, avec $k \neq j$, on a

$$r^{(k)T} r^{(j)} = 0 \quad \text{et} \quad p^{(k)T} A p^{(j)} = 0.$$

Autrement dit, les résidus forment un ensemble de vecteurs orthogonaux, et les vecteurs direction $p^{(k)}$ forment un ensemble de vecteurs A -conjugués.

Corollaire

Il existe $m \leq n$ tel que $r^{(m)} = 0$. Autrement dit, le gradient conjugué calcule la solution $x^ = A^{-1}b$ en n itérations au plus.*

Gradient conjugué : l'algorithme

Entrée : $A \in \mathbb{M}_{n \times n}(\mathbb{R})$ symétrique et définie positive, $b \in \mathbb{R}^n$, $x^{(0)} \in \mathbb{R}^n$.

Sortie : $x \in \mathbb{R}^n$ tel que $Ax = b$.

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

Théorème

$$\|x^* - x^{(k)}\|_A \leq 2 \left(\frac{\sqrt{\text{Cond}_2(A)} - 1}{\sqrt{\text{Cond}_2(A)} + 1} \right)^k \|x^* - x^{(0)}\|_A$$

- **Critère d'arrêt** : dans la pratique, $\|r^{(k)}\|_2 < \epsilon_M \|b\|_2$, et k borné par $k_{\max} \ll n$.
- Choix du vecteur initial $x^{(0)}$: a priori arbitraire, par ex. le vecteur nul.
- **Complexité** : une multiplication matrice-vecteur par itération, donc complexité **asymptotique équivalent à n^2 flops pour une matrice creuse**.
- Pas nécessaire de stocker A si on sait calculer le produit Ax .

- On souhaite **accélérer la convergence d'une méthode itérative** (e.g., le gradient conjugué).
- La convergence de la méthode du gradient conjugué est très rapide si la matrice A est proche de \mathbb{I} , ou si ses valeurs propres sont bien regroupées.
- Idée : remplacer le système $Ax = b$ par un système équivalent (même solution) mais mieux conditionné.

- Soit $(S) : Ax = b$ avec A symétrique et définie positive, et soit $C \in \mathbb{M}_{n \times n}(\mathbb{R})$ inversible.
- On écrit le système transformé

$$(\tilde{S}) : C^{-1}Ax = C^{-1}b$$

$$(\tilde{S}) : C^{-1}A(C^T)^{-1}C^T x = C^{-1}b$$

$$(\tilde{S}) : \tilde{A}\tilde{x} = \tilde{b},$$

où $\tilde{A} = C^{-1}A(C^{-1})^T$, $\tilde{x} = C^T x$ et $\tilde{b} = C^{-1}b$.

- \tilde{A} est aussi symétrique et définie positive, donc on peut appliquer le gradient conjugué à (\tilde{S}) .

Définition

La matrice $M = CC^T$ est dite *préconditionneur* du système (\tilde{S}) .

Le choix du préconditionneur est fait de manière que :

- la matrice \tilde{A} soit mieux conditionnée que A , ou proche de la matrice identité, pour que le gradient conjugué converge rapidement,
- M soit inversible de manière stable et rapide (idéalement avec complexité asymptotique de l'ordre de n).

Gradient conjugué préconditionné : algorithme

Entrée : $A \in \mathbb{M}_{n \times n}(\mathbb{R})$ symétrique et définie positive, $b \in \mathbb{R}^n$, $x^{(0)} \in \mathbb{R}^n$, un préconditionneur $M \in \mathbb{M}_{n \times n}(\mathbb{R})$ symétrique et défini positif.

Sortie : $x \in \mathbb{R}^n$ tel que $Ax = b$.

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

Le choix d'un préconditionneur est un problème délicat et il existe une vaste littérature à ce sujet.

Exemple 1 : préconditionnement diagonal. On note $A = (a_{i,j})_{1 \leq i,j \leq n}$, $M = (m_{i,j})_{1 \leq i,j \leq n}$ et on définit

$$m_{i,j} = \begin{cases} a_{i,i} & \text{si } i = j, \\ 0 & \text{si } i \neq j. \end{cases}$$

Choix d'un préconditionneur

Exemple 2 : préconditionnement de Cholesky incomplet.
On choisit $M = LL^T$, où $L = (\ell_{i,j})_{1 \leq i,j \leq n}$ est une matrice triangulaire inférieure calculée comme suit:

$$\ell_{i,i} = \sqrt{a_{i,i} - \sum_{r=1}^{i-1} \ell_{i,r}^2}, \quad i = 1, \dots, n$$
$$\ell_{i,j} = \begin{cases} 0 & \text{si } a_{i,j} = 0, \\ \frac{1}{\ell_{i,j}} \left(a_{i,j} - \sum_{r=1}^{j-1} \ell_{i,r} \ell_{j,r} \right) & \text{si } a_{i,j} \neq 0, \end{cases}$$
$$j = 1, \dots, i-1, \quad i = 2, \dots, n.$$

- La matrice L est définie de manière à préserver l'éventuelle structure creuse de A .
- Les éléments non nuls de L sont calculés comme pour le facteur de Cholesky de A .

Et si A n'est pas définie positive ?

- Si la matrice A n'est pas symétrique et définie positive, on ne peut pas appliquer le gradient conjugué au système $Ax = b$.
- Mais on peut appliquer le gradient conjugué au système

$$A^T Ax = A^T b$$

(méthode des **équations normales**).

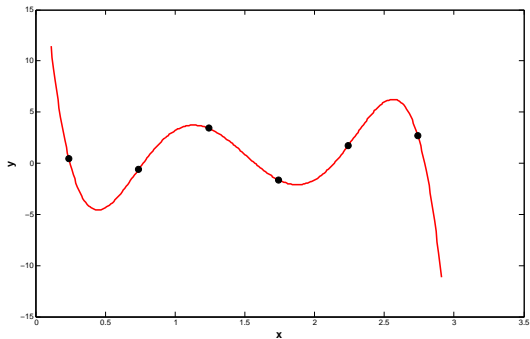
Attention : $\text{Cond}(A^T A) = \text{Cond}(A)^2$.

- Il existe aussi des méthodes itératives adaptées aux matrices non symétriques : l'une des plus utilisées est la méthode GMRES.

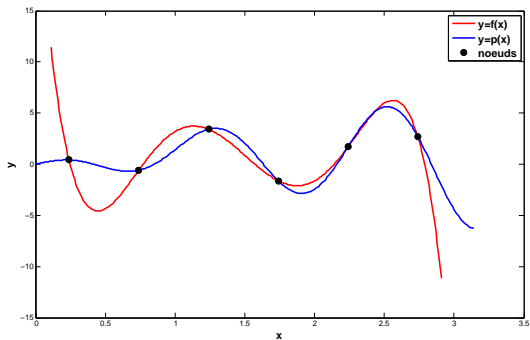
Chapitre 5

Interpolation polynomiale

Problème de l'interpolation



Problème de l'interpolation



Problème de l'interpolation

- $\mathcal{P}_n = \mathbb{R}_n[x]$: ensemble des poly. de degré $\leq n$ et à coeffs dans \mathbb{R} . (espace vect. de dimension $n + 1$ sur \mathbb{R})
- $(a, b) \in \mathbb{R}^2$ ($a < b$) et $f : [a, b] \rightarrow \mathbb{R}$ continue sur $[a, b]$
- On considère $n + 1$ points x_0, \dots, x_n de l'intervalle $[a, b]$ tels que $a \leq x_0 \leq x_1 \leq \dots \leq x_n \leq b$.
- **Problème (I)** $_{m,n}^f$: ? existe $P_m \in \mathcal{P}_m$ tel que $P_m(x_i) = f(x_i), \forall i$.
- $P_m(x) = \lambda_0 + \lambda_1 x + \dots + \lambda_m x^m$ avec les λ_i dans \mathbb{R} , alors ?
 $\lambda_0, \dots, \lambda_m$ tels que :

$$(S) : \begin{pmatrix} 1 & x_0 & x_0^2 & \dots & x_0^m \\ 1 & x_1 & x_1^2 & \dots & x_1^m \\ \vdots & & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^m \end{pmatrix} \begin{pmatrix} \lambda_0 \\ \lambda_1 \\ \vdots \\ \lambda_m \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_m) \end{pmatrix}.$$

Problème de l'interpolation

↪ Système linéaire $n + 1$ équations en $m + 1$ inconnues

Proposition

Le problème d'interpolation $(I)_{m,n}^f$ admet une unique solution ssi $m = n$ et les nœuds $(x_i)_{0 \leq i \leq n}$ sont deux à deux distincts.

- Dans la suite, on s'intéresse au cas où le problème admet une unique solution et on le note $(I)_n^f$: la solution notée $P_n(x; f)$ s'appelle **polynôme d'interpolation de f aux nœuds $(x_i)_{0 \leq i \leq n}$** .
- Problème qui apparaît dans un **contexte expérimental** : calcul des valeurs d'une fonction f inconnue.
- Il est naturel de supposer que l'on connaît un minimum d'information sur la fonction f à interpoler.

Problème de l'interpolation

En pratique, résoudre directement le système (S) n'est pas forcément une bonne idée, car :

- méthode coûteuse ($\mathcal{O}(n^3)$),
- le système est souvent mal conditionné,
- il n'est pas indispensable de calculer les coefficients de $P_n(x; f)$ en base monomiale; il y a d'autres bases de \mathcal{P}_n qui se prêtent mieux à résoudre le problème de l'interpolation.

Remarque: dans plusieurs applications on est surtout intéressé à évaluer $P_n(\tilde{x}; f)$ pour \tilde{x} donné.

Base d'interpolation de Lagrange (1)

Définition

Pour $j \in \{0, \dots, n\}$, le polynôme $L_j^{(n)}$ défini par

$$L_j^{(n)}(x) = \prod_{\substack{i=0 \\ i \neq j}}^n \frac{x - x_i}{x_j - x_i} = \frac{(x - x_0) \cdots (x - x_{j-1})(x - x_{j+1}) \cdots (x - x_n)}{(x_j - x_0) \cdots (x_j - x_{j-1})(x_j - x_{j+1}) \cdots (x_j - x_n)},$$

est appelé *interpolant de base de Lagrange* ou *polynôme de base de Lagrange associé à la suite $(x_i)_{0 \leq i \leq n}$ et relatif au point x_j* .

Proposition

Pour $n \in \mathbb{N}$ fixé, les $(L_j^{(n)}(x))_{0 \leq j \leq n}$ forment une base de l'espace vectoriel \mathcal{P}_n que l'on appelle base de Lagrange.

Proposition

Les interpolants de base de Lagrange vérifient les propriétés suivantes :

- 1 Pour tout $j = 0, \dots, n$, si on note g_j la fonction de $[a, b]$ dans \mathbb{R} définie par $\forall i = 0, \dots, n, g_j(x_i) = \delta_{ij}$, alors
$$P_n(x; g_j) = L_j^{(n)}(x) ;$$
- 2 Si on pose $\pi_{n+1}(x) = \prod_{j=0}^n (x - x_j) \in \mathcal{P}_{n+1}$, alors, pour tout $j = 0, \dots, n$,
$$L_j^{(n)}(x) = \frac{\pi_{n+1}(x)}{(x - x_j) \pi'_{n+1}(x_j)} .$$
- 3 Pour tout $k = 0, \dots, n$,
$$x^k = \sum_{j=0}^n x_j^k L_j^{(n)}(x) .$$

Méthode de Lagrange (1)

- La méthode d'interpolation de Lagrange consiste à écrire le polynôme d'interpolation sur la base de Lagrange.

Théorème

Soit $f : [a, b] \rightarrow \mathbb{R}$ et $n + 1$ nœuds $(x_i)_{0 \leq i \leq n}$ deux à deux distincts. Le polynôme d'interpolation de f aux nœuds $(x_i)_{0 \leq i \leq n}$ s'écrit alors :

$$P_n(x; f) = \sum_{j=0}^n f(x_j) L_j^{(n)}(x).$$

Méthode de Lagrange (2)

- **Intérêt** : pas besoin de résoudre un système linéaire pour écrire le polynôme d'interpolation
- Son expression s'écrit facilement : par exemple, si on choisit les nœuds $-1, 0, 1$, on obtient :

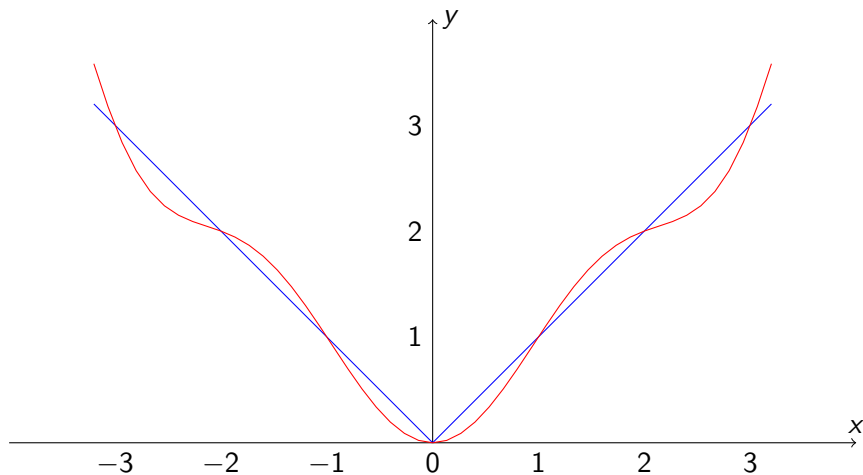
$$\begin{aligned}P_2(x; f) &= f(-1) \frac{(x-0)(x-1)}{(-1-0)(-1-1)} + f(0) \frac{(x+1)(x-1)}{(0+1)(0-1)} + f(1) \frac{(x+1)(x-0)}{(1+1)(1-0)}, \\&= f(-1) \frac{x(x-1)}{2} - f(0)(x^2-1) + f(1) \frac{x(x+1)}{2}, \\&= \frac{f(-1) - 2f(0) + f(1)}{2} x^2 + \frac{f(1) - f(-1)}{2} x + f(0).\end{aligned}$$

Exemple (1)

- $f : [-4, 4] \rightarrow \mathbb{R}, x \mapsto |x|$
- Nœuds $(x_j)_{0 \leq j \leq 8} = (-4, -3, -2, -1, 0, 1, 2, 3, 4)$

$$P_8(x; f) = \sum_{j=0}^8 |x_j| L_j^{(8)}(x) = \frac{533}{420} x^2 - \frac{43}{144} x^4 + \frac{11}{360} x^6 - \frac{1}{1008} x^8.$$

Exemple (2)



- On ne développe pas les $L_j^{(n)}(x)$ sur la base monomiale. Pour évaluer le polynôme d'interpolation de Lagrange, on utilise la formule moins coûteuse (complexité $\mathcal{O}(\frac{3}{2}n^2)$)

$$P_n(x; f) = \pi_{n+1}(x) \sum_{j=0}^n \frac{f(x_j)}{\pi'_{n+1}(x_j) (x - x_j)},$$

- **Principal inconvénient** : rajouter un nœud change complètement les interpolants de base de Lagrange et on doit donc recalculer entièrement le polynôme $P_n(x; f)$.
- **Méthode permet aussi d'interpoler un nuage de points** : on se donne une suite de valeurs discrètes $(b_i)_{0 \leq i \leq n}$ aux nœuds $(x_i)_{0 \leq i \leq n}$ et on cherche un polynôme P_n tel que $P_n(x_i) = b_i$ pour $i = 0, \dots, n$.

Base d'interpolation de Newton (1)

Définition

Les polynômes $N_j^{(n)}$ définis pour $j = 0, \dots, n$ par :

$$\left\{ \begin{array}{l} N_0^{(n)}(x) = 1, \\ N_1^{(n)}(x) = (x - x_0), \\ N_2^{(n)}(x) = (x - x_0)(x - x_1), \\ \vdots \\ N_j^{(n)}(x) = (x - x_0)(x - x_1) \cdots (x - x_{j-1}), \\ \vdots \\ N_n^{(n)}(x) = (x - x_0)(x - x_1) \cdots (x - x_{n-1}), \end{array} \right.$$

sont appelés *polynômes de base de Newton relatifs à la suite de points $(x_i)_{i=0, \dots, n-1}$* .

Base d'interpolation de Newton (2)

- Remarque : là où on avait besoin de $n + 1$ points pour définir les $L_j^{(n)}(x)$, $j = 0, \dots, n$, la **définition des $N_j^{(n)}(x)$, $j = 0, \dots, n$, ne nécessite que n points.**

Proposition

Pour $n \in \mathbb{N}$ fixé, les $(N_j^{(n)}(x))_{0 \leq j \leq n}$ forment une base de l'espace vectoriel \mathcal{P}_n .

Expression de l'interpolant de Newton

- $f : [a, b] \rightarrow \mathbb{R}$ et n nœuds $(x_i)_{0 \leq i \leq n-1}$
- ? $\alpha_i, i = 0, \dots, n$ tels que $P_n(x; f) = \sum_{i=0}^n \alpha_i N_i^{(n)}(x)$. On a :

$$P_n(x_0; f) = \alpha_0 = f(x_0) \implies \alpha_0 = f(x_0)$$

$$P_n(x_1; f) = f(x_0) + \alpha_1 (x_1 - x_0) = f(x_1) \implies \alpha_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

$$P_n(x_2; f) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0} (x_2 - x_0) + \alpha_2 (x_2 - x_0)(x_2 - x_1) = f(x_2)$$

$$\implies \alpha_2 = \frac{\frac{f(x_1) - f(x_2)}{x_1 - x_2} - \frac{f(x_1) - f(x_0)}{x_1 - x_0}}{x_2 - x_1}$$

En posant

$$f[u, v] = \frac{f(u) - f(v)}{u - v},$$

on a alors

$$\alpha_1 = f[x_0, x_1], \quad \alpha_2 = \frac{f[x_0, x_2] - f[x_0, x_1]}{x_2 - x_1} = \frac{f[x_0, x_1] - f[x_1, x_2]}{x_0 - x_2}.$$

Définition

Pour tout $k \in \mathbb{N}$, on appelle *différence divisée d'ordre k de f associée à la suite de points deux à deux distincts $(x_j)_{j \in \mathbb{N}}$* la quantité $f[x_0, x_1, \dots, x_k]$ définie par :

$$f[x_0] = f(x_0), \quad \forall k \in \mathbb{N}^*, \quad f[x_0, x_1, \dots, x_k] = \frac{f[x_0, x_1, \dots, x_{k-1}] - f[x_1, x_2, \dots, x_k]}{x_0 - x_k}.$$

Théorème

$$P_n(x; f) = \sum_{k=0}^n f[x_0, x_1, \dots, x_k] N_k^{(n)}(x).$$

Corollaire

$$P_n(x; f) = P_{n-1}(x; f) + f[x_0, x_1, \dots, x_n] N_n^{(n)}(x).$$

- Déf. de la base d'interpolation de Newton de \mathcal{P}_n ne nécessite que la donnée de n nœuds mais le coefficients $f[x_0, x_1, \dots, x_n]$ de $N_n^{(n)}(x)$ fait intervenir le nœud x_n .
- Calcul du polynôme d'interpolation de f sur la base de Newton relativement simple comparé à celui sur la base de Lagrange.

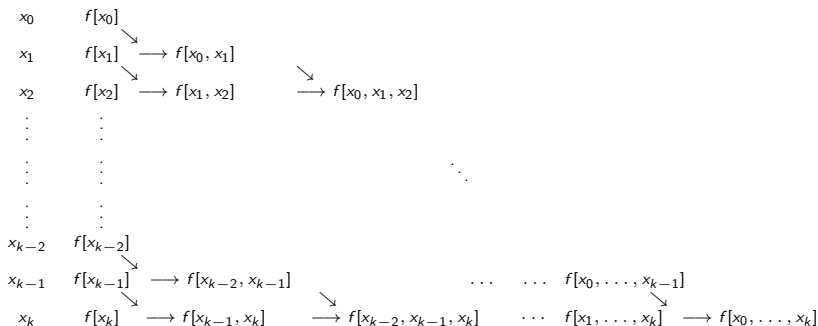
Proposition

$$f[x_0, x_1, \dots, x_k] = \sum_{j=0}^k \frac{f(x_j)}{\prod_{\substack{l=0 \\ l \neq j}}^k (x_j - x_l)} = \sum_{j=0}^k \frac{f(x_j)}{\pi'_{k+1}(x_j)}$$

Corollaire

Soit \mathcal{S}_{k+1} l'ensemble des permutations sur $\{0, 1, \dots, k+1\}$. Pour tout $\sigma \in \mathcal{S}_{k+1}$, on a $f[x_{\sigma(0)}, x_{\sigma(1)}, \dots, x_{\sigma(k)}] = f[x_0, x_1, \dots, x_k]$.

Algorithme de calcul des différences divisées



- Contrairement à Lagrange, l'ajout d'un nouveau nœud n'oblige pas à recalculer toutes les différences divisées : **passer de n à $n + 1$ nœuds demande simplement le calcul de n différences divisées.**

Erreur d'interpolation (1)

Lemme

Soit $(x_i)_{0 \leq i \leq n}$ tels que, pour tout $i = 0, \dots, n$, $x_i \in [a, b]$ et soit $P_n(x; f)$ le polynôme d'interpolation de f aux nœuds $(x_i)_{0 \leq i \leq n}$. Alors, avec les notations précédentes, pour tout $x \in [a, b]$ tel que, pour tout $i = 0, \dots, n$, $x \neq x_i$, on a :

$$f(x) - P_n(x; f) = f[x_0, x_1, \dots, x_n, x] \pi_{n+1}(x).$$

Lemme

Si $f \in C^n([a, b])$ est de classe C^n sur $[a, b]$, alors :

$$\exists \xi \in]a, b[, \quad f[x_0, x_1, \dots, x_n] = \frac{1}{n!} f^{(n)}(\xi).$$

Erreur d'interpolation (2)

Théorème

Soit $(x_i)_{0 \leq i \leq n}$ tels que, pour tout $i = 0, \dots, n$, $x_i \in [a, b]$ et soit $P_n(x; f)$ le polynôme d'interpolation de f aux nœuds $(x_i)_{0 \leq i \leq n}$. Si $f \in C^{n+1}([a, b])$, alors :

$$\forall x \in [a, b], \exists \xi_x \in]a, b[, \quad f(x) - P_n(x; f) = \frac{1}{(n+1)!} f^{(n+1)}(\xi_x) \pi_{n+1}(x).$$

Corollaire

Avec les mêmes hypothèses, on a :

$$\forall x \in [a, b], \quad |f(x) - P_n(x; f)| \leq \frac{|\pi_{n+1}(x)|}{(n+1)!} \sup_{y \in [a, b]} |f^{(n+1)}(y)|.$$

Chapitre 6

Intégration numérique

- On veut approcher de façon numérique la valeur d'intégrales de la forme

$$I(f) = \int_a^b f(x) dx$$

- Remarques :
 - En pratique, on ne connaît pas forcément l'expression symbolique de f ;
 - La plupart des fonctions n'admettent pas de primitives pouvant s'exprimer à l'aide de fonctions élémentaires.

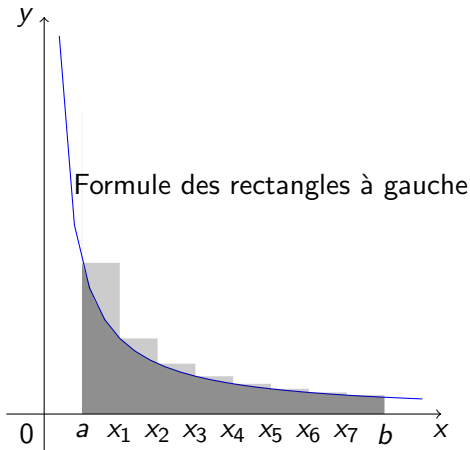
- Hypothèse : fonctions que l'on cherche à intégrer numériquement sont **continues sur l'intervalle $[a, b]$** .
- Soit $x_0 = a < x_1 < x_2 < \dots < x_{n-1} < x_n = b$ une subdivision de l'intervalle $[a, b]$.
- Théorie élémentaire de l'intégration \rightsquigarrow

$$I(f) = \int_a^b f(x)dx = \lim_{n \rightarrow +\infty} \underbrace{\sum_{j=0}^{n-1} f(\xi_j)(x_{j+1} - x_j)}_{\text{Somme de Riemann}}, \quad \forall j, \xi_j \in [x_j, x_{j+1}].$$

- Différents choix des ξ_j mènent aux méthodes classiques

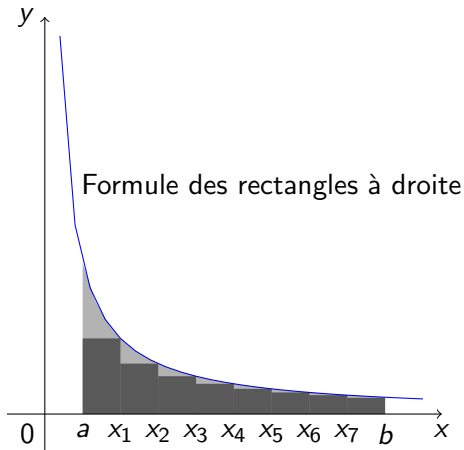
Formule des rectangles à gauche

$$\xi_j = x_j \rightsquigarrow I_{rg}(f) = \sum_{j=0}^{n-1} f(x_j) (x_{j+1} - x_j)$$



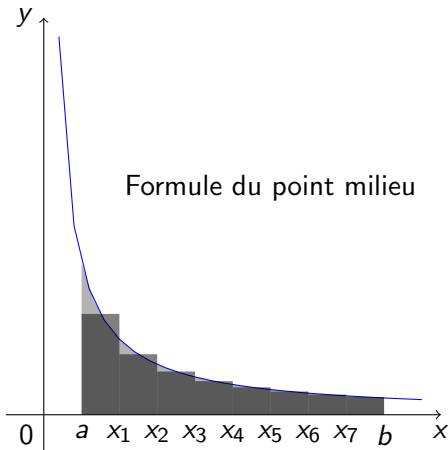
Formule des rectangles à droite

$$\xi_j = x_{j+1} \rightsquigarrow I_{rd}(f) = \sum_{j=0}^{n-1} f(x_{j+1})(x_{j+1} - x_j)$$



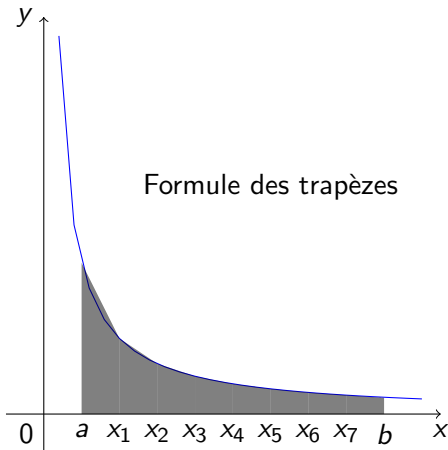
Formule du point milieu

$$\xi_j = \frac{x_j + x_{j+1}}{2} \rightsquigarrow I_{pm}(f) = \sum_{j=0}^{n-1} f\left(\frac{x_j + x_{j+1}}{2}\right) (x_{j+1} - x_j)$$



Méthode des trapèzes

$$I_t(f) = \sum_{j=0}^{n-1} \frac{f(x_j) + f(x_{j+1})}{2} (x_{j+1} - x_j)$$



Liens avec l'interpolation polynomiale

- Les méthodes des rectangles et la méthode du point milieu reviennent à interpoler f sur chaque intervalle $[x_j, x_{j+1}]$ par le polynôme d'interpolation de degré 0 relatif à l'unique nœud ξ_j .
- Ces formules seront donc exactes pour les fonctions constantes sur $[a, b]$ et en particulier pour $f \in \mathcal{P}_0$.
- La méthode des trapèzes revient à interpoler f sur chaque intervalle $[x_j, x_{j+1}]$ par le polynôme d'interpolation de degré 1.
- Cette formule sera donc exacte pour $f \in \mathcal{P}_1$.

Formalisation de l'intégration approchée

- f une fonction de $\mathcal{C}([a, b])$ ev des fonctions continues sur $[a, b]$
- Hypothèse : on connaît au moins les valeurs de f en certains points x_0, x_1, \dots, x_n de l'intervalle $[a, b]$
- On cherche alors une **formule d'intégration approchée** de la forme

$$I(f) = \int_a^b f(x) dx \approx \sum_{k=0}^n \lambda_k f(x_k) = \tilde{I}^{(n)}(f),$$

où les λ_k sont à déterminer.

- Terminologie : on parle aussi de **méthode d'intégration numérique** ou **formule de quadrature**.

Définition

*Une méthode d'intégration numérique est dite **d'ordre N** ($N \in \mathbb{N}$) si elle est exacte sur \mathcal{P}_N .*

- Exemple : la méthode des rectangles à gauche ou à droite et la méthode du point milieu sont d'ordre 0 et celle des trapèzes est d'ordre 1.

Méthodologie et Erreur d'intégration

- En pratique : connaissant les valeurs de f aux points x_0, \dots, x_n , on remplace f par le polynôme d'interpolation $\sum_{k=0}^n f(x_k) L_k^{(n)}(x)$ écrit dans la base de Lagrange

↪ Formule d'intégration approchée (exacte sur $\mathcal{P}_n([a, b])$) :

$$\tilde{I}^{(n)}(f) = \sum_{k=0}^n A_k^{(n)} f(x_k), \quad A_k^{(n)} = \int_a^b L_k^{(n)}(x) dx.$$

Théorème

Soit $f \in \mathcal{C}^{n+1}([a, b])$ et $\tilde{I}^{(n)}(f)$ donnée ci-dessus. Alors on a la majoration suivante de l'erreur d'intégration :

$$|I(f) - \tilde{I}^{(n)}(f)| \leq \frac{M_{n+1}}{(n+1)!} \int_a^b |\pi_{n+1}(x)| dx,$$

avec $M_{n+1} = \sup_{x \in [a, b]} |f^{(n+1)}(x)|$ et $\pi_{n+1}(x) = \prod_{j=0}^n (x - x_j)$.

Formules de Newton-Cotes (1)

- Problème : calcul des $A_k^{(n)} = \int_a^b L_k^{(n)}(x) dx$
- Hypothèses : $x_0 = a$, $x_n = b$, $n \geq 1$ et points d'interpolation équidistants

Proposition

Pour $k = 0, 1, \dots, n$, on a :

(i)

$$A_k^{(n)} = \frac{(b-a)}{n} \frac{(-1)^{n-k}}{k!(n-k)!} \int_0^n \prod_{\substack{j=0 \\ j \neq k}}^n (y-j) dy.$$

(ii) $A_{n-k}^{(n)} = A_k^{(n)}$.

Formules de Newton-Cotes (2)

- **Cas $n = 1$** , on obtient $A_0^{(1)} = A_1^{(1)} = \frac{b-a}{2}$ d'où

$$\tilde{I}^{(1)}(f) = \frac{b-a}{2} (f(a) + f(b)) \quad (\text{formule des trapèzes})$$

- **Cas $n = 2$** , on obtient $A_0^{(2)} = A_2^{(2)} = \frac{b-a}{6}$ et $A_1^{(2)} = \frac{4(b-a)}{6}$ d'où

$$\tilde{I}^{(2)}(f) = \frac{b-a}{6} \left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right) \quad (\text{formule de Simpson})$$

Calcul pratique des coefficients $A_k^{(n)}$

- En pratique, **utiliser le fait que $\tilde{I}^{(n)}$ est exacte sur $\mathcal{P}_n([a, b])$**
- Cas $n = 1$, $a = -1$ et $b = 1$: $\tilde{I}^{(1)}(f) = A_0^{(1)} f(-1) + A_1^{(1)} f(1)$
- $\tilde{I}^{(1)}$ exacte sur $\mathcal{P}_1([-1, 1])$ donc

$$\tilde{I}^{(1)}(1) = I(1) = \int_{-1}^1 1 \, dx = 2, \quad \tilde{I}^{(1)}(x) = I(x) = \int_{-1}^1 x \, dx = 0$$

$$\rightsquigarrow \begin{cases} A_0^{(1)} + A_1^{(1)} & = 2, \\ -A_0^{(1)} + A_1^{(1)} & = 0, \end{cases}$$

d'où $A_0^{(1)} = A_1^{(1)} = 1$.

- Remarque : cette formule n'est pas exacte sur $\mathcal{P}_2([-1, 1])$ puisque $I(x^2) = \int_{-1}^1 x^2 \, dx = \frac{2}{3}$ alors que $\tilde{I}^{(1)}(x^2) = 2$.

Théorème

Considérons l'erreur $\mathcal{E}_n(f) = I(f) - \sum_{i=0}^n A_i^{(n)} f(x_i)$. Alors :

- ① Si n impair et $f \in \mathcal{C}^{n+1}([a, b])$, alors $\exists \xi \in [a, b]$ tel que :

$$\mathcal{E}_n(f) = \left(\frac{b-a}{n}\right)^{n+2} \frac{f^{(n+1)}(\xi)}{(n+1)!} \int_0^n t(t-1)\cdots(t-n) dt.$$

- ② Si n pair et $f \in \mathcal{C}^{n+2}([a, b])$, alors $\exists \xi \in [a, b]$ tel que :

$$\mathcal{E}_n(f) = \left(\frac{b-a}{n}\right)^{n+3} \frac{f^{(n+2)}(\xi)}{(n+2)!} \int_0^n t^2(t-1)\cdots(t-n) dt.$$

- **Cas $n = 2$** : si $f \in \mathcal{C}^4([a, b])$, alors l'erreur d'approximation commise par la **formule de Simpson** vaut $-h^5 \frac{f^{(4)}(\xi)}{90}$ où $h = \frac{b-a}{2}$ et $\xi \in [a, b]$.

Stabilité des méthodes d'intégration (1)

- Mesure la *sensibilité de la méthode aux erreurs de calculs*
- Formule d'intégration approchée $\tilde{I}^{(n)}(f) = \sum_{k=0}^n A_k^{(n)} f(x_k)$
- **Supposons les valeurs calculées des $f(x_k)$ non exactes**

$$\sum_{k=0}^n A_k^{(n)} (f(x_k) + \epsilon_k) - \sum_{k=0}^n A_k^{(n)} f(x_k) = \sum_{k=0}^n A_k^{(n)} \epsilon_k.$$

$$\rightsquigarrow \left| \sum_{k=0}^n A_k^{(n)} \epsilon_k \right| \leq \left(\max_{0 \leq k \leq n} |\epsilon_k| \right) \sum_{k=0}^n |A_k^{(n)}|,$$

et le terme $\sum_{k=0}^n |A_k^{(n)}|$ dépend de la méthode.

Stabilité des méthodes d'intégration (1)

Définition

La formule d'intégration numérique $\tilde{I}^{(n)}(f) = \sum_{k=0}^n A_k^{(n)} f(x_k)$ est dite **stable** s'il existe $M \in \mathbb{R}$ tel que : $\forall n \in \mathbb{N}, \forall (\epsilon_0, \dots, \epsilon_n) \in \mathbb{R}^{n+1}, |\sum_{k=0}^n A_k^{(n)} \epsilon_k| \leq M \max_{0 \leq k \leq n} |\epsilon_k|$.

Théorème

Avec les notations précédentes, une condition nécessaire et suffisante de stabilité est qu'il existe $M \in \mathbb{R}$ (indépendant de n) tel que $\sum_{k=0}^n |A_k^{(n)}| \leq M$.

- Formules de **Newton-Côtes** : pour certaines valeurs de k , $\lim_{n \rightarrow \infty} |A_k^{(n)}| = +\infty$ donc **pour de grandes valeurs de n ces formules ne sont pas stables**

Formules d'intégration composées (1)

- Ceux sont **les plus utilisées en pratique**
- **Principe** : décomposer l'intervalle $[a, b]$ en k intervalles $[a_i, a_{i+1}]$, $i = 0, \dots, k - 1$

$$I(f) = \int_a^b f(x) dx = \sum_{i=0}^{k-1} \underbrace{\int_{a_i}^{a_{i+1}} f(x) dx}_{I_i(f)}$$

- $I_i(f)$ approché par une formule d'intégration numérique
- Remarque : pour la stabilité, il est judicieux de choisir une formule avec un n petit comme par exemple celle de Simpson ($n = 2$)

Formules d'intégration composées (2) : Simpson

- Méthodes composées sont d'autant plus intéressantes que l'erreur d'approximation diminue lorsque la taille de l'intervalle diminue
- Avec la formule de Simpson déjà vu, si l'on subdivise l'intervalle $[a, b]$ en k sous intervalles avec k pair, on obtient la **formule de Simpson composée**

$$\frac{h}{3} \left(f(a_0) + 2 \sum_{i=1}^{k/2-1} f(a_{2i}) + 4 \sum_{i=1}^{k/2} f(a_{2i-1}) + f(a_k) \right),$$

avec $h = \frac{b-a}{k}$, $a_0 = a$, $a_k = b$ et $a_i = a_{i-1} + h$.

- Si $f \in C^4([a, b])$, **erreur d'approximation** $-k h^5 \frac{f^{(4)}(\xi)}{180}$ où $h = \frac{b-a}{k}$ et $\xi \in [a, b]$.

Exemple

On veut calculer numériquement $\int_0^1 e^{-x^2} dx$ avec erreur $< 10^{-6}$.
Combien d'intervalles faut-il utiliser ?

- $f(x) = e^{-x^2}$, $f^{(4)}(x) = e^{-x^2}(16x^4 - 48x^2 + 12)$, $|f^{(4)}(\xi)| \leq 12 \forall \xi \in [0, 1]$
- erreur = $k \left(\frac{b-a}{k} \right)^5 \frac{f^{(4)}(\xi)}{180} = \frac{f^{(4)}(\xi)}{180k^4} \leq \frac{1}{15k^4}$
- on pose $\frac{1}{15k^4} < 10^{-6}$ d'où $k > \left(\frac{10^6}{15} \right)^{1/4} \approx 16,0686$
- donc $k \geq 17$.

- **Formules de quadrature de Gauss**: famille de formules de quadrature assez précises, qui utilisent des polynômes d'interpolation. Les nœuds sont les zéros des polynômes d'interpolation, et/ou des points donnés.
- **Méthodes adaptives**: le nombre de nœuds est choisi suivant le comportement de la fonction.

Plusieurs implémentations sont disponibles:

- `trapz`: méthode des trapèzes, intervalles uniformes,
- `quad`: formule de Simpson, quadrature adaptative,
- `integral`: méthode adaptative globale.

Chapitre 7

Résolution d'équations et de systèmes d'équations non linéaires

Problème considéré

- $f : \mathbb{R} \rightarrow \mathbb{R}$ fonction d'une seule variable réelle
- On cherche à résoudre l'équation $f(x) = 0$ = trouver une valeur approchée \bar{x} d'un réel \tilde{x} vérifiant $f(\tilde{x}) = 0$.
- **Mise en oeuvre pratique** : on se donne une **tolérance** sur la solution cherchée. L'algorithme numérique utilisé doit alors avoir un **critère d'arrêt** dépendant de cette tolérance et nous assurant que la solution calculée a bien la précision recherchée
- 2 possibilités :
 - on sait à l'avance combien d'étapes de l'algorithme sont nécessaires
 - à chaque étape, on vérifie une condition nous permettant d'arrêter le processus après un nombre fini d'étapes

Vitesse de convergence (1)

Définition

Soit $(x_n)_{n \in \mathbb{N}}$ une suite convergente et soit \tilde{x} sa limite.

- 1 On dit que la convergence de $(x_n)_{n \in \mathbb{N}}$ est **linéaire de facteur** $K \in]0, 1[$ s'il existe $n_0 \in \mathbb{N}$ tel que, pour tout $n \geq n_0$,
 $|x_{n+1} - \tilde{x}| \leq K |x_n - \tilde{x}|$.
- 2 On dit que la convergence de $(x_n)_{n \in \mathbb{N}}$ est **superlinéaire d'ordre** $p \in \mathbb{N}$, $p > 1$ s'il existe $n_0 \in \mathbb{N}$ et $K > 0$ tels que, pour tout $n \geq n_0$, $|x_{n+1} - \tilde{x}| \leq K |x_n - \tilde{x}|^p$. Si $p = 2$, on parle de **convergence quadratique** et si $p = 3$ on parle de **convergence cubique**.

- Remarque : K n'est pas unique.
- En pratique il peut être difficile de prouver la convergence d'une méthode d'autant plus qu'il faut tenir compte des erreurs d'arrondis.

Vitesse de convergence (2)

Définition

Soit $(x_n)_{n \in \mathbb{N}}$ une suite convergent vers une limite \tilde{x} . On dit que la convergence de $(x_n)_{n \in \mathbb{N}}$ est **linéaire de facteur K** (resp. **superlinéaire d'ordre p**) s'il existe une suite $(y_n)_{n \in \mathbb{N}}$ convergent vers 0, linéaire de facteur K (resp. superlinéaire d'ordre p) au sens de la définition précédente telle que $|x_n - \tilde{x}| \leq y_n$.

• $d_n = -\log_{10}(|x_n - \tilde{x}|)$ mesure du nbre de décimales exactes de x_n .

\rightsquigarrow Convergence d'ordre $p \Rightarrow$ asymptotiquement, on a

$|x_{n+1} - \tilde{x}| \sim K |x_n - \tilde{x}|^p$ d'où $-d_{n+1} \sim \log_{10}(K) - p d_n$ et donc asymptotiquement x_{n+1} a p fois plus de décimales exactes que x_n

\rightsquigarrow l'ordre p représente asymptotiquement le facteur multiplicatif du nombre de décimales exactes que l'on gagne à chaque itération

\rightsquigarrow Nous avons donc intérêt à ce qu'il soit le plus grand possible.

Méthode de dichotomie : principe

- Méthode de localisation des racines d'une équation $f(x) = 0$ basée sur le théorème des valeurs intermédiaires

Si f est continue sur $[a, b]$ et $f(a)f(b) < 0$, alors il existe $\tilde{x} \in]a, b[$ tel que $f(\tilde{x}) = 0$.

- Principe :

- 1 On part d'un intervalle $[a, b]$ vérifiant la propriété $f(a)f(b) < 0$
- 2 On le scinde en deux intervalles $[a, c]$ et $[c, b]$ avec $c = \frac{a+b}{2}$
- 3 On teste les bornes des nouveaux intervalles (on calcule $f(a)f(c)$ et $f(c)f(b)$) pour en trouver un (au moins) qui vérifie encore la propriété, *i.e.*, $f(a)f(c) < 0$ ou/et $f(c)f(b) < 0$.
- 4 On itère ensuite ce procédé un certain nombre de fois dépendant de la précision que l'on recherche sur la solution.

Méthode de dichotomie : algorithme

Entrées : la fonction¹ f , $(a, b) \in \mathbb{R}^2$ tels que f est continue sur $[a, b]$ et $f(a)f(b) < 0$ et la précision ϵ .

Sortie : x_{k+1} valeur approchée de \tilde{x} solution de $f(\tilde{x}) = 0$ à ϵ près.

- 1 $x_0 \leftarrow a, y_0 \leftarrow b$;
- 2 Pour k de 0 à $E\left(\frac{\ln(b-a) - \ln(\epsilon)}{\ln(2)}\right)$ par pas de 1, faire :
 - Si $f(x_k)f\left(\frac{x_k + y_k}{2}\right) > 0$, alors $x_{k+1} \leftarrow \frac{x_k + y_k}{2}, y_{k+1} \leftarrow y_k$;
 - Si $f(x_k)f\left(\frac{x_k + y_k}{2}\right) < 0$, alors $x_{k+1} \leftarrow x_k, y_{k+1} \leftarrow \frac{x_k + y_k}{2}$;
 - Sinon retourner $\frac{x_k + y_k}{2}$;
- 3 Retourner x_{k+1} .

¹Il suffit en fait de connaître un moyen d'évaluer les valeurs de la fonction

Méthode de dichotomie : remarques et preuve de l'algo.

- **Remarques sur l'algorithme précédent :**
 - Il construit une suite de segments emboîtés contenant tous \tilde{x}
 - À chaque passage dans la boucle : une évaluation de f
 - En pratique, avec les arrondis, > 0 et < 0 ne veulent rien dire !

Théorème

Le nombre minimum d'itérations de la méthode de dichotomie nécessaire pour approcher \tilde{x} à ϵ près est $E\left(\frac{\ln(b-a)-\ln(\epsilon)}{\ln(2)}\right) + 1$, où $E(x)$ désigne la partie entière d'un réel x .

Proof.

$$\frac{b-a}{2^n} \leq \epsilon \Leftrightarrow n \geq \frac{\ln(b-a)-\ln(\epsilon)}{\ln(2)}.$$



Proposition

La convergence de la dichotomie est linéaire de facteur $\frac{1}{2}$.



Exemple

On cherche un zéro de $f(x) = x^3 + 4x \cos(x) - 2$ sur $[0, 1]$,

$$\epsilon = 10^{-3}.$$

k	x_k	$f(x_k)$
1	0,5000000	-0,1198349
2	0,7500000	0,6169415
3	0,6250000	0,2715483
4	0,5625000	$0,8130836 \cdot 10^{-1}$
5	0,5312500	$-0,1794720 \cdot 10^{-1}$
6	0,5468750	$0,3201580 \cdot 10^{-1}$
7	0,5390625	$0,7117271 \cdot 10^{-2}$
8	0,5351563	$0,5393982 \cdot 10^{-2}$
9	0,5371094	$0,8668900 \cdot 10^{-3}$
10	0,5361328	$-0,2263069 \cdot 10^{-2}$

En effet $E \left(\frac{\ln(1-0) - \ln(10^{-3})}{\ln(2)} \right) + 1 = 10.$

Méthode du point fixe (ou approximations successives)

Définition

Soit $g : \mathbb{R} \rightarrow \mathbb{R}$. On dit que $x \in \mathbb{R}$ est un point fixe de g si $g(x) = x$.

- Principe : associer à l'équation $f(x) = 0$ une équation de point fixe $g(x) = x$ de sorte que trouver une solution de $f(x) = 0$ équivaut à trouver un point fixe de g .

Lemme

Soit $(x_n)_{n \in \mathbb{N}}$ la suite définie par $x_0 \in \mathbb{R}$ donné et $x_{n+1} = g(x_n)$. Si $(x_n)_{n \in \mathbb{N}}$ est convergente et g est continue, alors la limite de $(x_n)_{n \in \mathbb{N}}$ est un point fixe de g .

Définition

Soit $g : \Omega \subseteq \mathbb{R} \rightarrow \mathbb{R}$. On dit que g est *lipschitzienne sur Ω de constante de Lipschitz γ (ou γ -lipschitzienne)* si pour tout $(x, y) \in \Omega^2$, on a $|g(x) - g(y)| \leq \gamma |x - y|$. On dit que g est *strictement contractante sur Ω* si g est γ -lipschitzienne sur Ω avec $\gamma < 1$.

Proposition

Soit g une fonction dérivable sur l'intervalle $[a, b]$. Si sa dérivée g' vérifie $\max_{x \in [a, b]} |g'(x)| = L < 1$, alors g est strictement contractante sur $[a, b]$ de constante de Lipschitz L .

Théorème du point fixe

Théorème

Soit g une application strictement contractante sur un intervalle $[a, b] \subset \mathbb{R}$ de constante de Lipschitz $\gamma < 1$. Supposons que l'intervalle $[a, b]$ soit stable sous g , i.e., $g([a, b]) \subseteq [a, b]$ ou encore pour tout $x \in [a, b]$, $g(x) \in [a, b]$. Alors g admet un unique point fixe $x^* \in [a, b]$ et la suite définie par $x_{n+1} = g(x_n)$ converge linéairement de facteur γ vers x^* pour tout point initial $x_0 \in [a, b]$. De plus,

$$\forall n \in \mathbb{N}, |x_n - x^*| \leq \frac{\gamma^n}{1 - \gamma} |x_1 - x_0|.$$

- Erreur d'autant plus petite que γ est proche de 0

- De plus $\forall n \in \mathbb{N}, |x_n - x^*| \leq \frac{\gamma}{1 - \gamma} |x_n - x_{n-1}|$

Si $\gamma \leq \frac{1}{2}$, alors $|x_n - x^*| \leq |x_n - x_{n-1}| \rightsquigarrow$ **test d'arrêt**

$|x_n - x_{n-1}| < \epsilon$ qui certifiera une précision ϵ sur le résultat

Proposition

Soit $x^* \in [a, b]$ un point fixe d'une fonction $g \in \mathcal{C}^1([a, b])$.

- Si $|g'(x^*)| < 1$, alors il existe un intervalle $[\alpha, \beta] \subseteq [a, b]$ contenant x^* pour lequel la suite définie par $x_0 \in [\alpha, \beta]$ et $x_{n+1} = g(x_n)$ converge vers x^* ;
- Si $|g'(x^*)| > 1$, alors pour tout $x_0 \neq x^*$, la suite définie par x_0 et $x_{n+1} = g(x_n)$ ne converge pas vers x^* ;
- Si $|g'(x^*)| = 1$, on ne peut pas conclure.

• En pratique, on estime $g'(x^*)$

Si $\overline{|g'(x^*)|} > 1$, alors on élimine la méthode,

Si $\overline{|g'(x^*)|} < 1$, on cherche un intervalle $[\alpha, \beta] \subseteq [a, b]$ dans lequel $\max_{x \in [\alpha, \beta]} |g'(x)| < 1$ et $g([\alpha, \beta]) \subseteq [\alpha, \beta]$.

Première méthode de résolution de $f(x) = 0$

- Posons $g(x) = x - f(x)$
- Thm du point fixe \Rightarrow CS pour que g admette un point fixe dans $[a, b]$: g contractante sur $[a, b]$ de constante de Lipschitz $\gamma < 1$ de et $[a, b]$ stable sous g

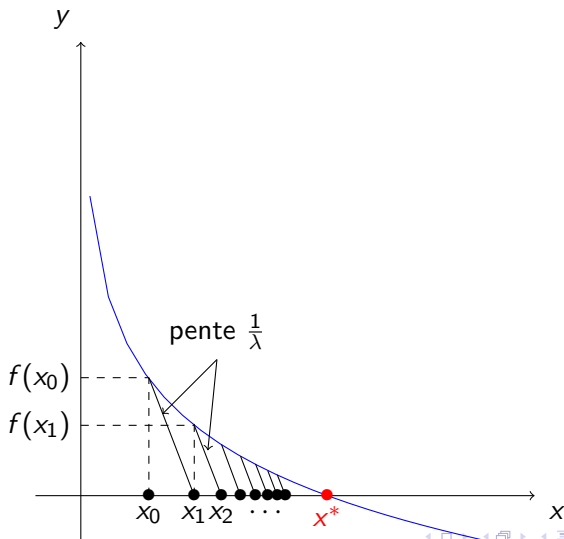
$$\rightsquigarrow \forall x \in [a, b], \quad |g'(x)| < \gamma \iff |1 - f'(x)| < \gamma$$

- Avec $g(x) = x - \lambda f(x)$, on obtient :

$$\forall x \in [a, b], \quad |1 - \lambda f'(x)| < \gamma < 1$$

Première méthode de résolution de $f(x) = 0$

- Suite des itérés $x_{n+1} = x_n - \lambda f(x_n)$



Première méthode de résolution de $f(x) = 0$

- Suite des itérés $x_{n+1} = x_n - \lambda f(x_n)$
- En effet: la droite de pente $\frac{1}{\lambda}$ qui passe par $(x_n, f(x_n))$ a équation

$$y - f(x_n) = \frac{1}{\lambda}(x - x_n),$$

donc le point d'intersection avec l'axe des abscisses $y = 0$ est donné par

$$-\lambda f(x_n) = x - x_n \Rightarrow x = x_n - \lambda f(x_n).$$

Proposition

On considère l'équation $g(x) = x$ où g est une fonction au moins $p + 1$ fois dérivable avec $p \geq 1$. Supposons que les hypothèses du théorème du point fixe soient vérifiées de sorte que g admette un unique point fixe $x^ \in [a, b]$. Si $g'(x^*) = g''(x^*) = \dots = g^{(p)}(x^*) = 0$ et $g^{(p+1)}(x^*) \neq 0$, alors la convergence de la méthode $x_{n+1} = g(x_n)$ est superlinéaire d'ordre $p + 1$.*

Exemple

On reprend l'exemple précédent: calculer le zéro de $f(x) = x^3 + 4x \cos(x) - 2$ sur $[0; 1]$.

- La dérivée de $f(x)$ est $f'(x) = 3x^2 + 4 \cos(x) - 4x \sin(x) > 0$.
- On a $\max_{[0;1]} |f'(x)| = f'(0) = 4$, \rightsquigarrow méthode convergente pour $\lambda < \frac{1}{2}$.
- Pour avoir 5 décimales justes:

λ	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{5}$	$\frac{1}{6}$
n. itér.	25	5	7	11	15

(On observe que $f'(x^*) \approx 3,20$).

- À partir de $x_0 = 0$:

k	$\lambda = \frac{1}{3}$	$\lambda = \frac{1}{4}$
1	0,6666666	0,5000000
2	0,5360014	0,5299587
3	0,5368957	0,5354853
4	0,5368347	0,5365698
5	0,5368388	0,5367854
6		0,5368283
7		0,5368369

Méthode de Newton (1)

- Revenons à

$$\forall x \in [a, b], \quad |1 - \lambda f'(x)| < \gamma < 1$$

- La méthode convergera d'autant plus vite que γ est petite

\rightsquigarrow Idée : poser $\lambda = \frac{1}{f'(x)}$ cad $g(x) = x - \frac{f(x)}{f'(x)}$.

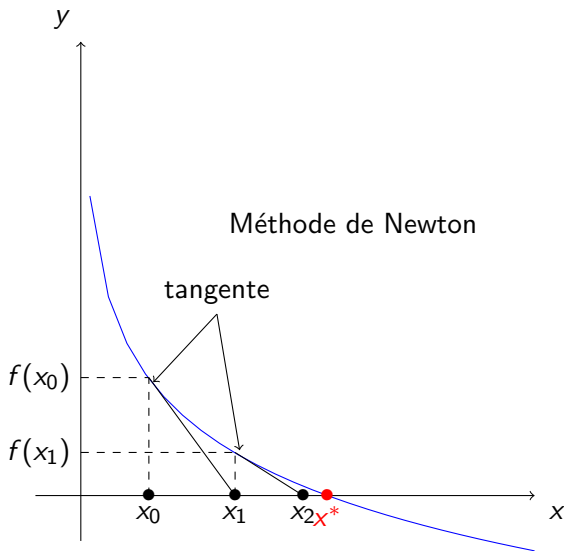
Définition

La *fonction d'itération de Newton associée à l'équation $f(x) = 0$ sur $[a, b]$* est

$$\mathcal{N} : \begin{cases} [a, b] & \rightarrow \mathbb{R}, \\ x & \mapsto \mathcal{N}(x) = x - \frac{f(x)}{f'(x)}. \end{cases}$$

Cette fonction est définie pour f dérivable sur $[a, b]$ et telle que f' ne s'annule pas sur $[a, b]$.

Méthode de Newton (2)



Théorème

Soit f une fonction de classe \mathcal{C}^2 sur un intervalle $[a, b]$ de \mathbb{R} . On suppose qu'il existe $\tilde{x} \in [a, b]$ tel que $f(\tilde{x}) = 0$ et $f'(\tilde{x}) \neq 0$ (\tilde{x} est un zéro simple de f). Alors il existe $\epsilon > 0$, tel que pour tout $x_0 \in [\tilde{x} - \epsilon, \tilde{x} + \epsilon]$, la suite des itérés de Newton donnée par $x_{n+1} = \mathcal{N}(x_n)$ pour $n \geq 1$ est bien définie, reste dans l'intervalle $[\tilde{x} - \epsilon, \tilde{x} + \epsilon]$ et converge vers \tilde{x} quand n tend vers l'infini. De plus, cette convergence est (au moins) quadratique.

Exemple : calcul de la racine carrée

- Équation $f(x) = 0$ avec $f(x) = x^2 - a$
- On a alors $\mathcal{N}(x) = x - \frac{f(x)}{f'(x)} = x - \frac{x^2 - a}{2x} = \frac{1}{2} \left(x + \frac{a}{x} \right)$.
- Si $a = 2$ et $x_0 = 1$ on obtient :
 - $x_0 = 1,0000000000000000$
 - $x_1 = 1,5000000000000000$
 - $x_2 = 1,4166666666666667$
 - $x_3 = 1,414215686274510$
 - $x_4 = 1.414213562374690$
 - $x_5 = 1,414213562373095$
- MATLAB donne $\sqrt{2} = 1,414213562373095$
- Nombre de décimales justes double approximativement à chaque itération (convergence quadratique)
- Dichotomie sur $[1, 2]$: 51 itérations pour $v. a$, à 10^{-15}

Théorème

Avec les notations, précédentes, si \tilde{x} est un zéro de multiplicité m de f , i.e., $f(x^*) = f'(x^*) = \dots = f^{(m-1)}(x^*) = 0$ et $f^{(m)}(x^*) \neq 0$, alors la méthode itérative définie par $x_{n+1} = \mathcal{N}_m(x_n)$ avec
$$\mathcal{N}_m(x_n) = x - m \frac{f(x)}{f'(x)}$$
 est d'ordre supérieure ou égal à 2.

Théorème

Soit $f \in \mathcal{C}^2([a, b])$ vérifiant :

- $f(a)f(b) < 0$,
- $\forall x \in [a, b], f'(x) \neq 0$,
- $\forall x \in [a, b], f''(x) \neq 0$.

Alors, en choisissant $x_0 \in [a, b]$ tel que $f(x_0)f''(x_0) > 0$, la suite $(x_n)_{n \in \mathbb{N}}$ définie par x_0 et $x_{n+1} = \mathcal{N}(x_n)$ converge vers l'unique solution de $f(x) = 0$ dans $[a, b]$.

Méthode de la sécante (1)

- Newton nécessite le calcul de la dérivée de la fonction f qui peut s'avérer difficile
- Idée : remplacer la dérivée f' de f qui apparait dans la méthode de Newton par une différence divisée

Définition

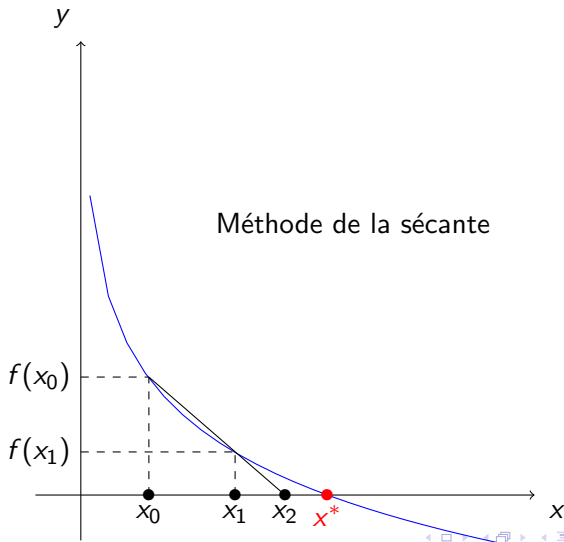
Pour tout $k \in \mathbb{N}$, on appelle *différence divisée d'ordre k de f associée à la suite de points deux à deux distincts $(x_j)_{j \in \mathbb{N}}$* la quantité $f[x_0, x_1, \dots, x_k]$ définie par :

$$f[x_0] = f(x_0), \quad \forall k \in \mathbb{N}^*, \quad f[x_0, x_1, \dots, x_k] = \frac{f[x_0, x_1, \dots, x_{k-1}] - f[x_1, x_2, \dots, x_k]}{x_0 - x_k}.$$

$$\rightsquigarrow x_{n+1} = x_n - \frac{f(x_n)}{f[x_n, x_{n-1}]}, \quad \text{où} \quad f[x_n, x_{n-1}] = \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$$

Méthode de la sécante (2)

- Initialisation : deux points x_0 et x_1



Théorème

Soit f une fonction de classe \mathcal{C}^2 sur un intervalle $[a, b]$ de \mathbb{R} . On suppose qu'il existe $\tilde{x} \in [a, b]$ tel que $f(\tilde{x}) = 0$ et $f'(\tilde{x}) \neq 0$ (\tilde{x} est un zéro simple de f). Alors il existe $\epsilon > 0$, tel que pour tout $x_0, x_1 \in [\tilde{x} - \epsilon, \tilde{x} + \epsilon]$, la suite des itérés de la méthode de la sécante est bien définie, reste dans l'intervalle $[\tilde{x} - \epsilon, \tilde{x} + \epsilon]$ et converge vers \tilde{x} quand n tend vers l'infini. De plus, cette convergence est d'ordre $p = \frac{1+\sqrt{5}}{2} \approx 1,618$ (nombre d'or).

Exemple: calcul de la racine carrée

- Méthode de la sécante

$$x_{n+1} = x_n - \frac{x_n^2 - a}{\frac{(x_n^2 - a) - (x_{n-1}^2 - a)}{x_n - x_{n-1}}} = x_n - \frac{x_n^2 - a}{x_n + x_{n-1}}$$

- Si $a = 2$ et $x_0 = x_1 = 1$ on obtient :

$$\begin{aligned}x_0 &= 1,0000000000000000 \\x_1 &= 1,0000000000000000 \\x_2 &= 1,5000000000000000 \\x_3 &= 1,4000000000000000 \\x_4 &= 1,413793103448276 \\x_5 &= 1,414215686274510 \\x_6 &= 1,414213562057320 \\x_7 &= 1,414213562373095\end{aligned}$$

- + d'it. que Newton mais pas de calcul de dérivée

Systemes d'equations non lineaires

$$f : \begin{cases} \mathbb{R}^n & \rightarrow \mathbb{R}^n \\ x = (x_1 \dots x_n)^T & \mapsto f(x) = (f_1(x_1, \dots, x_n), \dots, f_n(x_1, \dots, x_n))^T. \end{cases}$$

On cherche donc un vecteur $x = (x_1 \dots x_n)^T \in \mathbb{R}^n$ tel que

$$f(x) = 0_{\mathbb{R}^n} \iff \begin{cases} f_1(x_1, \dots, x_n) = 0, \\ \vdots \\ f_n(x_1, \dots, x_n) = 0. \end{cases}$$

- Methode 1 vu precedemment se generalise :

$$x^{(n+1)} = x^{(n)} + M^{-1} f(x^{(n)}),$$

ou M est une certaine matrice, et nous avons les memes resultats de convergence que dans le cas d'une seule equation.

Définition

La **matrice jacobienne** d'une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ notée J_f est définie (lorsqu'elle existe) par :

$$\forall x = (x_1 \dots x_n)^T \in \mathbb{R}^n, \quad J_f(x) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1}(x) & \frac{\partial f_1}{\partial x_2}(x) & \dots & \frac{\partial f_1}{\partial x_n}(x) \\ \frac{\partial f_2}{\partial x_1}(x) & \frac{\partial f_2}{\partial x_2}(x) & \dots & \frac{\partial f_2}{\partial x_n}(x) \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1}(x) & \frac{\partial f_n}{\partial x_2}(x) & \dots & \frac{\partial f_n}{\partial x_n}(x) \end{pmatrix}$$

Méthode de Newton (1)

- Méthode de Newton se généralise : $x^{(0)} \in \mathbb{R}^n$ et

$$x^{(n+1)} = x^{(n)} - J_f(x^{(n)})^{-1} f(x^{(n)}),$$

où $J_f(x^{(n)})^{-1}$ désigne l'inverse de la matrice jacobienne de f évaluée en $x^{(n)}$.

Théorème

Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ une fonction de classe \mathcal{C}^2 sur une boule fermée B de \mathbb{R}^n . On suppose qu'il existe un zéro \tilde{x} de f dans B et que $J_f(\tilde{x})$ est inversible. Alors il existe $\epsilon > 0$ tel que pour tout $x^{(0)} \in B$ tel que $\|x^{(0)} - \tilde{x}\| \leq \epsilon$, la suite des itérés de la méthode de Newton ci-dessus est bien définie et converge vers \tilde{x} quand n tend vers l'infini.

Méthode de Newton (2)

- Calculer l'itéré $n + 1$ à partir de l'itéré n : on a besoin d'inverser la matrice $J_f(x^{(n)})$
- Pour éviter ce calcul d'inverse :

$$J_f(x^{(n)}) (x^{(n+1)} - x^{(n)}) = -f(x^{(n)}),$$

- À chaque itération, calcul de l'inverse remplacé par la résolution d'un système d'équations linéaires ce qui est asymptotiquement moins coûteux (Cf chapitre précédent)

Exemple (1)

- Considérons le système d'équations non linéaires :

$$(S) : \begin{cases} x_1^2 + 2x_1 - x_2^2 - 2 = 0, \\ x_1^3 + 3x_1x_2^2 - x_2^3 - 3 = 0. \end{cases}$$

- Notations précédentes : $n = 2$, $f_1(x_1, x_2) = x_1^2 + 2x_1 - x_2^2 - 2$, et $f_2(x_1, x_2) = x_1^3 + 3x_1x_2^2 - x_2^3 - 3$
- Matrice jacobienne de f :

$$J_f(x_1, x_2) = \begin{pmatrix} 2x_1 + 2 & -2x_2 \\ 3(x_1^2 + x_2^2) & 6x_1x_2 - 3x_2^2 \end{pmatrix}.$$

Exemple (2)

- Point de départ : $x^{(0)} = (1 \quad -1)^T$. Calculons le premier itéré de la méthode de Newton
- Formule d'itération pour $n = 1$:

$$J_f(x^{(0)}) (x^{(1)} - x^{(0)}) = -f(x^{(0)}),$$

c'est-à-dire

$$\begin{pmatrix} 4 & 2 \\ 6 & -9 \end{pmatrix} \begin{pmatrix} x_1^{(1)} - 1 \\ x_2^{(1)} + 1 \end{pmatrix} = - \begin{pmatrix} 0 \\ 2 \end{pmatrix}.$$

- En résolvant ce système linéaire, on trouve $x_1^{(1)} - 1 = -\frac{1}{12}$ et $x_2^{(1)} + 1 = \frac{1}{6} \rightsquigarrow x^{(1)} = \left(\frac{11}{12} \quad -\frac{5}{6}\right)^T$.

- La **méthode de la sécante ne se généralise pas** facilement au cas de plusieurs équations

En pratique :

- Méthode de Newton,
- Méthode $x^{(n+1)} = x^{(n)} + M^{-1} f(x^{(n)})$ en ajustant M au bout d'un certain nombre d'itérations.

En général, M assez proche de $J_f \rightsquigarrow$ convergence d'ordre ≥ 1

\rightsquigarrow Méthodes de Newton généralisées (utilisées en optimisation)

Dans ce chapitre nous nous intéressons au polynômes de la forme

$$p(z) = a_0 + a_1z + \dots + a_nz^n,$$

où $a_0, a_1, \dots, a_n \in \mathbb{C}$.

Théorème (fondamental de l'algèbre)

Tout polynôme non constant à coefficients complexes admet une racine complexe.

Donc $p(z)$ a n racines sur \mathbb{C} (avec multiplicité).

Exemple: les racines de $p(z) = z^3 - 1$ sont $\alpha_1 = 1$,

$$\alpha_2 = -\frac{1}{2} - \frac{\sqrt{3}}{2}i, \alpha_3 = -\frac{1}{2} + \frac{\sqrt{3}}{2}i.$$

Application de la méthode de Newton

- Etant donné $z_0 \in \mathbb{C}$, la méthode de Newton appliquée à $p(z)$ définit une suite (**système dynamique discret**) z_0, z_1, z_2, \dots
où

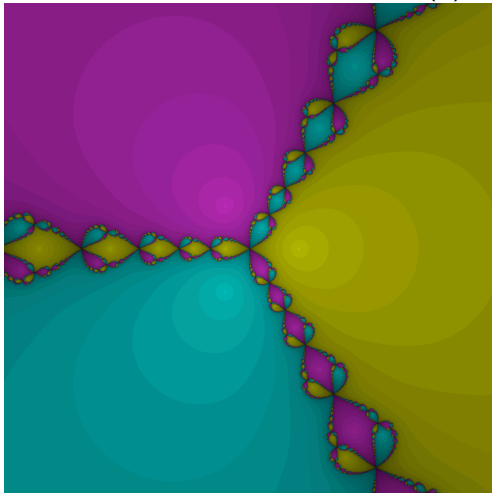
$$z_{k+1} = z_k - \frac{p(z_k)}{p'(z_k)}, \quad k \in \mathbb{N}.$$

- Quels sont les comportements possibles pour une telle suite?
- Pour chaque racine α_i de $p(z)$ on définit le **bassin d'attraction** $B_i \subset \mathbb{C}$ comme l'ensemble des points $z_0 \in \mathbb{C}$ tels que la suite de Newton définie à partir de z_0 converge vers α_i .

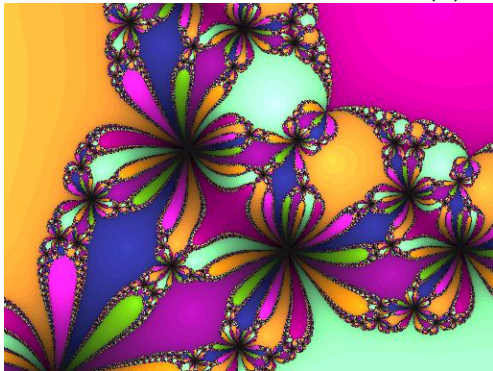
- On choisit une fenêtre d'observation (rectangle $R \subset \mathbb{C}$).
- On discrétise $R \rightsquigarrow$ nœuds z_{jk} , $j = 1, \dots, N$, $k = 1, \dots, M$.
- Pour chaque z_{jk} , on applique l'itération de Newton à partir de z_{jk} : y a-t-il convergence, et si oui, vers quelle limite ?
 - z_{jk} représenté en noir si pas de convergence,
 - z_{jk} représenté en autre couleur (suivant la limite) si convergence.

Fractales

Méthode de Newton appliquée à $p(z) = z^3 - 1$.



Méthode de Newton appliquée à $p(z) = z^8 + 15z^4 - 16$.



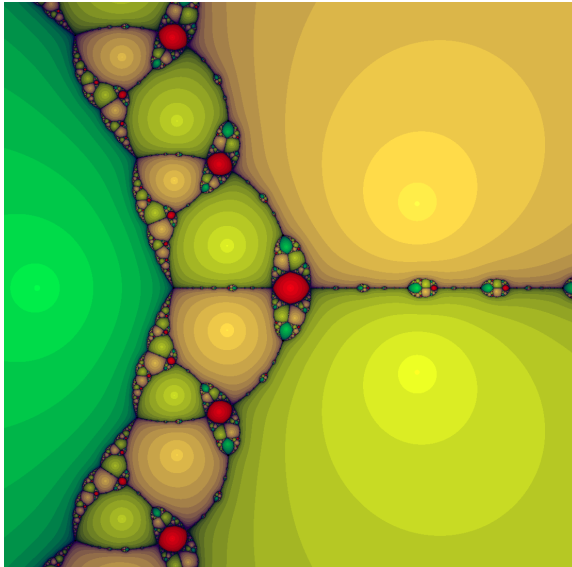
- Soit $p(z) = z^3 - 2z + 2$.
- L'itération de Newton associée est

$$z_{k+1} = z_k - \frac{z_k^3 - 2z_k + 2}{3z_k^2 - 2}.$$

- $z_k = 0 \rightsquigarrow z_{k+1} = 1 \rightsquigarrow z_{k+2} = 0 \rightsquigarrow \dots$
- $0, 1, 0, 1, \dots$ est un cycle attractif.

Cycles attractifs

Méthode de Newton appliquée à $p(z) = z^3 - 2z + 2$.



- La méthode de Halley est définie par l'itération

$$z_{k+1} = z_k - \frac{2p(x_k)p'(x_k)}{2(f'(x_k))^2 - f(x_k)f''(x_k)}.$$

- Convergence cubique au voisinage des racines.
- Newton et Halley sont exemples d'une famille plus générale de méthodes itératives pour le calcul des racines des polynômes (méthodes de König):

$$z_{k+1} = z_k + d \frac{(1/p)^{(d-1)}(z_k)}{(1/p)^{(d)}(z_k)}$$

(méthode d'ordre d).

Calcul des racines des polynômes: Newton

Théorème (Hubbard, Schleicher, Sutherland 2001)

Étant donné $d \in \mathbb{N}$, on peut déterminer un ensemble fini de points $\mathcal{S}_d \subset \mathbb{C}$ tel que, pour tout polynôme complexe $p(z)$ de degré d et pour toute racine α de $p(z)$, il existe au moins un point $z_\alpha \in \mathcal{S}_d$ tel que l'itération de Newton appliquée à partir de z_α converge à α .

En pratique, pour la méthode de Newton on a:

- convergence quadratique mais seulement si on est proche de la racine,
- comportement difficile à déterminer en dehors d'un voisinage suffisamment petit des racines.

↪ méthode généralement utilisée pour **améliorer** un calcul approché des racines.

Calcul des racines des polynômes: valeurs propres

La **matrice compagnon** ou **matrice de Frobenius** associée au polynôme

$$p(z) = z^n + a_{n-1}z^{n-1} + a_{n-2}z^{n-2} + \cdots + a_1z + a_0 = 0$$

est définie comme

$$A = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 & -a_0 \\ 1 & 0 & 0 & \cdots & 0 & -a_1 \\ 0 & 1 & 0 & \cdots & 0 & -a_2 \\ 0 & 0 & 1 & \cdots & 0 & -a_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & -a_{n-1} \end{pmatrix}$$

Théorème

Le polynôme caractéristique de A est un multiple scalaire de $p(z)$.

- Les valeurs propres de A sont racines de $p(z)$, et inversement.
- Pour calculer toutes les racines de $p(z)$ il suffit donc de calculer les valeurs propres de A .
- Des méthodes efficaces sont disponibles pour le calcul des valeurs propres
(**méthode QR**: stable, complexité $\mathcal{O}(n^3)$).
- Commande Matlab `roots`.

Exemples de polynômes mal conditionnés:

- Racines multiples: $p(z) = (z - 1)^n$
 - racine $\alpha = 1$ de multiplicité n ,
 - on introduit une perturbation: $\tilde{p}(z) = (z - 1)^n - \varepsilon$ ($\varepsilon > 0$),
 - les racines de $\tilde{p}(z)$ sont $\alpha_j = 1 + \omega_j \varepsilon^{1/n}$ (ω_j =racine n -ième de 1),
 - exemple: si $p(z) = (z - 1)^3$, le polynôme perturbé $\tilde{p}(z) = (z - 1)^3 - 10^{-6}$ a 3 racines distinctes à distance 10^{-2} par rapport à 1.
- Polynôme de Wilkinson:

$$w(z) = (z - 1) \cdot (z - 2) \cdot (z - 3) \cdot \dots \cdot (z - 20).$$