

[www.goodprepa.tech](http://www.goodprepa.tech)

# Polycopié du cours Analyse Numérique

Hassan DOUZI  
Faculté des Sciences d'Agadir

[www.goodprepa.tech](http://www.goodprepa.tech)

## Table des matières

Chapitre 0 : Introduction Générale à l'Analyse Numérique.....	3
Chapitre 1 : Compléments d'algèbre linéaire.....	8
Chapitre 2 : Méthodes directes de résolution d'un système linéaire.....	14
Chapitre 3 : Méthodes itératives de résolution d'un système linéaire.....	28
Chapitre 4 : Méthodes de résolution des équations non linéaires.....	35
Chapitre 5 : Introduction à l'optimisation.....	44
Chapitre 6 : Interpolation et Intégration numérique.....	51
Chapitre 7 : Méthodes numériques pour les équations différentielles.....	60
Chapitre 8 : Introduction aux différences finies et éléments finis.....	66

## **Chapitre 0 : Introduction Générale à l'Analyse Numérique**

### **1. Définition de l'analyse numérique**

L'Analyse Numérique comprend deux mots : l'analyse qui fait référence aux mathématiques et le mot numérique qui fait référence au traitement informatique. En d'autres termes c'est l'élaboration de méthodes de calcul mathématiques adaptées au traitement par ordinateur. En général le but de ces méthodes est la résolution de problèmes concrets qui se posent dans différentes disciplines : Physique, Economie, ...etc.



**Un exemple représentatif** de ces méthodes est la prévision météorologique : Les données collectées par les satellites et les stations d'observations donnent un aperçu sur l'état actuel du temps ; La simulation numérique permet à partir de cet état initial de prévoir le temps qui fera les jours suivants. Or cette simulation est la mise en œuvre sur ordinateur de méthodes de résolutions numériques des équations mathématiques de la mécanique de fluide.

L'analyse numérique connaît un développement phénoménal depuis la fin de la seconde guerre mondiale avec le développement de l'informatique. Aujourd'hui c'est un outil indispensable dans toutes les disciplines scientifiques sans exception.

Développer une méthode numérique revient à créer **un algorithme**. Un algorithme numérique décompose un problème en plusieurs étapes où chacune d'elle peut facilement être interprétée par l'ordinateur. Le mot Algorithme vient du nom du savant musulman **Alkhawarizmi** qui a développé entre autre des méthodes pour résoudre des équations algébriques.

## Exemple d'Algorithme :

Problème : Résolution de l'équation réelle :  $ax^2 + bx + c = 0$

Algorithme :

1. On calcule le discriminant :  $\Delta = b^2 - 4ac$
2. Si  $\Delta < 0$  il n'y a pas de solution
3. Si  $\Delta = 0$  la solution est donnée par :  $x = (-b)/(2a)$
4. Si  $\Delta > 0$  la solution est donnée par :  $x = (-b + \sqrt{\Delta})/(2a)$  ou  $x = (-b - \sqrt{\Delta})/(2a)$

Avant de trouver les méthodes numériques il faut exprimer les problèmes concrets sous forme mathématiques parmi les formes les plus courantes on peut citer :

- Equations linéaires :  $Ax = b$
- Equations non linéaires :  $f(x)=0$
- Equations différentielles :  $f(x,y,y')=0 \dots etc$

Pourquoi on a besoin de l'Analyse Numérique pour résoudre certains problèmes mathématiques ? :

1. **Solution Analytique et solution Numérique** : On aurait pu se contenter, pour résoudre des problèmes mathématiques, de chercher des solutions analytiques obtenu uniquement par l'analyse mathématiques et qui donnent des formules analytiques pour les solutions. Seulement c'est trop beau pour être vrai ; seules une infimes partie négligeables des problèmes mathématiques ont des solutions analytique par exemple :
  - Pour les équations algébriques on sait d'après la théorie d'Evariste Galois que les équations algébriques de degré  $\geq 5$  n'ont pas de solution analytique simple.
  - Pour les équations différentielles on sait par exemple résoudre :  $ay'' + by' + cy + d = 0$  mais on ne sait pas résoudre  $\sin(x)y'' + by' + cy + d = 0$

Donc souvent pour résoudre un problème mathématiques on est obligé, à défaut d'une solution analytique, de chercher une solution numérique.

2. **Solutions Analytiques inefficaces** : Dans certains cas les solutions analytiques existent mais sont complètement inefficaces à mettre en œuvre sur le plan pratique. Par exemple la résolution d'un système linéaires  $Ax=b$  peut s'effectuer par le calcul de inverse de la matrice A. Seulement le calcul de l'inverse de A devient rapidement rédhibitoire lorsque la taille de la matrice augmente car le nombre d'opérations à effectuer augmente d'une manière exponentielle. On utilise donc d'autres méthodes numériques plus rapides pour la résolution comme la méthode de Gauss.

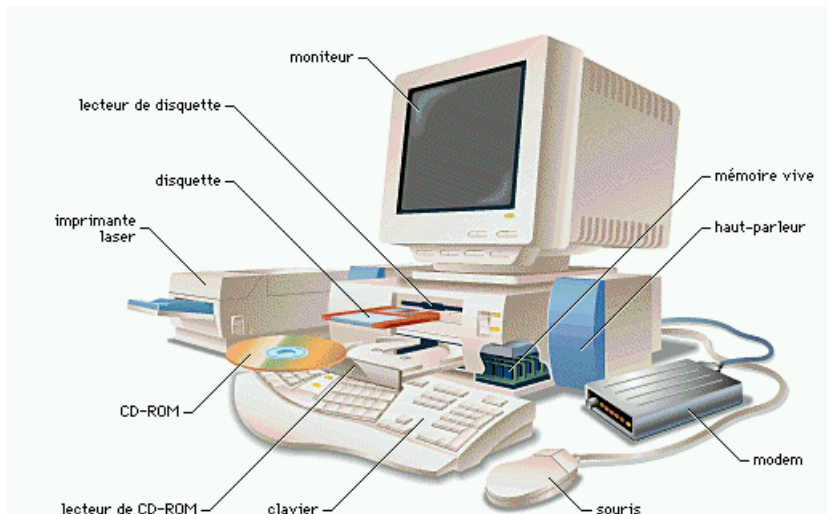
3. **Solutions Approchées** : Dans certains problèmes on ne peut pas atteindre les solutions exactes mais on peut se contenter de solutions approchées. On peut donc simplifier ces problèmes et obtenir des méthodes rapides pour calculer des solutions approchées acceptables. On peut par exemple transformer le problème non linéaire  $f(x)=0$  en une récurrence  $U(n+1)=g(U_n)$  qui converge vers une solution du problème (comme par exemple dans la méthode de Newton). On peut alors, pour  $U_0$  donné, s'arrêter après quelques itérations pour avoir une solution approchée du problème.

## 2. Représentation des nombres sur l'ordinateur

### Description simplifiée d'un ordinateur :

Un ordinateur est composé d'une unité centrale et des périphériques qui sont essentiellement des périphériques d'entrées des données (comme le clavier et la souris ...etc.) et des périphériques de sortie des résultats (comme l'écran, l'imprimante ...etc.) ; quand à l'unité centrale elle est composée essentiellement des mémoires pour stocker les données (comme disque dur, DVD, etc.) et un processeur pour les traiter.

Sur les disques mémoires l'information est stockée sur des pistes concentriques qui constituent une bande de cases mémoire. Chaque case mémoire contient une information binaire (0 ou 1) par exemple dans les Cdrom l'information est codée sous forme de bosses et des trous détecté lors de la lecture par des rayons laser.



### Représentation normalisée des nombres sur ordinateur :

Pour faire des calculs mathématiques sur ordinateur on a besoin de stocker et traiter des nombres. Chaque nombre est représenté par un certain nombre fini de cases mémoires. Une première constatation s'impose : On ne peut pas représenter tous les nombres réels ni même tous les nombres rationnels. Un ordinateur ne peut représenter qu'un nombre fini de rationnels avec un nombre fini de chiffres.

Pour faciliter le traitement tous les nombres sont représentés de la même façon c'est le principe des représentations normalisées.

## Exemple : Représentation en virgule flottante :

Les nombres à virgule flottante sont les nombres les plus souvent utilisés dans un ordinateur pour représenter des valeurs non entières. Ce sont des approximations de nombres réels.

Les nombres à virgule flottante possèdent un signe  $s$  (dans  $\{-1, 1\}$ ), une mantisse  $m$  et un exposant  $e$  (généralement 2 sur ordinateur, mais aussi 16 sur certaines anciennes machines, 10 sur de nombreuses calculatrices, ou éventuellement toute autre valeur). En faisant varier  $e$ , on fait « flotter » la virgule décimale. Généralement,  $m$  est d'une taille fixée.

Ceci s'oppose à la représentation dite en virgule fixe, où l'exposant  $e$  est fixé.

$$x = \pm 2^n * 0, C_1 C_2 \dots C_m$$

- $0, C_1 C_2 \dots C_m$  : est appelé la mantisse.
- $-M_1 < n < M_1$  est appelé l'exposant la largeur de l'intervalle de  $n$  dépend de la puissance le l'ordinateur.
- $m$  est le nombre de chiffre de la représentation et sa valeur dépend de la puissance de l'ordinateur.

## Propagation des erreurs numériques

Les calculs en virgule flottante sont pratiques, mais présentent divers désagréments, notamment :

- leur précision limitée, qui se traduit par des arrondis (dus aux opérations, ainsi qu'aux changements de base implicites, si la base est différente de 10) qui peuvent s'accumuler de façon gênante. Pour cette raison, les travaux de comptabilité ne sont pas effectués en virgule flottante, car tout doit tomber juste au centième près. En particulier, la soustraction de deux nombres très proches provoque une grande perte de précision relative : on parle de « cancellation ».
- une plage d'exposants limitée, pouvant donner lieux à des « overflows » (lorsque le résultat d'une opération est plus grand que la plus grande valeur représentable) et à des « underflows » (lorsqu'un résultat est plus petit, en valeur absolue, que le plus petit flottant normalisé positif), puis à des résultats n'ayant plus aucun sens.

**Exemple :** le calcul de l'exponentielle avec la série :

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!} + O(x^n)$$

x	Exp(x)	Somme, n=14
-10	$4.54 \cdot 10^{-5}$	$4.54 \cdot 10^{-5}$
-15	$3.06 \cdot 10^{-7}$	$3.05 \cdot 10^{-7}$
-20	$2.06 \cdot 10^{-9}$	$-1.55 \cdot 10^{-7}$
-25	$1.39 \cdot 10^{-11}$	$1.87 \cdot 10^{-5}$
-30	$9.36 \cdot 10^{-14}$	$6.25 \cdot 10^{-4}$

Lorsque  $x$  est négative ( $< -20$ ) et si on utilise la série ci-dessus pour calculer l'exponentielle de  $x$ , on obtient des résultats très éloignés des valeurs exactes à cause de l'accumulation des erreurs d'arrondis dans les additions et multiplications des termes successifs de la série. Donc ce n'est pas une bonne méthode numérique pour calculer la fonction exponentielle. En Analyse numérique on dit que c'est une méthode instable.

Une méthode numérique est dite **stable** si elle donne de bons résultats quelque soit la nature de ses données.

Les données initiales qui vont être traitées par ordinateur sont souvent tronquées. Il faut donc que les méthodes numériques utilisées soient insensibles aux petites variations des données initiales on dit dans ce cas que la méthode numérique est **bien conditionnée**. Une méthode est mal conditionnée si de petites variations sur les données peut produire de grandes perturbations sur les résultats obtenus.

Il faut donc prendre en compte tous ces paramètres au cours de l'élaboration des méthodes numériques. En Analyse Numérique, pour résoudre un problème donné, souvent le problème qui se pose n'est pas de trouver une méthode numérique mais la difficulté réside dans la démonstration que le problème est bien conditionné et la méthode utilisée est stable.

# Chapitre 1 : Compléments d'algèbre linéaire

## • Introduction

Une matrice d'ordre  $(n,k)$  est un tableau de  $n$  lignes et  $k$  colonnes:

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1j} & \cdots & a_{1k} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{i1} & \cdots & a_{ij} & \cdots & a_{ik} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nj} & \cdots & a_{nk} \end{pmatrix}$$

Lorsque  $n=k$  on dit que la matrice est carré d'ordre  $n$

## Matrices particulières:

- Matrices diagonales et triangulaires
- Matrice adjointe et transposée :  $A^* = (\overline{a_{ji}})$   $A^t = (a_{ji})$
- Matrices hermitiennes et symétriques:  $A^* = A$  ;  $A^t = A$  ;
- Matrices Unitaire et orthogonales:  $A^{-1} = A^*$  ;  $A^{-1} = A^T$
- Matrices à diagonale dominante:

$$\forall i \in 1, n \quad \sum_{j=1, j \neq i}^n |a_{ij}| \leq (<) a_{ii}$$

- Matrices semblables :  $A$  et  $B$  sont semblables s'il existe une matrice inversible  $P$  tel que :  $B = P^{-1}AP$

On note  $\mathcal{M}_N(\mathbb{R})$  l'ensemble des matrices carrées d'ordre  $N$ . Soit  $A \in \mathcal{M}_N(\mathbb{R})$  une matrice inversible, et  $b \in \mathbb{R}^N$ , on a comme objectif de résoudre le système linéaire  $Ax = b$ , c'est à dire de trouver  $x$  solution de :

$$\begin{cases} x \in \mathbb{R}^N \\ Ax = b \end{cases} \quad (1.1.1)$$

Comme  $A$  est inversible, il existe un unique vecteur  $x \in \mathbb{R}^N$  solution de (1.1.1).

## • Rappel sur les normes matricielles

### Définition :

Une norme vectorielle sur  $\mathbb{R}^n$  est une application de l'ensemble  $\mathbb{R}^n$  vers  $\mathbb{R}$  qui vérifie un certain nombre de propriétés :

- $\|v\|=0 \Leftrightarrow v=0$
- $\|\lambda v\|=|\lambda| \|v\|$
- $\|v+w\| \leq \|v\| + \|w\|$



## Exemples

Quelques normes vectorielles très utilisées : Soit  $v = (v_i)_{1 \leq i \leq n}$  alors on a :

- $\|v\|_1 = \sum_{i=1}^n |v_i|$
- $\|v\|_\infty = \max_{1 \leq i \leq n} |v_i|$

## Définition

Une norme matricielle est une application de l'ensemble des matrices carré d'ordre  $n$  vers  $\mathbb{R}$  qui vérifie un certain nombre de propriétés :

- $\|A\|=0 \Leftrightarrow A=0$
- $\|\lambda A\|=|\lambda| \|A\|$
- $\|A+B\| \leq \|A\| + \|B\|$
- $\|AB\| \leq \|A\| \|B\|$

## Exemples

Quelques normes matricielles très utilisées : Soit  $A = (a_{ij})_{1 \leq i, j \leq n}$  alors on a :

- $\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|$
- $\|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}|$  ■

## Définition

On considère  $\mathbb{R}^N$  muni d'une norme  $\|\cdot\|$ . On appelle norme matricielle induite (ou norme induite) sur  $\mathcal{M}_N(\mathbb{R})$  par la norme  $\|\cdot\|$ , encore notée  $\|\cdot\|$ , la norme sur  $\mathcal{M}_N(\mathbb{R})$  définie par :  $\|A\| = \sup\{\|Ax\|; x \in \mathbb{R}^N, \|x\| = 1\}$  pour toute matrice  $A \in \mathcal{M}_N(\mathbb{R})$ .

**Proposition** Soit  $\mathcal{M}_N(\mathbb{R})$  muni d'une norme induite  $\|\cdot\|$ . Alors pour toute matrice  $A \in \mathcal{M}_N(\mathbb{R})$ , on a :

1.  $\|Ax\| \leq \|A\| \|x\|, \forall x \in \mathbb{R}^N,$
2.  $\|A\| = \max\{\|Ax\| ; \|x\| = 1, x \in \mathbb{R}^N\},$
3.  $\|A\| = \max\left\{\frac{\|Ax\|}{\|x\|} ; x \in \mathbb{R}^N \setminus \{0\}\right\}.$
4.  $\|\cdot\|$  est une norme matricielle.

## • Valeurs propres, Rayon spectral et Polynôme caractéristique

**Définition** (Valeurs propres et rayon spectral) Soit  $A \in \mathcal{M}_N(\mathbb{R})$  une matrice inversible. On appelle valeur propre de  $A$  tout  $\lambda \in \mathbb{C}$  tel qu'il existe  $x \in \mathbb{C}^N$ ,  $x \neq 0$  tel que  $Ax = \lambda x$ . L'élément  $x$  est appelé vecteur propre de  $A$  associé à  $\lambda$ . On appelle rayon spectral de  $A$  la quantité  $\rho(A) = \max\{|\lambda|; \lambda \in \mathbb{C}, \lambda \text{ valeur propre de } A\}$ .

**Proposition** Soit  $A \in \mathcal{M}_N(\mathbb{R})$  une matrice carrée quelconque, et  $\|\cdot\|$  une norme matricielle (induite ou non). Alors

$$\rho(A) \leq \|A\|.$$

**Proposition** (Rayon spectral et norme induite)

Soient  $A \in \mathcal{M}_N(\mathbb{R})$  et  $\varepsilon > 0$ . Il existe une norme sur  $\mathbb{R}^N$  (qui dépend de  $A$  et  $\varepsilon$ ) telle que la norme induite sur  $\mathcal{M}_N(\mathbb{R})$ , notée  $\|\cdot\|_{A,\varepsilon}$ , vérifie  $\|A\|_{A,\varepsilon} \leq \rho(A) + \varepsilon$ .

**Proposition** Soit  $A$  une matrice carrée sur de  $n$  éléments de  $\mathbb{C}$ , alors les propriétés suivantes sont équivalentes :

1. un scalaire  $\lambda$  est une valeur propre de  $A$
2. la matrice  $M = A - \lambda I$  est singulière
3. le scalaire  $\lambda$  est une racine du polynôme de degré  $n$  appelé polynôme caractéristique de  $A$  :

$$P_A(\lambda) = \det(A - \lambda I)$$

**Remarque** On peut donc déduire grâce aux propriétés des polynômes et leurs racines, que le polynôme caractéristique de  $A$  s'écrit :

$$P_A(\lambda) = (\lambda - \lambda_1)^{k_1} (\lambda - \lambda_2)^{k_2} \dots (\lambda - \lambda_p)^{k_p}$$

où les  $\lambda_i$  sont les  $p$  racines distinctes de  $P_A(\lambda)$ ,  $k_i$  est la multiplicité de la racine  $\lambda_i$ .

**Remarque** Si on se restreint à des matrices d'éléments de  $\mathbb{R}$  et des valeurs propres dans  $\mathbb{R}$ , certaines matrices peuvent ne pas avoir de valeurs propres et de vecteurs propres.

## • Matrices diagonalisables

### Définition

Soit  $A$  une matrice réelle carrée d'ordre  $n$ .

On dit que  $A$  est diagonalisable dans  $\mathbb{R}$  si il existe une base  $(\phi_1, \dots, \phi_n)$

et des réels  $\lambda_1, \dots, \lambda_n$  (pas forcément distincts) tels que  $A\phi_i = \lambda_i\phi_i$  pour  $i = 1, \dots, n$ .

### Lemme

Soit  $A$  une matrice réelle carrée d'ordre  $n$ , diagonalisable dans  $\mathbb{R}$ .

Alors  $A = P^{-1} \text{diag}(\lambda_1, \dots, \lambda_n) P$ , où les vecteurs colonnes de la matrice  $P$  égaux aux vecteurs  $\phi_1, \dots, \phi_n$ .

On a donc

$$P^{-1}AP = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix} = D.$$

### Remarques :

- Une matrice est diagonalisable si elle est semblable à une matrice diagonale
- Pour toute matrice carrée et symétrique  $A$  il existe une matrice orthogonale  $O$  telle que  $O^{-1}AO$  soit diagonale

## • Conditionnement d'un système linéaire

Soit  $Ax=b$  un système linéaire, une matrice est mal conditionnée lorsque de petites variations sur les données  $A$  ou  $b$  entraînent de très fortes variations sur le résultat  $x$ , (et même si  $x$  est obtenu sans erreurs d'arrondi ni de troncatures). Le conditionnement de la matrice d'un système linéaire (qu'on note par  $\text{Cond}(A)$ ) est un outil qui permet de mesurer l'instabilité numérique de ce système.

### Définition :

Soit  $A$  une matrice carrée inversible d'ordre  $n$ , alors le conditionnement de  $A$  est donné par:

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

Où  $\|\cdot\|$  représente une norme matricielle choisie.

Si le conditionnement est très élevé par rapport à 1 ( $\text{Cond}(A) \gg 1$ ) alors le système est mal conditionné. Le conditionnement est une valeur qui est toujours supérieure ou égale à 1 et le système est idéal lorsque on a l'égalité avec 1 ( $\text{Cond}(A)=1$ ). C'est le cas par exemple des matrices orthogonales.

### Théorème (Majoration des perturbations) :

Soit  $Ax = b$  un système linéaire, on note par  $\Delta A$  et  $\Delta b$  une faible perturbation respectivement sur  $A$  et  $b$ . Et soit  $\Delta x$  la perturbation sur  $x$  obtenue suivant qu'on modifie  $A$  ou  $b$  (où  $x$  est la solution du système  $Ax=b$ ).

(i) Cas d'une perturbation du second membre:

$$\text{Si } A(x+\Delta x) = (b+\Delta b) \text{ alors on a : } \frac{\|\Delta x\|}{\|x\|} \leq \text{Cond}(A) \frac{\|\Delta b\|}{\|b\|}$$

(ii) Cas d'une perturbation de la matrice:

$$\text{Si } (A+\Delta A)(x+\Delta x) = b \text{ alors on a : } \frac{\|\Delta x\|}{\|x+\Delta x\|} \leq \text{Cond}(A) \frac{\|\Delta A\|}{\|A\|}$$

### Exemple :

On considère le système linéaire  $Ax=b$  avec :

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

$$\text{La solution de ce système est donnée par } x = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

$$\text{Considérons une perturbation du second membre : } b + \Delta b = \begin{pmatrix} 32.1 \\ 22.9 \\ 33.1 \\ 30.9 \end{pmatrix}$$

$$\text{La solution du système : } A(x + \Delta x) = (b + \Delta b) \text{ est donnée par } x + \Delta x = \begin{pmatrix} 9.2 \\ -12.6 \\ 4.5 \\ -1.1 \end{pmatrix}$$

$$\text{Dans ce cas on a : } \frac{\|\Delta b\|_{\infty}}{\|b\|_{\infty}} = \frac{0.1}{33} \approx 0,003 \quad \text{et} \quad \frac{\|\Delta x\|_{\infty}}{\|x\|_{\infty}} = \frac{13.6}{1} = 13.6 \quad \text{Ainsi}$$

$$\frac{\|\Delta x\|_{\infty}}{\|x\|_{\infty}} = 4488 \frac{\|\Delta b\|_{\infty}}{\|b\|_{\infty}} \text{ donc le conditionnement de ce système est supérieur à 4488 et le}$$

système est, bien sur, mal conditionné.

Considérons maintenant une perturbation de la matrice  $A$  donné par :

$$A = \begin{pmatrix} 10 & 7 & 8.1 & 7.2 \\ 7.08 & 5.08 & 6 & 5 \\ 8 & 5.98 & 9.89 & 9 \\ 6.99 & 4.99 & 9 & 9.89 \end{pmatrix}$$

La solution du système :  $(A + \Delta A)(x + \Delta x) = b$  est donnée par  $x + \Delta x = \begin{pmatrix} -81 \\ 137 \\ -34 \\ 22 \end{pmatrix}$

Là encore de petites variations sur les données modifient énormément le résultat.

## Chapitre 2 : Méthodes directes de résolution d'un système linéaire

### 1. Introduction

On se propose de résoudre le système linéaire suivant :

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n} = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n} = b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_m \end{cases}$$

Où  $x_1, \dots, x_n$  sont les inconnues et les nombres  $a_{ij}$  sont les coefficients du système.

Un système d'équations linéaires peut aussi s'écrire sous la forme :  $Ax = b$

avec :

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}; \quad x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad \text{et} \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}$$

Dans la suite on va toujours considérer que  $n=m$  c'est-à-dire qu'il y a autant d'équations que d'inconnus ou aussi que la matrice est carrée.

La première méthode qui vient à l'esprit pour résoudre ce système est l'utilisation des formules de Cramer :

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

où  $A_i$  est la matrice  $A$  avec le remplacement de la  $i$ ème colonne par le vecteur  $b$ .

Ces formules nécessitent le calcul de plusieurs déterminant or le nombre d'opérations nécessaires pour le calcul d'un déterminant pour un système d'ordre est

de l'ordre de  $n!$ . Ce nombre devient rapidement impossible à réaliser lorsque  $n$  devient grand. ( $n!$  augmente plus vite que l'exponentielle).

Les méthodes numériques de résolutions d'un système linéaire sont de deux types :

- Les méthodes directes : qui conduisent à la solution en un nombre fini d'opérations élémentaires comme par exemple la méthode de Gauss (ou encore LU ou Choleski)
- Les méthodes itératives qui génèrent une suite qui converge vers la solution du système linéaire comme par exemple la méthode de House-Holder.

Le choix de la méthode utilisée dépend des propriétés caractéristiques de la matrice du système considéré. Par exemple on distingue particulièrement les matrices « creuses » qui contiennent « beaucoup » de zéros des matrices pleines qui n'en contiennent que « peu » : Les matrices diagonales et les matrices triangulaires sont des exemples particuliers de matrices creuses.

- une matrice **diagonale** ( $A = D$ ) si et seulement si :  $a_{i,j} = 0, \quad \forall i \neq j$
- une matrice **triangulaire supérieure** ( $A = U$ ) si et seulement si  $a_{i,j} = 0, \quad \forall i < j$
- une matrice **triangulaire inférieure** ( $A = L$ ) si et seulement si  $a_{i,j} = 0, \quad \forall i > j$

## 2. Méthode pour les matrices triangulaires

Le principe des méthodes directes, que nous allons étudier repose sur la remarque intéressante suivante :

Si la matrice  $A$  du système linéaire est triangulaire supérieure (ou triangulaire inférieure) alors la résolution numérique peut s'effectuer très rapidement :

$$\left\{ \begin{array}{l} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n-1}x_{n-1} + a_{1n}x_n = b_1 \\ a_{22}x_2 + \dots + a_{2n-1}x_{n-1} + a_{2n}x_n = b_2 \\ \vdots \\ a_{n-1,n-1}x_{n-1} + a_{n-1n}x_n = b_{n-1} \\ a_{nn}x_n = b_n \end{array} \right.$$

$$\left\{ \begin{array}{l} x_n = a_{nn}^{-1}b_n \\ x_{n-1} = a_{n-1,n-1}^{-1}(b_{n-1} - a_{n-1n}x_n) \\ \vdots \\ x_2 = a_{22}^{-1}(b_2 - \dots - a_{2n-1}x_{n-1} - a_{2n}x_n) \\ x_1 = a_{11}^{-1}(b_1 - a_{12}x_2 - \dots - a_{1n-1}x_{n-1} - a_{1n}x_n) \end{array} \right.$$

**Algorithme :**

**Fonction  $x = \text{triang}(A,b)$**

```

$$x_1 \leftarrow \frac{b_1}{a_{11}}$$


pour  $i = 2$  jusqu'à  $n$



$\text{somme} \leftarrow b_i$



pour  $j = 1$  jusqu'à  $i - 1$



$\text{somme} \leftarrow \text{somme} - a_{ij}x_j$



fait



$x_i \leftarrow \frac{\text{somme}}{a_{ii}}$



fait


```

Cette méthode pour les matrices triangulaires supérieures s'appelle : méthode de remontée elle nécessite :  $n(n-1)/2$  additions,  $n(n-1)/2$  multiplications et  $n$  division on dit que la complexité de l'algorithme de la méthode de remontée est  $O(n^2)$ .

De la même manière on a une méthode de descente pour les matrices triangulaires inférieures .

### **3. Méthode d'élimination de Gauss**

Dans les méthodes directes pour résoudre le système linéaire  $Ax=b$  on cherche à trouver une matrice  $M$  inversible telle que le produit  $MA$  est une matrice triangulaire supérieure. Il suffit alors de résoudre le système triangulaire  $(MA)x=Mb$  ,qui est équivalent à  $Ax=b$ , par la méthode de remontée .

On vient de voir qu'il est très simple de résoudre des systèmes triangulaires. La transformation du système original en un système triangulaire se fait en choisissant des combinaisons linéaires appropriées des équations du système.

#### **Exemple**

Soit le système : 
$$\begin{cases} 3x_1 + 5x_2 = 9 \\ 6x_1 + 7x_2 = 4 \end{cases}$$

Si l'on multiplie la première équation par 2 et que l'on la soustrait de la deuxième on obtient :

$$\begin{cases} 3x_1 + 5x_2 = 9 \\ -3x_2 = -14 \end{cases}$$

On obtient un système triangulaire et équivalent au système initial. Cette procédure s'appelle élimination de Gauss. Il reste ainsi à résoudre le système triangulaire par la méthode de remontée pour trouver la solution du système initial.



$$\begin{cases} x_1 = \frac{13}{9} \\ x_2 = \frac{14}{3} \end{cases}$$

Sous forme matricielle on peut effectuer cette même opération en représentant la matrice du système original comme le produit de deux matrices triangulaires :

$$\underbrace{\begin{bmatrix} 3 & 5 \\ 6 & 7 \end{bmatrix}}_A = \underbrace{\begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix}}_L \underbrace{\begin{bmatrix} 3 & 5 \\ 0 & -3 \end{bmatrix}}_U.$$

On appelle cette opération la décomposition LU de la matrice A.

La solution du système original  $Ax = b$  est alors obtenue en résolvant deux systèmes triangulaires successivement en remplaçant A par sa factorisation :

$$Ax = L \underbrace{Ux}_y = Ly = b$$

On cherche la solution y à partir du système triangulaire inférieur  $Ly = b$ , puis la solution de x à partir du système triangulaire supérieur  $Ux = y$ .

## Formalisation de l'élimination de Gauss :

Il s'agit de formaliser une procédure qui transforme une matrice en une matrice triangulaire supérieure en éliminant, colonne après colonne, les éléments non nuls en dessous de la diagonale comme illustré dans l'exemple suivant :

Soit le système linéaire associé à la matrice suivante :

$$\begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 10 \end{bmatrix}$$

On veut la transformer en une matrice triangulaire supérieure. On obtient cette forme en deux étapes. Lors de la première étape on soustrait 2 fois la première ligne de la deuxième ligne et 3 fois la première ligne de la troisième ligne pour obtenir la matrice :

$$\begin{bmatrix} 1 & 4 & 7 \\ 0 & -3 & -6 \\ 0 & -6 & -11 \end{bmatrix}$$

Lors de la deuxième étape on transforme cette nouvelle matrice en soustrayant 2 fois la deuxième ligne de la troisième ligne pour obtenir la forme triangulaire recherchée :

$$\begin{bmatrix} 1 & 4 & 7 \\ 0 & -3 & -6 \\ 0 & 0 & 1 \end{bmatrix}$$

De manière générale, étant donné un vecteur x avec  $x_k \neq 0$ , on peut annuler tout les éléments  $x_i$ ,  $i > k$ , en multipliant le vecteur x par la matrice  $M_k$  suivante :

$$M_k x = \begin{bmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & -\tau_{k+1}^{(k)} & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & -\tau_n^{(k)} & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_k \\ x_{k+1} \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} x_1 \\ \vdots \\ x_k \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

où :  $\tau_i^{(k)} = x_i/x_k$ ,  $i = k + 1, \dots, n$ .

Le diviseur  $x_k$  est appelé le pivot, la matrice  $M_k$  est appelée la matrice de transformation de Gauss et les  $\tau^{(k)}_i$  constituent les multiplicateurs de Gauss.

En appliquant  $(n - 1)$  fois dans l'ordre  $M_1, \dots, M_{n-1}$  la transformation de Gauss à une matrice  $A$  d'ordre  $n$  :

$$M_{n-1} \cdots M_2 M_1 A = U$$

On obtient une matrice  $U$  qui est triangulaire supérieure. Ce procédé est appelé élimination de Gauss.

### Algorithme

**Fonction  $A, b = \text{descent}(A, b)$**

```
pour  $k = 1$  jusqu'à  $n - 1$ 
     $\text{pivot} \leftarrow a_{kk}$  (* stratégie de pivot *)
    si  $\text{pivot} \neq 0$  alors
        pour  $i = k + 1$  jusqu'à  $n$ 
             $b_i \leftarrow b_i - \frac{a_{ik}}{\text{pivot}} b_k$ 
            pour  $j = k + 1$  jusqu'à  $n$ 
                 $a_{ij} \leftarrow a_{ij} - \frac{a_{ik}}{\text{pivot}} a_{kj}$ 
            fait
        fait
    sinon "problème"
fait
```

**Fonction  $x = \text{Gauss}(A, b)$**

$U, c = \text{descent}(A, b)$

$x = \text{triang}(U, c)$

## Exemple

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 10 \end{bmatrix}, \quad M_1 = \begin{bmatrix} 1 & 0 & 0 \\ -4 & 1 & 0 \\ -7 & 0 & 1 \end{bmatrix}, \quad \underbrace{M_1 A}_{A^{(1)}} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & -6 & -11 \end{bmatrix},$$

$$M_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -2 & 1 \end{bmatrix}, \quad \underbrace{M_2 M_1 A}_{A^{(2)}} = M_2 A^{(1)} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & 0 & 1 \end{bmatrix}.$$

## 4. Décomposition LU

Propriétés des matrices  $M_k$  :

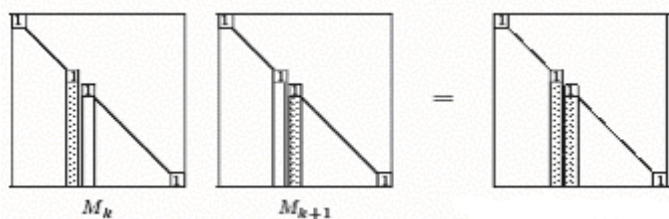
- D'abord se sont des matrices triangulaires inférieures avec diagonale unitaire donc elles sont inversibles :

$$\begin{bmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & -\tau_{k+1}^{(k)} & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & -\tau_n^{(k)} & 0 & \cdots & 1 \end{bmatrix}$$

- Ensuite le calcul de l'inverse est très facile, c'est tout simplement un changement de signe sous la diagonale de la k-ième colonne :

$$\begin{bmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & \tau_{k+1}^{(k)} & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \tau_n^{(k)} & 0 & \cdots & 1 \end{bmatrix}$$

- Enfin le produit des deux matrices  $M_k M_{k+1}$  est aussi très facile à calculer avec des 1 sur la diagonale plus les colonnes k de  $M_k$  et (k+1) de  $M_{k+1}$  :



Ainsi on obtient le procédé pour calculer la décomposition LU d'une matrice  $A$  inversible :

A partir de l'élimination de Gauss

$$M_{n-1} \cdots M_1 A = U$$

qui conduit à une matrice triangulaire supérieure  $U$  on déduit, en écrivant

$$\underbrace{(M_{n-1} \cdots M_1)^{-1} (M_{n-1} \cdots M_1)}_I A = \underbrace{(M_{n-1} \cdots M_1)^{-1}}_L U$$

que la matrice  $L$  dans la factorisation  $A = LU$  est constitué par le produit

$$L = (M_{n-1} \cdots M_1)^{-1} = M_1^{-1} \cdots M_{n-1}^{-1}.$$

**Algorithme :**

Fonction  $L, U = \text{décompose}(A)$

pour  $k = 1$  jusqu'à  $n - 1$

$pivot \leftarrow a_{kk}$  (\* stratégie de pivot \*)

si  $pivot \neq 0$  alors

$\ell_{kk} \leftarrow 1$

pour  $i = k + 1$  jusqu'à  $n$

$\ell_{ik} \leftarrow \frac{a_{ik}}{pivot}$

pour  $j = k + 1$  jusqu'à  $n$

$a_{ij} \leftarrow a_{ij} - \ell_{ik} a_{kj}$

fait

fait

fait sinon "problème"

Fonction  $x = \text{LU}(A, b)$

$L, U = \text{decompose}(A)$

$y = \text{triang}(L, b)$

$x = \text{triang}(U, y)$

**Existence de la factorisation LU**

Il apparaît qu'à chaque étape de l'élimination de Gauss il faut avoir un pivot non nul pour pouvoir déduire à la fin la décomposition LU. Donc la factorisation LU n'existe pas toujours. Le théorème qui suit, donne une condition nécessaire pour assurer l'existence de la décomposition :

**Théorème :**

Soit A une matrice d'ordre n. les sous matrices diagonales  $\Delta_k$  de A sont définies par :

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix}, \Delta_k = \begin{pmatrix} a_{11} & \cdots & a_{1k} \\ \vdots & \ddots & \vdots \\ a_{k1} & \cdots & a_{kk} \end{pmatrix} \quad k = 1 \cdots n$$

Si la matrice A et ses sous matrices diagonales sont inversibles alors A admet une factorisation LU unique telle que  $A=LU$ ■

**Techniques de pivot** Dans la présentation de la méthode de Gauss et de la décomposition LU, on a supposé que la condition  $a_{i,i}^{(i)} \neq 0$  était vérifiée à chaque étape. Or il peut s'avérer que ce ne soit pas le cas, ou que, même si la condition est vérifiée, le "pivot"  $a_{i,i}^{(i)}$  soit très petit, ce qui peut entraîner des erreurs d'arrondi importantes dans les calculs. On peut résoudre ce problème en utilisant les techniques de "pivot partiel" ou "pivot total", qui reviennent à choisir une matrice de permutation P qui n'est pas forcément égale à la matrice identité dans le théorème 1.2.

Plaçons-nous à l'itération i de la méthode de Gauss. Comme la matrice  $A^{(i)}$  est forcément non singulière, on a :

$$\det(A^{(i)}) = a_{1,1}^{(i)} a_{2,2}^{(i)} \cdots a_{i-1,i-1}^{(i)} \det \begin{pmatrix} a_{i,i}^{(i)} & \cdots & a_{i,N}^{(i)} \\ \vdots & \ddots & \vdots \\ a_{N,i}^{(i)} & \cdots & a_{N,N}^{(i)} \end{pmatrix} \neq 0.$$

On a donc en particulier

$$\det \begin{pmatrix} a_{i,i}^{(i)} & \cdots & a_{i,N}^{(i)} \\ \vdots & \ddots & \vdots \\ a_{N,i}^{(i)} & \cdots & a_{N,N}^{(i)} \end{pmatrix} \neq 0.$$

**Pivot partiel** On déduit qu'il existe  $i_0 \in \{i, \dots, N\}$  tel que  $a_{i_0,i}^{(i)} \neq 0$ . On choisit alors  $i_0 \in \{i, \dots, N\}$  tel que  $|a_{i_0,i}^{(i)}| = \max\{|a_{k,i}^{(i)}|, k = i, \dots, N\}$ . On échange alors les lignes i et  $i_0$  (dans la matrice A et le second membre b) et on continue la procédure de Gauss décrite plus haut.

**Pivot total** On choisit maintenant  $i_0$  et  $j_0 \in \{i, \dots, N\}$  tels que  $|a_{i_0, j_0}^{(i)}| = \max\{|a_{k,j}^{(i)}|, k = i, \dots, N, j = i, \dots, N\}$ , et on échange alors les lignes  $i$  et  $i_0$  (dans la matrice  $A$  et le second membre  $b$ ), les colonnes  $j$  et  $j_0$  de  $A$  et les inconnues  $x_j$  et  $x_{j_0}$ .

L'intérêt de ces stratégies de pivot est qu'on aboutit toujours à la résolution du système (dès que  $A$  est inversible). La stratégie du pivot total permet une moins grande sensibilité aux erreurs d'arrondi. L'inconvénient majeur est qu'on change la structure de  $A$  : si, par exemple la matrice avait tous ses termes non nuls sur quelques diagonales seulement, ceci n'est plus vrai pour la matrice  $A^{(N)}$ .

## 5. Factorisation de Choleski

### Exemple

La matrice symétrique  $A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 5 & 5 & 5 \\ 1 & 5 & 14 & 14 \\ 1 & 5 & 14 & 15 \end{pmatrix}$  est égale au produit à droite de la matrice triangulaire  $L : \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ 1 & 2 & 3 & 0 \\ 1 & 2 & 3 & 1 \end{pmatrix}$  et de sa transposée  $L^T$ .

**Théorème** (Factorisation de Choleski d'une matrice) :

Si  $A$  est une matrice symétrique définie positive, il existe au moins une matrice réelle triangulaire inférieure  $L$  telle que :  $A=LL^T$

On peut également imposer que les éléments diagonaux de la matrice  $L$  soient tous positifs, et la factorisation correspondante est alors unique■

### Algorithme

On cherche la matrice :

$$L = \begin{bmatrix} l_{11} & & & \\ l_{21} & l_{22} & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix}$$

De l'égalité  $A=LL^T$  on déduit :

$$a_{ij} = (LL^T)_{ij} = \sum_{k=1}^n l_{ik}l_{jk} = \sum_{k=1}^{\min\{i,j\}} l_{ik}l_{jk}, \quad 1 \leq i, j \leq n$$

puisque  $l_{pq}=0$  si  $1 \leq p < q \leq n$ .

La matrice  $A$  étant symétrique, il suffit que les relations ci-dessus soient vérifiées pour  $i \leq j$ , c'est-à-dire que les éléments  $l_{ij}$  de la matrice  $L$  doivent satisfaire :

$$a_{ij} = \sum_{k=1}^i l_{ik} l_{jk}, \quad 1 \leq i, j \leq n$$

Pour  $j=1$ , on détermine la première colonne de  $L$  :

$$(i=1) \quad a_{11} = l_{11} l_{11} \text{ d'où } l_{11} = \sqrt{a_{11}}$$

$$(i=2) \quad a_{12} = l_{11} l_{21} \text{ d'où } l_{21} = \frac{a_{12}}{l_{11}}$$

...

$$(i=n) \quad a_{1n} = l_{11} l_{n1} \text{ d'où } l_{n1} = \frac{a_{1n}}{l_{11}}$$

On détermine la  $j$ ème colonne de  $L$ , après avoir calculé les  $(j-1)$  premières colonnes :

$$(i=j) \quad a_{ii} = l_{i1} l_{i1} + \dots + l_{ii} l_{ii} \text{ d'où } l_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} l_{ik}^2}$$

$$(i=j+1) \quad a_{i,i+1} = l_{i1} l_{i+1,1} + \dots + l_{ii} l_{i+1,i} \text{ d'où } l_{i+1,i} = \frac{a_{i,i+1} - \sum_{k=1}^{i-1} l_{ik} l_{i+1,k}}{l_{ii}}$$

...

$$(i=n) \quad a_{i,n} = l_{i1} l_{n1} + \dots + l_{ii} l_{ni} \text{ d'où } l_{ni} = \frac{a_{i,n} - \sum_{k=1}^{i-1} l_{ik} l_{nk}}{l_{ii}}$$

Il résulte du théorème précédent qu'il est possible de choisir tous les éléments  $l_{ij} > 0$  en assurant que toutes les quantités

$$a_{11}, \dots, a_{ii} - \sum_{k=1}^{i-1} l_{ik}^2, \dots$$

sont positives.

## Fonction $L = \text{Choleski}(A)$

```

 $\ell_{11} \leftarrow \sqrt{a_{11}}$ 
pour  $j = 2$  jusqu'à  $n$ 
     $\ell_{j1} \leftarrow \frac{a_{j1}}{\ell_{11}}$ 
fait
    pour  $i = 2$  jusqu'à  $n-1$ 
        somme  $\leftarrow 0$ 
        pour  $k = 1$  jusqu'à  $i-1$ 
            somme  $\leftarrow$  somme +  $(\ell_{ik})^2$ 
        fait
         $\ell_{ii} \leftarrow \sqrt{a_{ii} - \text{somme}}$ 
        pour  $j = i+1$  jusqu'à  $n$ 
            somme  $\leftarrow 0$ 
            pour  $k = 1$  jusqu'à  $i-1$ 
                somme  $\leftarrow$  somme +  $\ell_{jk} \ell_{ik}$ 
            fait
             $\ell_{ji} \leftarrow \frac{a_{ji} - \text{somme}}{\ell_{ii}}$ 
    fait
     $\ell_{nn} \leftarrow \sqrt{a_{nn} - \text{somme}}$ 
    
```

## 6. Décomposition QR

### Définition (Matrice orthogonales)

On appelle matrice orthogonale une matrice dont les colonnes sont orthonormées. C'est à dire les matrices  $O$  telles que :  ${}^tOO = I$

Remarques :

- Si  $O$  est orthogonale,  $\det(O) = \pm 1$ .
- Elles ne changent pas la norme associée au produit scalaire :  

$$\|Ou\| = \|u\|$$
- Le produit de deux matrices orthogonales est une matrice orthogonale :  

$${}^t(OO')OO' = {}^tO{}^tOOO' = I$$

### Exemples :

- Matrice de rotation,

$$A = \begin{pmatrix} \cos \theta & -\sin(\theta) \\ \sin \theta & \cos(\theta) \end{pmatrix}$$

- Matrice de permutation,

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

- Matrice de symétries de HouseHolder :  $Q = I - 2vv^T$ .

En algèbre linéaire, la décomposition QR (appelée aussi, décomposition QU) d'une matrice  $A$  est une décomposition de la forme  $A = QR$  où  $Q$  est une matrice orthogonale ( $QQ^T = I$ ), et  $R$  une matrice triangulaire supérieure.

Comme la décomposition LU, cette décomposition QR transforme une matrice quelconque en produit de matrices plus simples  $A = QR$

### Théorème

*Soit  $A$  une matrice quelconque de  $M_{n,m}(C)$  alors:*

- $\exists Q \in M_{n,m}$  une matrice orthogonale,
- $\exists R \in M_{n,m}(C)$  une matrice triangulaire supérieure telles que  $A = QR$

- Une matrice orthogonale est facile à inverser ( ${}^tQ = Q^{-1}$ ).  
Cela peut être utilisé pour la résolution d'équations linéaires : au lieu de résoudre  $Ax = b$ , on résout  $Rx = {}^tQb$
- Cela fonctionne pour les matrices rectangulaires : résolution d'équations linéaires surdéterminés (avec plus d'équations que d'inconnues)
- Les matrices orthogonales ont de bonnes propriétés :
  - le déterminant de  $A$  est égale à celui de  $R$ ;
  - multiplier par une matrice orthogonale ne change pas la norme;
  - le conditionnement de  $A$  est égale à celui de  $R$ .
- Cette décomposition est aussi utilisée pour le calcul des valeurs propres d'une matrice (Méthodes des puissances).
- ...



Il existe plusieurs méthodes pour réaliser cette décomposition QR :

- la méthode de Householder où Q est obtenue par produits successifs de matrices orthogonales élémentaires
- la méthode de Givens où Q est obtenue par produits successifs de matrices de rotation plane
- la méthode de Schmidt

Chacune d'entre elles a ses avantages et ses inconvénients. (La décomposition QR n'étant pas unique, les différentes méthodes produiront des résultats différents).

## Méthode de Householder

Soit  $x$  un vecteur colonne arbitraire de dimension  $m$  et de longueur  $|\alpha|$

Soit  $e_1$  le vecteur  $(1, 0, \dots, 0)^T$ , et  $\| \cdot \|$  la norme euclidienne, définissons

$$u = x - \alpha e_1,$$

$$v = \frac{u}{\|u\|},$$

$$Q = I - 2vv^T.$$

Q est la matrice de Householder ou matrice orthogonale élémentaire et

$$Qx = (\alpha, 0, \dots, 0)^T.$$

Nous pouvons utiliser ces propriétés pour transformer une matrice A de dimension  $m \times n$  en une matrice triangulaire supérieure. Tout d'abord, on multiplie A par la matrice de Householder  $Q_1$  en ayant pris le soin de choisir pour  $x$  la première colonne de A. Le résultat est une matrice  $Q_1 A$  avec des zéros dans la première colonne excepté du premier élément qui vaudra  $\alpha$ .

$$Q_1 A = \begin{bmatrix} \alpha_1 & \star & \dots & \star \\ 0 & & & \\ \vdots & & A' & \\ 0 & & & \end{bmatrix}$$

Ceci doit être réitéré pour  $A'$  qui va être multiplié par  $Q'_2$  ( $Q'_2$  est plus petite que  $Q_1$ ). Si toutefois, vous souhaiteriez utiliser  $Q_1 A$  plutôt que  $A'$ , vous deviez remplir la matrice de Householder avec des 1 dans le coin supérieur gauche :

$$Q_k = \begin{pmatrix} I_{k-1} & 0 \\ 0 & Q'_k \end{pmatrix}$$

Après  $t$  itérations,  $t = \min(m - 1, n)$ ,

$$R = Q_t \cdots Q_2 Q_1 A$$

est une matrice triangulaire supérieure. Si  $Q = Q_1^T Q_2^T \cdots Q_t^T$  alors  $A = QR$  est la décomposition QR de A.

## Exemple :

Calculons la décomposition QR de

$$A = \begin{pmatrix} 12 & -51 & 4 \\ 6 & 167 & -68 \\ -4 & 233 & -41 \end{pmatrix}$$

On choisit donc le vecteur  $a_1 = (12, 6, -4)^T$ .

On a donc  $\|a_1\| = \sqrt{12^2 + 6^2 + (-4)^2} = 14$ . Ce qui nous conduit à écrire  $\|a_1\|e_1 = (14, 0, 0)^T$ .

Le calcul nous amène à  $u = (-2, 6, -4)^T$  et  $v = 14^{-\frac{1}{2}}(-1, 3, -2)^T$ . La première matrice de Householder vaut

$$Q_1 = I - \frac{2}{14} \begin{pmatrix} -1 \\ 3 \\ -2 \end{pmatrix} \begin{pmatrix} -1 & 3 & -2 \end{pmatrix}$$

$$= I - \frac{1}{7} \begin{pmatrix} 1 & -3 & 2 \\ -3 & 9 & -6 \\ 2 & -6 & 4 \end{pmatrix} = \begin{pmatrix} 6/7 & 3/7 & -2/7 \\ 3/7 & -2/7 & 6/7 \\ -2/7 & 6/7 & 3/7 \end{pmatrix}$$

Observons que:

$$Q_1 A = \begin{pmatrix} 14 & -271/7 & -14 \\ 0 & 911/7 & -14 \\ 0 & 1803/7 & -77 \end{pmatrix}$$

Nous avons maintenant sous la diagonale uniquement des zéros dans la 1<sup>re</sup> colonne.

Pour réitérer le processus, on prend la sous matrice principale

$$A' = M_{11} = \begin{pmatrix} 911/7 & -14 \\ 1803/7 & -77 \end{pmatrix}$$

Par la même méthode, on obtient la 2e matrice de Householder

$$Q_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -((911(-911 + \sqrt{4080730}))/(-4080730 + 911\sqrt{4080730})) & -((1803(-911 + \sqrt{4080730}))/(-4080730 + 911\sqrt{4080730})) \\ 0 & -((1803(-911 + \sqrt{4080730}))/(-4080730 + 911\sqrt{4080730})) & 1 + 3250809/(-4080730 + 911\sqrt{4080730}) \end{pmatrix}$$

Finalement, on obtient

$$Q = Q_1^T Q_2^T = \begin{pmatrix} 6/7 & (873(-911 + \sqrt{4080730}))/(-4080730 + 911\sqrt{4080730}) & -((1033(-911 + \sqrt{4080730}))/(-4080730 + 911\sqrt{4080730})) \\ 3/7 & -((8996(-911 + \sqrt{4080730}))/(-4080730 + 911\sqrt{4080730})) & (1296(-911 + \sqrt{4080730}))/(-4080730 + 911\sqrt{4080730}) \\ -2/7 & -((10875(-911 + \sqrt{4080730}))/(-4080730 + 911\sqrt{4080730})) & -((1155(-911 + \sqrt{4080730}))/(-4080730 + 911\sqrt{4080730})) \end{pmatrix}$$

$$R = Q^T A = \begin{pmatrix} 14 & -271/7 & -14 \\ 0 & -((4080730(-911 + \sqrt{4080730}))/(-4080730 + 911\sqrt{4080730})) & (151585(-911 + \sqrt{4080730}))/(-4080730 + 911\sqrt{4080730}) \\ 0 & 0 & -((44905(-911 + \sqrt{4080730}))/(-4080730 + 911\sqrt{4080730})) \end{pmatrix}$$

La matrice Q est orthogonale et R est triangulaire supérieure, par conséquent, on obtient la décomposition  $A = QR$ .

## 7. Calcul des valeurs propres par la méthode QR (méthode des puissances)

On effectue la décomposition QR de la matrice  $A$  de départ et on note  $A = Q_1 R_1$ , on calcule ensuite le produit  $A_2 = R_1 Q_1$ .

Une fois  $A_2 \dots A_k$  calculées, on effectue la décomposition QR de la matrice  $A_k$  que l'on note  $A_k = Q_k R_k$  et on calcule le produit  $A_{k+1} = R_k Q_k$ .

La matrice  $A_k$  ainsi construite vérifie donc l'égalité

$$A_k = Q_k^{-1} \dots Q_2^{-1} A Q_2 \dots Q_k$$

et a donc les mêmes valeurs propres que la matrice  $A$  de départ.

Voici le théorème de convergence :

### Théorème

*On suppose que  $A$  est inversible et que ses valeurs propres sont toutes de modules différents. Il existe donc une matrice inversible  $P$  telle que  $A = P \Lambda P^{-1}$  où  $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$  et  $|\lambda_1| > \dots > |\lambda_n| > 0$  et l'on suppose que la matrice  $P$  admet une factorisation LU. Alors la suite de matrice  $(A_k)$  est telle que*

$$\lim_{k \rightarrow +\infty} (A_k)_{i,i} = \lambda_i, 1 \leq i \leq n, \text{ et } \lim_{k \rightarrow +\infty} (A_k)_{i,j} = 0, 1 \leq j < i \leq n.$$

## 8. Comparaison des méthodes directes :

- Coût des méthodes de Gauss et Choleski pour une matrice d'ordre  $n$  :

	Gauss	Choleski
Additions	$(n^3-n)/3$	$n^3/6$
Multiplications	$(n^3-n)/3$	$n^3/6$
Divisions	$n(n-1)/2$	$n^2/2$
Racine		$n$
Total	$2n^3/3$	$n^3/3$

- Le coût de la méthode QR de House-Holder pour une matrice  $n \times n$  est proportionnel à :  $\frac{4}{3} \times n^3$ . Ce coût est relativement élevé (la méthode de Cholesky, pour les matrices symétriques définies positives est en  $\frac{1}{3} \times n^3$ ). Cependant, la méthode de Householder présente l'avantage considérable d'être beaucoup plus stable numériquement, en limitant les divisions par des nombres petits. La méthode de Givens, malgré un coût encore supérieur à celui-ci, offrira encore davantage de stabilité.

## **Chapitre 3 : Méthodes itératives de résolution d'un système linéaire**

### **1. Principe Général**

On cherche toujours à résoudre une équation de la forme :  $Ax = b$

Les méthodes directes fournissent la solution  $x^*$  en un nombre fini d'opérations.

Mais :

- Si la taille du système est élevée, le nombre d'opérations est important, or les erreurs de calcul dépendent directement du nombre de calculs.
- Elles utilisent des propriétés mathématiques nécessitant un calcul exact, il est difficile de tenir compte des erreurs de calcul dans ce processus.

Donc le résultat n'est jamais rigoureusement égal à  $x^*$ . Il peut même en être très différent.

Les fonctions linéaires ont de bonnes propriétés :

- Ce sont des fonctions très régulières ( $C^\infty$ ) elles sont linéaires
- L'utilisation de la formule de TAYLOR dans les méthodes numérique en général sert justement à « linéariser » une fonction non linéaire

On construit une suite de vecteurs  $(x_k)$   $k = 0, 1, \dots$  qui tend vers  $x^*$ .

Le point de départ est une approximation  $x_0$  de  $x^*$  obtenue par exemple par une méthode directe. Pour construire cette suite, on utilise la linéarité pour décomposer la matrice  $A$  en une partie facilement inversible et un reste.

On décompose la matrice  $A$  :  $A = M - N$ , de telle façon que  $M$  soit facilement inversible.

Alors,  $Ax = b \Rightarrow Mx = Nx + b$

On calcule la suite de vecteurs  $(x_i)$  à partir d'un vecteur  $x_0$  choisi arbitrairement et de la relation :  $Mx_{k+1} = Nx_k + b \Rightarrow x_{k+1} = M^{-1}Nx_k + M^{-1}b$

C'est à dire :

$$\begin{cases} x^0 & \text{donné} \\ x^{k+1} & = M^{-1}Nx^k + M^{-1}b \end{cases}$$

## 2. Convergence

Posons  $C = M^{-1}N$ , et  $d = M^{-1}b$ . Nous devons donc étudier la suite récurrente :

$$\begin{cases} x^0 & \text{donné} \\ x^{k+1} & = Cx^k + d \end{cases}$$

Avec :

■  $x^*$  est point fixe de la fonction linéaire

$$x \mapsto Cx + d$$

■ cette fonction est  $C^\infty$

■ il faut démontrer que c'est une fonction contractante.

**Théorème**  $\forall C \in \mathcal{M}_n(\mathbb{C})$ , s'il existe une norme matricielle induite  $\|\cdot\|$  telle que

$$\|C\| < 1$$

alors :

1. L'équation  $x = Cx + d$  admet une solution unique  $\bar{x}$ .
2. La suite  $x^k \rightarrow \bar{x}$  quelle que soit  $x^0$ .

### Démonstration

Existence de la solution :

D'après les propriétés du rayon spectral on a :

$$\rho(C) \leq \|C\| < 1$$

Donc les valeurs propres  $\lambda$  de  $C$  sont telles que  $|\lambda| < 1$ .

Cela signifie que la matrice  $I - C$  est inversible

Donc il existe une solution unique à l'équation

$$x = Cx + d$$

On appelle  $\bar{x}$  cette solution

Convergence :

Soit  $e^k = x^k - \bar{x}$

On peut déduire une relation entre  $e^k$  et  $e^{k-1}$ .

$$\begin{aligned} Ce^{k-1} &= C(x^{k-1} - \bar{x}) \\ &= C(x^{k-1}) - C(\bar{x}) \\ &= C(x^{k-1}) + d - \bar{x} \end{aligned}$$

Donc  $e^k = Ce^{k-1}$  pour  $k = 1, 2, \dots$

Nous avons alors :

$$e^k = C^k e^0$$

Soit la norme matricielle induite  $\|\cdot\|$  et sa norme vectorielle  $\|\cdot\|_v$  telle que  $\|C\| < 1$  :

$$\|e^k\|_v \leq \|C\|^k \|e^0\|_v.$$

Donc  $e^k \rightarrow 0$  et  $x^k \rightarrow \bar{x}$

**Théorème** Soit  $A$  une matrice symétrique définie positive, Si

$$A = M - N$$

et si  $M + {}^t N$  est définie positive (elle est forcément symétrique) alors la suite

$$x^{k+1} = M^{-1}Nx^k + d$$

est convergente

**idée de la preuve :**

Si on considère la norme vectorielle définie par  $A$  :

$$\|x\|_A = \sqrt{{}^t x A x}$$

Et sa norme matricielle induite  $\|\cdot\|$ , alors

$$\|A\| < 1$$

## 3. Méthode de JACOBI

Soit la matrice A, on note :

- D la matrice des éléments diagonaux de A
- E la matrice des éléments sous-diagonaux
- F la matrice des éléments sur-diagonaux

C'est à dire :

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & a_{ij} & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix} \quad D = \begin{pmatrix} a_{11} & & 0 \\ & \ddots & \\ 0 & & a_{nn} \end{pmatrix}$$

$$E = \begin{pmatrix} 0 & & 0 \\ a_{ij} & \ddots & \\ & & 0 \end{pmatrix} \quad F = \begin{pmatrix} 0 & a_{ij} & \\ & \ddots & \\ 0 & & 0 \end{pmatrix}$$

L'algorithme de JACOBI décompose la matrice en la somme  $A = M - N$  et calcule la suite

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

Avec :

$$\begin{cases} M = D \\ N = -E - F \end{cases}$$

On appelle **matrice de JACOBI** la matrice

$$\begin{aligned} J &= D^{-1}(-E - F) \\ &= M^{-1}N \end{aligned}$$

## Conditions de convergence

D'après ce qui précède une condition nécessaire et suffisante (CNS) pour la convergence de la méthode de Jacobi est que :

- D soit inversible : les éléments diagonaux de A doivent être non nuls
- $\rho(J) < 1$  (la valeur propre de plus grand module est  $< 1$ )

Remarque :

Cette CNS est difficile à vérifier, mais il y a deux conditions suffisantes.

- Pour les matrices à diagonale strictement dominante.
- Pour les matrices symétriques.

**Définition** Une matrice A est dite à diagonale strictement dominante si :

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

**Théorème** Si A est une matrice à diagonale strictement dominante, alors la méthode de JACOBI est convergente quel que soit le vecteur initial  $x^0$ .

**Preuve :**

Si  $|a_{ii}| > \sum_{j=1, j \neq i}^n |a_{ij}| \Rightarrow a_{ii} \neq 0$

Pour démontrer la convergence, on peut prouver que  $\|C\|_{\infty} < 1$  avec  $C = D^{-1}(-E - F)$ . Donc  $C_{ii} = 0$ ,  $C_{ij} = -\frac{a_{ij}}{a_{ii}}$  pour  $i \neq j$ .

$$\|C\|_{\infty} = \max_i \sum_{j=1}^n |C_{ij}| = \max_i \left( \sum_{j=1}^n |C_{ij}| \right) = \max_i \left( \sum_{j=1, j \neq i}^n \frac{|a_{ij}|}{|a_{ii}|} \right) < 1, \text{ puisque}$$

A est à diagonale strictement dominante.

**Théorème** Si A et  $2D - A$  sont symétriques définies positives, alors la méthode de JACOBI converge

**preuve :** Si on pose

$$\begin{aligned} M &= D \\ N &= -E - F \\ &= D - A \end{aligned}$$

alors  $M + {}^t N = 2D - A$



**Algorithme**

**Données :**  $A, b, x^0, n, \varepsilon$  et MAXITER

**début**

```

pour  $i = 1$  à  $n$  faire
   $x_i^{new} \leftarrow x_i^0$ 
nb  $\leftarrow 0$ 
tant que  $(\|Ax^{new} - b\| > \varepsilon)$  et  $(nb < MAXITER)$  faire
  nb  $\leftarrow nb + 1$ 
  pour  $i = 1$  à  $n$  faire
     $x_i^{old} \leftarrow x_i^{new}$ 
  pour  $i = 1$  à  $n$  faire
     $x_i^{new} \leftarrow \frac{b_i - \sum_{j=1, j \neq i} a_{ij} x_j^{old}}{a_{ii}}$ 

```

**fin**

**4. Méthode de GAUSS-SEIDEL**

L'algorithme de GAUSS-SEIDEL décompose la matrice en la somme  $A = M - N$  et calcule la suite

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

Avec :

$$\begin{cases} M = D + E \\ N = -F \end{cases}$$

On appelle *matrice de GAUSS-SEIDEL* la matrice

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

## Conditions de convergence

D'après ce qui précède, une condition nécessaire et suffisante (CNS) est que :

- $(D+E)$  est inversible  
 $\Rightarrow$  les éléments diagonaux de  $A$  doivent être non nuls.
- et  $\rho(GS) < 1$  avec  $GS = -(D + E)^{-1}F$ .

Il y a aussi deux conditions suffisantes :

**Théorème** Si  $A$  est une matrice à diagonale strictement dominante, alors la méthode de GAUSS-SEIDEL est convergente quel que soit le vecteur initial  $x^0$ .

**Théorème** Si  $A$  est une matrice symétrique définie positive alors la méthode de GAUSS-SEIDEL est convergente quel que soit le vecteur initial  $x^0$ .

## Algorithme

**Données :**  $A, b, x^0, n, \varepsilon$  et MAXITER

**début**

pour  $i = 1$  à  $n$  faire

└  $x_i^{new} \leftarrow x_i^0$

nb  $\leftarrow 0$

tant que  $(\|Ax^{new} - b\| > \varepsilon)$  et  $(nb < MAXITER)$  faire

└ nb  $\leftarrow nb + 1$

pour  $i = 1$  à  $n$  faire

└  $x_i^{old} \leftarrow x_i^{new}$

pour  $i = 1$  à  $n$  faire

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

**fin**

## **Chapitre 4 : Méthodes de résolution des équations non linéaires**

### **1. Introduction**

On considère une fonction continue  $f : \mathbb{R} \rightarrow \mathbb{R}$ . On se propose de trouver une ou plusieurs solutions à l'équation  $f(x) = 0$ .

Les méthodes analytiques de résolution de ce type d'équation sont limitées à certaines formes algébriques ( $a_n x^n + a_{n-1} x^{n-1} + \dots + a_0 = 0$  pour  $n < 5$ ) ou formes particulières. Par conséquent pour les autres formes d'équations il faut employer des méthodes numériques pour trouver ou approcher les racines.

Dans toute la suite on va supposer qu'on dispose d'un intervalle  $[a, b]$  où la fonction  $f$  admet une seule solution  $\alpha$  à l'intérieur de l'intervalle. Cette localisation de la racine peut être fournie par des considérations à priori sur le problème à l'origine de l'équation (des considérations physique, d'ordre de grandeur ou de conditions initiales). On suppose aussi que la fonction est dérivable autant de fois qu'il sera nécessaire.

### **2. Méthode de Dichotomie**

La dichotomie (du Grec diviser par deux) peut être vue comme une variante simplifiée de la stratégie plus générale diviser pour régner. Prenons un exemple simple et ludique pour illustrer le mécanisme de recherche par dichotomie: Mohamed propose à Moustapha le jeu suivant: « choisis en secret un nombre compris entre 0 et 100; je vais essayer de le deviner le plus rapidement possible, mais tu ne dois répondre à mes questions que par oui ou par non ». Moustapha choisit 65 et attend les questions de Mohamed:

- est-ce que le nombre est plus grand que 50? (100 divisé par 2)
- oui
- est-ce que le nombre est plus grand que 75? ((50 + 100) / 2)
- non
- est-ce que le nombre est plus grand que 63? ((50 + 75 + 1) / 2)
- oui

Mohamed réitère ses questions jusqu'à trouver 65. Par cette méthode itérative, Mohamed est sûr de trouver beaucoup plus rapidement le nombre qu'en posant des questions du type « est-ce que le nombre est égal à 30? ».

La Dichotomie est aussi la plus simple méthode connue pour chercher une racine d'une fonction  $f$  continue sur un intervalle fermé  $[a, b]$  :

#### **Algorithme**

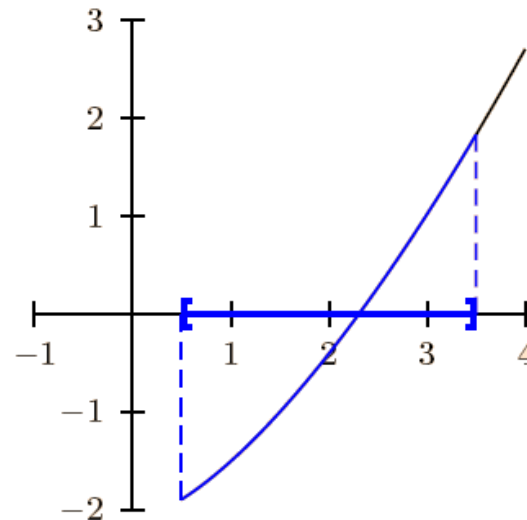
- On commence par prendre  $x_0 = (a+b)/2$  le milieu de l'intervalle
- On a alors trois cas possibles :
  - Si  $f(x_0)f(a) < 0$  la racine se trouve dans l'intervalle  $[a, x_0]$  et on pose dans ce cas :  $[a_1, b_1] = [a, x_0]$
  - Si  $f(x_0)f(b) < 0$  la racine se trouve dans l'intervalle  $[x_0, b]$  et on pose dans ce cas :  $[a_1, b_1] = [x_0, b]$

- Si  $f(x_0)=0$  alors  $x_0$  est la racine recherchée
- Si la racine n'est pas atteinte on recommence l'opération avec l'intervalle  $[a_1, b_1]$
- On obtient ainsi une suite  $(x_n)_n$  avec  $x_n=(a_n+b_n)/2$

On remarque qu'à l'itération  $n$  on a  $|x_n - \alpha| < (b-a)/2^n$  ceci permet de connaître à l'avance le nombre d'itération nécessaires pour approcher la solution avec une précision donnée.

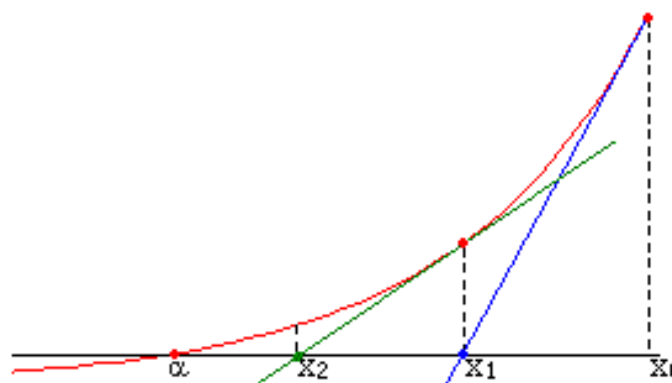
```

Données :  $f, a < b.$ 
début
  tant que  $|f(c)| > \varepsilon$ 
    faire
       $c \leftarrow a + \frac{b-a}{2}$ 
      si  $f(a)f(c) < 0$ 
        alors
           $b \leftarrow c$ 
        sinon
           $a \leftarrow c$ 
  Résultat :  $c$ 
fin
    
```



### 3. Méthode de Newton

On suppose ici que la fonction est dérivable sur l'intervalle  $[a, b]$ . Le principe de la méthode de Newton, qu'on appelle aussi méthode de la tangente, consiste à choisir un point  $x_0$  dans l'intervalle et à remplacer la fonction  $f$  par sa tangente en ce point, puis on calcule la racine de cette approximation affine dans l'intervalle  $[a, b]$ . En général la racine de cette tangente est plus proche que  $x_0$  de la racine de  $f$ . On peut donc continuer itérativement (par la construction d'une suite  $(x_n)_n$ ) de la même façon jusqu'à ce qu'on s'approche suffisamment de la racine de  $f$ .



**Algorithme :**

- L'équation de la tangente est donnée par :  $(y-f(x_0))/(x-x_0)=f'(x_0)$
- La racine de la tangente est donné par :  $x_1=x_0 - f(x_0)/f'(x_0)$
- La suite  $(x_n)_n$  est donc donnée par la formule itérative :

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

**Données :**  $x_0$ , NMAX et  $\varepsilon$

**début**

$n \leftarrow 0$

**répéter**

$n \leftarrow n + 1$

$x \leftarrow x_0$

$x_0 \leftarrow x - \frac{f(x)}{f'(x)}$

**jusqu'à**  $(|x - x_0| \leq \varepsilon)$  ou  $(n = NMAX)$

**Résultat :** si  $(n = NMAX)$  alors

écrire "Trop d'itérations"

rendre NaN

**sinon**

rendre  $x_0$

**fin**

## Exemple

Considérons le problème de trouver le nombre positif  $x$  vérifiant  $\cos(x) = x^3$ . Reformulons la question pour introduire une fonction devant s'annuler : on recherche le zéro de  $f(x) = \cos(x) - x^3$ . La dérivation donne  $f'(x) = -\sin(x) - 3x^2$ . Comme  $\cos(x) \leq 1$  pour tout  $x$  et  $x^3 > 1$  pour  $x > 1$ , nous savons que notre zéro se situe entre 0 et 1. Nous essayons une valeur de départ de  $x_0 = 0.5$ .

$$\begin{array}{llll} x_1 & = & x_0 - \frac{f(x_0)}{f'(x_0)} & = \frac{\cos(0,5) - 0,5^3}{-\sin(0,5) - 3 \times 0,5^2} \simeq 1,1121416371 \\ x_2 & = & x_1 - \frac{f(x_1)}{f'(x_1)} & \vdots \simeq 0,909672693736 \\ x_3 & & \vdots & \vdots \simeq 0,867263818209 \\ x_4 & & \vdots & \vdots \simeq 0,865477135298 \\ x_5 & & \vdots & \vdots \simeq 0,865474033111 \\ x_6 & & \vdots & \vdots \simeq 0,865474033101 \\ x_7 & & \vdots & \vdots \simeq 0,865474033102 \end{array}$$

et les 12 premiers chiffres de cette valeur coïncident avec les 12 premiers chiffres du vrai zéro.

## 4. Méthode de Lagrange

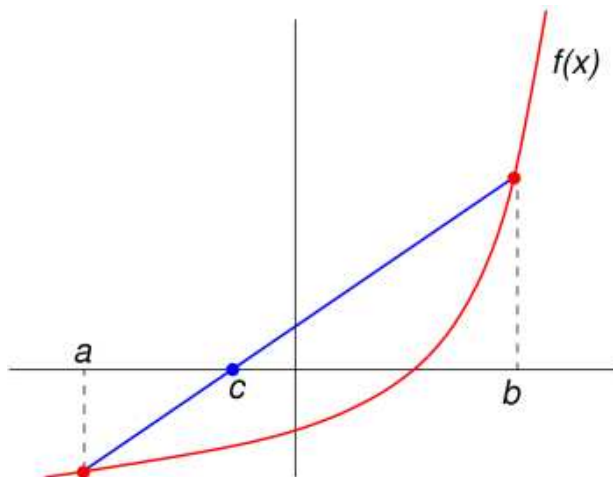
On l'appelle aussi la méthode de la sécante car elle consiste à remplacer la tangente de la méthode de Newton par une sécante qui est plus facile à calculer. En effet un

défaut de la méthode de Newton est la nécessité de pouvoir calculer la dérivée ce qui n'est toujours aisé dans les problèmes pratiques. On peut donc remplacer le calcul exact de la dérivée par une approximation qui ne fait intervenir que les valeurs de la fonction :

$$f'(x_n) \approx \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$$

On obtient ainsi la méthode itérative suivante :

$$x_{n+1} = x_n - \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} f(x_n).$$



**Remarque :**

Dans une autre variante on utilise l'approximation de la dérivée suivante :

$$f'(x_n) \approx \frac{f(x_n) - f(a)}{x_n - a} \quad \text{ou} \quad \frac{f(x_n) - f(b)}{x_n - b}$$

**Exemple :**

$$f(x) = \cos(x) - x^3 = 0$$

$$\begin{aligned} x_0 &= 0 \\ x_1 &= 1 \\ x_2 &= 0,685073357326045 \\ x_3 &= 0,841355125665652 \\ x_4 &= 0,870353470875526 \\ x_5 &= 0,865358300319342 \\ x_6 &= 0,865473486654304 \\ x_7 &= 0,865474033163012 \\ x_8 &= 0,865474033101614 \end{aligned}$$

## 5. Etude des méthodes d'approximation successives

### Convergence d'une méthode

Dans ces méthode on construit une suite  $(x_n)_n$  qui converge vers la solution de l'équation

$f(x)=0$ . Pour engendrer une telle suit on remplace cette équation par une équation équivalente de la forme  $g(x)=x$  avec  $g$  aussi une fonction continue :

$$f(x) = 0 \Leftrightarrow g(x) = x$$

On remplace ainsi le problème du calcul d'une racine de  $f$  par le calcul d'un point fixe de  $g$ .

La suite  $(x_n)_n$  est donnée par :

$$\begin{cases} x_0 \in [a, b] \\ x_{n+1} = g(x_n) \end{cases}$$

Si la suite converge alors elle convergera vers la solution du problème. En effet si  $l$  est la limite de la suite alors par continuité de  $g$  on a  $l=g(l)$ , c'est-à-dire que  $l$  est un point fixe de la fonction  $g$  et par équivalence c'est aussi une racine de la fonction  $f$ .

Il y a une infinité de façons pour choisir la fonction  $g$ . On peut par exemple utiliser simplement  $g(x)=x-f(x)$ . La méthode de Newton et celle de Lagrange sont aussi des exemples d'approximations successives :

- Pour la méthode de Newton :  $g(x) = x - \frac{f(x)}{f'(x)}$
- Pour la méthode de Lagrange :  $g(x) = x - \frac{f(x)(x-a)}{f(x)-f(a)}$  ou  $x - \frac{f(x)(x-b)}{f(x)-f(b)}$

On peut étudier la convergence des méthodes des approximations successives grâce au fameux théorème du point fixe.

### Théorème 1

On suppose que la fonction  $g$  est continue sur l'intervalle  $[a,b]$  et qu'elle vérifie les conditions suivantes :

1. Si  $x \in [a,b]$  alors  $g(x) \in [a,b]$
2. La fonction  $g$  est contractante dans l'intervalle  $[a,b]$ , c'est-à-dire que :  
il existe un réel  $K$  :  $0 \leq K < 1$  tel que pour tout  $x$  et  $y$  dans  $[a,b]$  on a :  
 $|g(x) - g(y)| \leq K|x - y|$

Alors pour tout  $x_0$  dans l'intervalle  $[a,b]$  la suite récurrente  $x_{n+1} = g(x_n)$  converge vers l'unique de l'équation  $g(x)=x$  dans l'intervalle  $[a,b]$ .

### Démonstration

D'abord si la suite  $(x_n)_n$  converge alors, d'après la condition 2, la limite est dans l'intervalle  $[a,b]$ .

Ensuite l'équation  $g(x) = x$  admet une solution unique dans  $[a,b]$  en effet si  $l_1$  et  $l_2$  sont deux solutions alors  $g(l_1)=l_1$  et  $g(l_2)=l_2$  , or d'après la condition 2 on a :  
 $|g(l_1) - g(l_2)| \leq K|l_1 - l_2|$  c'est-à-dire que  $|l_1 - l_2| \leq K|l_1 - l_2|$  donc forcément  $l_1 = l_2$  car  $0 \leq K < 1$ .

Enfin montrons que la suite  $(x_n)_n$  converge :

On a :  $|x_{n+1} - x_n| = |g(x_n) - g(x_{n-1})| \leq K|x_n - x_{n-1}|$

D'où :  $|x_{n+1} - x_n| \leq K^n |x_1 - x_0|$

De même comme on a :  $|x_{n+p} - x_n| \leq |x_{n+p} - x_{n+p-1}| + \dots + |x_{n+1} - x_n|$

On peut dire que :

$$|x_{n+p} - x_n| \leq |x_1 - x_0| K^n (K^{p-1} + K^{p-2} + \dots + 1) = |x_1 - x_0| K^n \frac{1 - K^p}{1 - K} \leq |x_1 - x_0| \frac{K^n}{1 - K} \quad \text{car } K < 1$$

Ainsi pour p fixé  $\lim_{n \rightarrow \infty} |x_{n+p} - x_n| = 0$  c'est-à-dire que la suite  $(x_n)_n$  est une suite de Cauchy donc convergente ■

On peut identifier les fonctions contractantes à partir de leurs dérivées

## Théorème 2

Si la fonction g est dérivable sur [a,b] et si la dérivée g' vérifie :  $\max_{x \in [a,b]} |g'(x)| = K < 1$  alors g est une fonction contractante sur l'intervalle [a,b].

## Démonstration

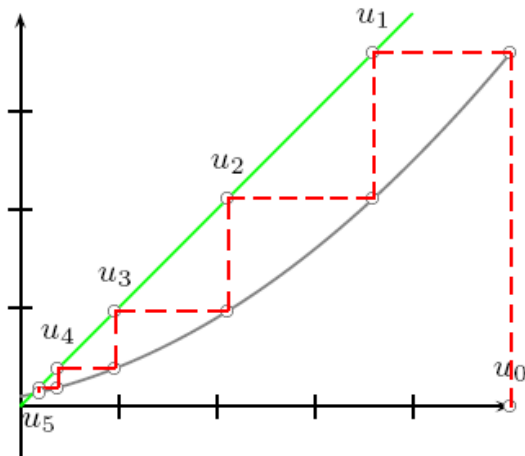
On utilise la formule des accroissements finies : pour tout x, y  $\in [a, b]$  on peut trouver  $\xi \in ]x, y[$  telle que  $|g(x) - g(y)| = |g'(\xi)| |x - y| \leq K |x - y|$  ■

## Corollaire 1

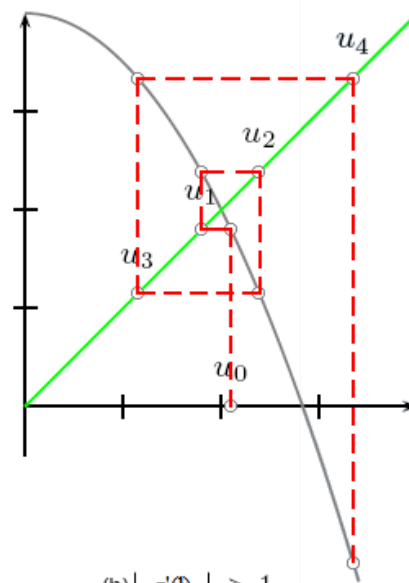
Soit l une solution de l'équation g(l)=l et g' continue au voisinage de l alors on a les 3 cas suivantes :

1. **Point fixe attractif** : si  $|g'(l)| < 1$  alors il existe un intervalle [a,b] contenant l pour lequel  $\forall x_0 \in [a,b]$  la suite  $(x_n)_n$  définie par  $x_{n+1} = g(x_n)$  converge vers l
2. **Point fixe répulsif** : si  $|g'(l)| > 1$  alors  $\forall x_0 \neq l$  la suite  $(x_n)_n$  définie par  $x_{n+1} = g(x_n)$  ne converge pas vers l
3. **Point fixe douteux** : si  $|g'(l)| = 1$  on ne peut pas conclure il peut y avoir convergence ou divergence.





(a)  $|g'(l)| < 1$



(b)  $|g'(l)| > 1$

## Ordre d'une méthode

En plus de la convergence, on veut aussi savoir si la suite  $(x_n)_n$ , définie par  $x_{n+1} = g(x_n)$ , engendrée par une méthode donnée converge assez rapidement. Ceci revient à étudier comment diminue la valeur  $|x_n - l|$  au cours des itérations.

### Définition 1

Une méthode définie par une suite  $(x_n)_n$  est dite d'ordre  $p$  si et seulement si la valeur de  $\frac{|x_{n+1} - l|}{|x_n - l|^p}$  tends vers une limite finie non nulle

### Théorème 3

Si la suite  $(x_n)_n$ , définie par  $x_{n+1} = g(x_n)$  converge vers  $l$  et si  $g$  est suffisamment dérivable au voisinage de  $l$  alors l'ordre de la méthode est donnée par :

$$\begin{cases} g'(l) = g''(l) = \dots = g^{(p-1)}(l) = 0 \\ g^{(p)}(l) \neq 0 \end{cases}$$

De plus on a :

$$\lim_{n \rightarrow \infty} \frac{(x_{n+1} - l)}{(x_n - l)^p} = \frac{g^{(p)}(l)}{p!}$$

### Démonstration

On utilise le développement de Taylor :

$$(x_{n+1} - l) = (g(x_n) - g(l)) = \sum_{k=1}^p g^{(k)}(l) \frac{(x_n - l)^k}{k!} + o((x_n - l)^{p+1})$$

$$\text{donc : } (x_{n+1} - l) = g^{(p)}(l) \frac{(x_n - l)^p}{p!} + o((x_n - l)^{p+1})$$

$$\text{d'où : } \frac{(x_{n+1} - l)}{(x_n - l)^p} = \frac{g^{(p)}(l)}{p!} + o(x_n - l) \blacksquare$$

### Exemples :

- Pour la méthode de Newton : on a  $g(x) = x - \frac{f(x)}{f'(x)}$

donc  $g'(x) = \frac{f(x)f''(x)}{(f'(x))^2}$  ainsi si  $f'(l) \neq 0$  on a  $g'(l) = 0$  car  $f(l) = 0$  donc la méthode

est au moins d'ordre 2 on dit que la méthode de Newton a une convergence quadratique.

- Pour la méthode de Lagrange : dans le cas où on a  $g(x) = x - \frac{f(x)(x-a)}{f(x)-f(a)}$

On obtient  $g'(l) = \frac{f(a) + f'(l)(l-a)}{f(a)}$  donc si  $g'(l) \neq 0$  ( ce qui est possible) la

méthode est d'ordre 1 on dit que la méthode de Lagrange a une convergence linéaire.

## 6. Méthodes pour les systèmes d'équations non linéaires

**Méthode de Newton en dimension  $>1$  :**

$$\begin{aligned} \text{Si } f : \mathbb{R}^n &\longrightarrow \mathbb{R}^n \\ \vec{x} &\longmapsto (f_1(\vec{x}), \dots, f_n(\vec{x})) \end{aligned}$$

on utilise la matrice Jacobienne de  $f$  :

$$\nabla f(x) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1}(x) & \dots & \frac{\partial f_1}{\partial x_i}(x) & \dots & \frac{\partial f_1}{\partial x_n}(x) \\ \vdots & & \vdots & & \vdots \\ \frac{\partial f_i}{\partial x_1}(x) & \dots & \frac{\partial f_i}{\partial x_i}(x) & \dots & \frac{\partial f_i}{\partial x_n}(x) \\ \vdots & & \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1}(x) & \dots & \frac{\partial f_n}{\partial x_i}(x) & \dots & \frac{\partial f_n}{\partial x_n}(x) \end{pmatrix}$$

Pour calculer d'une manière itérative la solution  $r$  de  $f(x)=0$  :

- On considère le développement de Taylor d'ordre 2:

$$f(x+h) = f(x) + \nabla f(x)h + \frac{1}{2}h' H_f(x)h + \|h\| \mathcal{E}(h^2)$$

- On choisit  $x_0$  suffisamment proche de la solution  $r$ .
- A chaque itération  $k$  :

- On considère l'approximation affine :

$$f(x_k + h) \approx f(x_k) + \nabla f(x_k)h$$

- Et on cherche  $h$  tel que  $f(x_k+h)=0$  :

- On construit le vecteur  $f(x_k) = b$ .

- On construit la matrice Jacobienne  $A = \nabla f(x_k)$
- On résout le système  $Ay = b$
- On pose  $x_{k+1} = x_k - y$

**Données :**  $x_0$ ,  $NMAX$  et  $\varepsilon$

**début**

$n \leftarrow 0$

**répéter**

$n \leftarrow n + 1$

$x \leftarrow x_0$

$b \leftarrow f(x)$

$A \leftarrow \nabla f(x)$

**résoudre**  $Ay = b$

$x_0 \leftarrow x - y$

**jusqu'à**  $(\|x - x_0\| \leq \varepsilon) \text{ ou } (n = NMAX)$

**Résultat :** **si**  $(n = NMAX)$  **alors**

| échouer

**sinon**

| rendre  $x_0$

**fin**

## Condition de convergence

Si

•  $\exists \vec{a} \in \mathbb{R}^n$  tel que  $f(\vec{a}) = \vec{0}$

•  $f$  est différentiable sur un voisinage  $\mathcal{V}$  de  $a$  et  $\forall x \in \mathcal{V}$

$$\|\nabla f(x) - \nabla f(a)\| \leq \alpha \|x - a\|$$

•  $\nabla f(a)$  inversible

alors  $\exists \eta > 0$  t.q. si  $\|x_0 - a\| < \eta$  la méthode de NEWTON converge vers  $a$  de façon quadratique.

• À chaque itération, il faut inverser la matrice  $\nabla f(x)$

• Le point de départ doit toujours être au voisinage de la solution

## Avantages de la méthode de Newton :

- Convergence très rapide.

## Inconvénients :

- La convergence n'est pas assurée.
- Il faut choisir un point de départ « au voisinage de la solution ».
- En dimension  $n$ , il faut inverser  $\nabla f(x_n)$

## Chapitre 5 : Introduction à l'optimisation

### 1. Formulation générale des problèmes d'optimisation non linéaire

La forme générale d'un problème d'optimisation est la suivante :

$$(PC) \left\{ \begin{array}{ll} \min_{x \in \mathbb{R}^n} f(x), & (I.1.1) \\ \text{sous les contraintes} & \\ g(x) \leq 0, & (I.1.2) \\ h(x) = 0, & (I.1.3) \end{array} \right.$$

où les fonctions  $f$ ,  $g$  et  $h$  sont typiquement non-linéaires.

L'équation (I.1.2) désigne ce que nous appelleront des contraintes d'inégalité et l'équation (I.1.3) des contraintes d'égalité.

Nous nous limiterons dans ce cours à l'étude des problèmes d'optimisation sans contraintes.

### 2. Rappels de calcul différentiel

**Définition** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$  représentée dans la base canonique de  $\mathbb{R}^m$  par le vecteur

$$f(x) = \begin{pmatrix} f_1(x) \\ \vdots \\ f_m(x) \end{pmatrix}, \quad (I.3.1)$$

continue en  $a \in \mathbb{R}^n$ . On dit que  $f$  est différentiable en  $a$  s'il existe une application linéaire, notée  $f'(a)$ , telle que pour tout  $h \in \mathbb{R}^n$  on ait

$$f(a+h) = f(a) + f'(a)h + \|h\| \epsilon(h), \quad (I.3.2)$$

où  $\epsilon(\cdot)$  est une fonction continue en 0 vérifiant  $\lim_{h \rightarrow 0} \epsilon(h) = 0$ . On appelle  $f'(a)$  dérivée de  $f$  au point  $a$ .

**Proposition** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$  différentiable en  $a$ , alors on peut représenter  $f'(a)$  par sa matrice dans les bases canoniques de  $\mathbb{R}^n$  et de  $\mathbb{R}^m$  et on a

$$[f'(a)]_{ij} = \frac{\partial f_i}{\partial x_j}(a)$$

On appelle souvent  $f'(a)$  la matrice jacobienne de  $f$  au point  $a$ . Lorsque  $m = 1$  on adopte une notation et un nom particuliers : le gradient est le vecteur noté  $\nabla f(a)$  et défini par

$$f'(a) = \nabla f(a)^\top,$$

On se place maintenant dans le cas  $m = 1$ , soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ .

**Définition I.3.3.** L'application  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  est dite deux fois différentiable s'il existe une matrice symétrique  $\nabla^2 f(a)$  telle que

$$f(a+h) = f(a) + \nabla f(a)^T h + h^T \nabla^2 f(a) h + \|h\|^2 \epsilon(h).$$

On appelle  $\nabla^2 f(a)$  matrice hessienne de  $f$  au point  $a$ . Comme l'énonce le théorème suivant (non démontré), cette matrice s'obtient à partir des dérivées secondes de  $f$  :

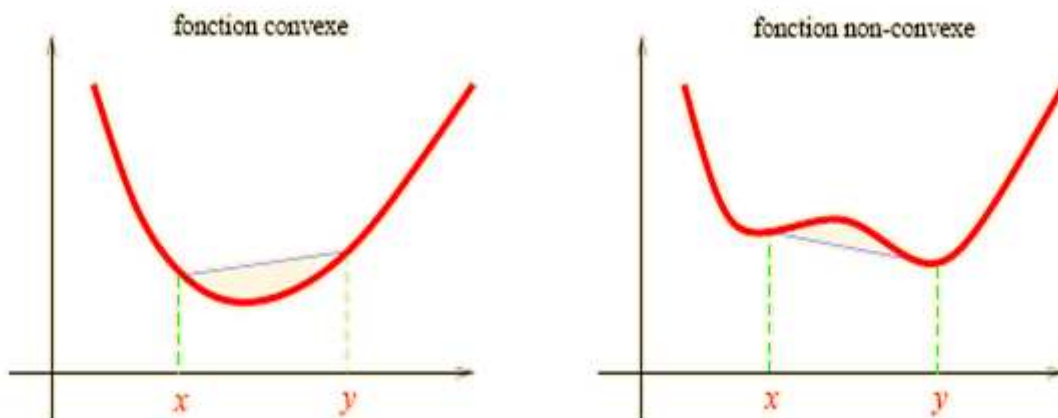
**Théorème I.3.4.** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction deux fois différentiable en un point  $a$ . Si on note  $g(x) = \nabla f(x)$  alors la matrice hessienne est définie par  $\nabla^2 f(a) = g'(a)$ , soit

$$[\nabla^2 f(a)]_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}.$$

## Définition

On dit qu'une fonction  $f : K \subset \mathbb{R}^n \rightarrow \mathbb{R}$ , définie sur un ensemble convexe  $K$ , est convexe si elle vérifie

$$\forall (x, y) \in K^2, \forall \lambda \in [0, 1], f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y).$$



## Théorème

Soit  $f : K \subset \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction deux fois différentiable, alors  $f$  est convexe si et seulement si  $\nabla^2 f(x) \geq 0, \forall x \in K$ , et strictement convexe si et seulement si  $\nabla^2 f(x) > 0, \forall x \in K$ .

## 3. Théorèmes généraux d'existence et d'unicité

**Théorème** Si  $f : K \subset \mathbb{R}^n \rightarrow \mathbb{R}$  est continue et si de plus  $K$  est un ensemble compact, alors le problème  $\min_{x \in K} f(x)$  admet une solution optimale  $\hat{x} \in K$ , qui vérifie donc

$$f(\hat{x}) \leq f(x), \forall x \in K.$$

**Théorème** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction continue sur  $\mathbb{R}^n$ .

Si  $\lim_{\|x\| \rightarrow \infty} f(x) = \infty$ , alors le problème  $\min_{x \in K} f(x)$

admet une solution optimale  $\hat{x}$ .

L'unicité résulte en général de propriétés de convexité (de  $f$  et de  $K$ ).

**Théorème**

Soit  $f : K \subset \mathbb{R}^n \rightarrow \mathbb{R}$  strictement convexe sur  $K$  convexe.

Le minimum de  $f$  sur  $K$ , s'il existe, est unique.

## 4. Optimisation sans contraintes

### Conditions nécessaires d'optimalités

On va maintenant regarder de plus près le cas où  $K = \mathbb{R}^n$ ,

c'est à dire le problème sans contraintes (P).

Dans le cas où  $f$  est différentiable, on a le résultat suivant :

**Théorème** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  différentiable et  $\hat{x}$  vérifiant

$$f(\hat{x}) \leq f(x), \forall x \in \mathbb{R}^n,$$

alors on a nécessairement

$$\nabla f(\hat{x}) = 0.$$

La condition de gradient nul devient suffisante dans le cas où  $f$  est convexe :

**Théorème**

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  convexe et différentiable. Si  $\hat{x}$  vérifie

$$\nabla f(\hat{x}) = 0,$$

alors on a  $f(\hat{x}) \leq f(x), \forall x \in \mathbb{R}^n$ .



Lorsque la fonction n'est pas convexe, on ne peut donner qu'une condition nécessaire et suffisante d'optimalité locale.

**Définition** . On appellera  $x^*$  minimum local de  $f$ , s'il existe  $\delta > 0$  tel que

$$f(x^*) \leq f(x), \forall x, \|x - x^*\| \leq \delta.$$

Dans le cas où  $f$  est deux fois différentiable on peut alors donner le résultat suivant :

**Théorème** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  deux fois différentiable. Si 
$$\begin{cases} \nabla f(x^*) = 0, \\ \nabla^2 f(x^*) > 0, \end{cases}$$

alors  $x^*$  est un minimum local de  $f$ .

## 5. Principe des méthodes de descentes

On considère le problème d'optimisation suivant :

$$\min_{x \in \mathbb{R}^n} f(x)$$

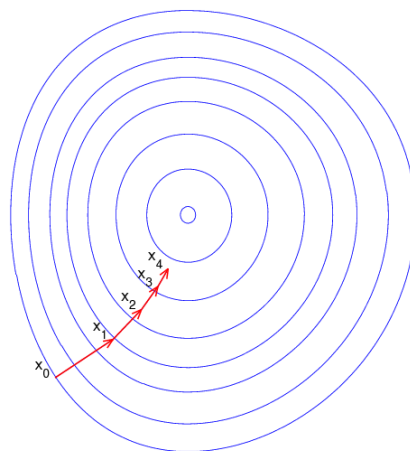
### Définition

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ . On dira qu'un vecteur  $d$  est une direction de descente en  $x$  s'il existe  $\bar{t} > 0$  tel que  $f(x + td) < f(x)$ ,  $t \in ]0, \bar{t}]$ .

Le principe d'une méthode de descente consiste à faire les itérations suivantes  $x_{k+1} = x_k + t_k d_k$ ,  $t_k > 0$ ,

tout en assurant la propriété  $f(x_{k+1}) < f(x_k)$ .

Le vecteur  $d_k$  est la direction de descente en  $x_k$ . Le scalaire  $t_k$  est appelé le pas de la méthode à l'itération  $k$ .



On peut caractériser les directions de descente à l'aide du gradient :

**Proposition** Soit  $d \in \mathbb{R}^n$  vérifiant  $\nabla f(x)^\top d < 0$ ,

alors  $d$  est une direction de descente en  $x$ .

*Démonstration :* on a pour  $t > 0$

$$f(x + td) = f(x) + t \nabla f(x)^\top d + t\varepsilon(t),$$

donc si on écrit

$$\frac{f(x + td) - f(x)}{t} = \nabla f(x)^\top d + \varepsilon(t),$$

### 6. Méthode de gradient

On cherche la direction de descente qui fait décroître localement la fonction:

$$\varphi(t) = f(x + td)$$

le plus vite possible

On a  $\varphi'(0) = \nabla f(x)^\top d$ ,

et on cherche  $d$  solution du problème  $\min_{d \in \mathbb{R}^n, \|d\|=1} \varphi'(0)$ .

La solution est bien sûr  $d = -\frac{\nabla f(x)}{\|\nabla f(x)\|}$ ,

en vertu de l'inégalité de Schwartz.

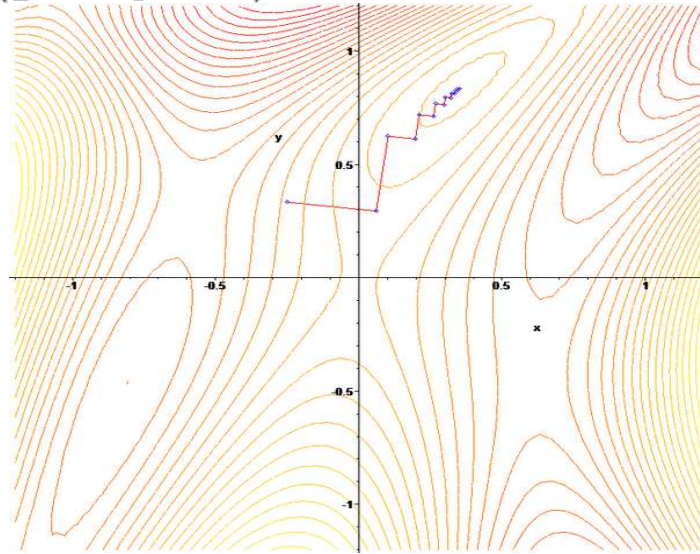
On obtient alors la méthode du gradient simple :

$$\begin{cases} d_k &= -\nabla f(x_k), \\ x_{k+1} &= x_k + \rho d_k. \end{cases}$$

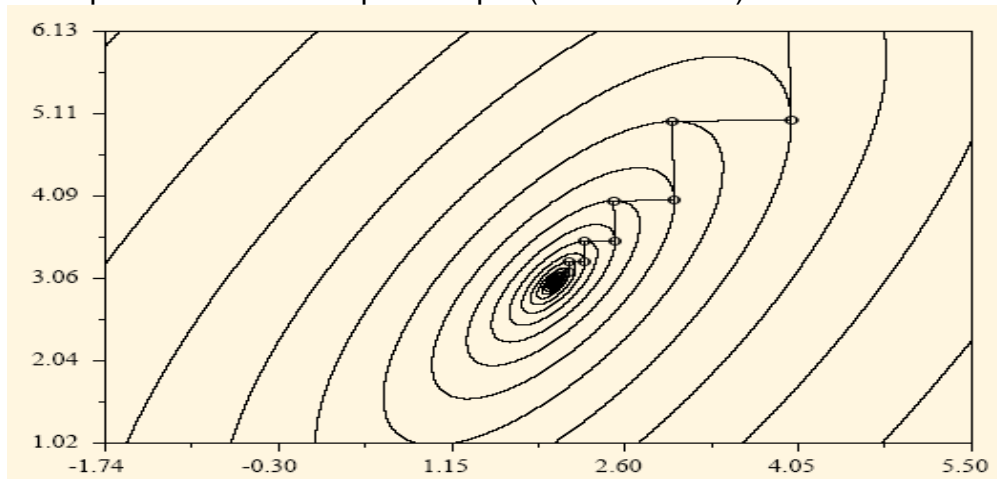


**Exemple :**

$$F(x, y) = \sin\left(\frac{1}{2}x^2 - \frac{1}{4}y^2 + 3\right) \cos(2x + 1 - e^y)$$



Exemple d'une fonction quadratique (donc convexe) :



## 7. La méthode de Newton

La méthode de Newton permet de construire un algorithme permettant de résoudre le système d'équations non-linéaires  $g(x) = 0$ , où  $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$  est différentiable : on se donne  $x_0 \in \mathbb{R}^n$  et on fait les itérations  $x_{k+1} = x_k - g'(x_k)^{-1}g(x_k)$ , où  $g'(x)$  est la dérivée (ou jacobienne) de  $g$  au point  $x$ .

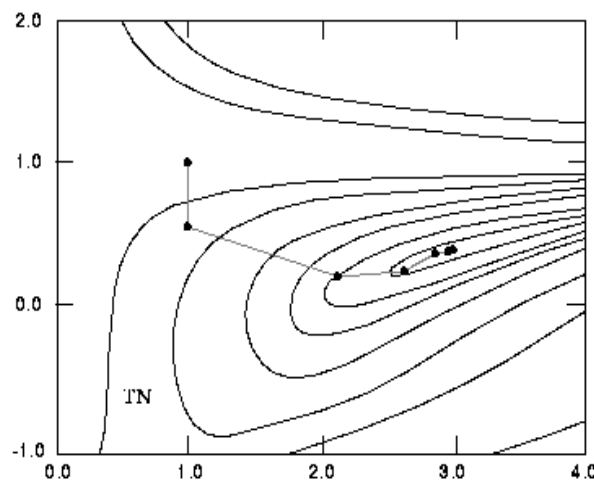
L'application de cette méthode au problème d'optimisation  $\min_{x \in \mathbb{R}^n} f(x)$ , consiste à poser  $g(x) = \nabla f(x)$  on obtient les itérations

$$x_{k+1} = x_k - \nabla^2 f(x_k)^{-1} \nabla f(x_k).$$

$d_k = -\nabla^2 f(x_k)^{-1} \nabla f(x_k)$  est une direction de descente en  $x_k$  si

$$\nabla f(x_k)^\top d_k = -\nabla f(x_k)^\top \nabla^2 f(x_k)^{-1} \nabla f(x_k) < 0,$$

ce qui sera le cas si  $\nabla^2 f(x_k)$  est une matrice définie positive,



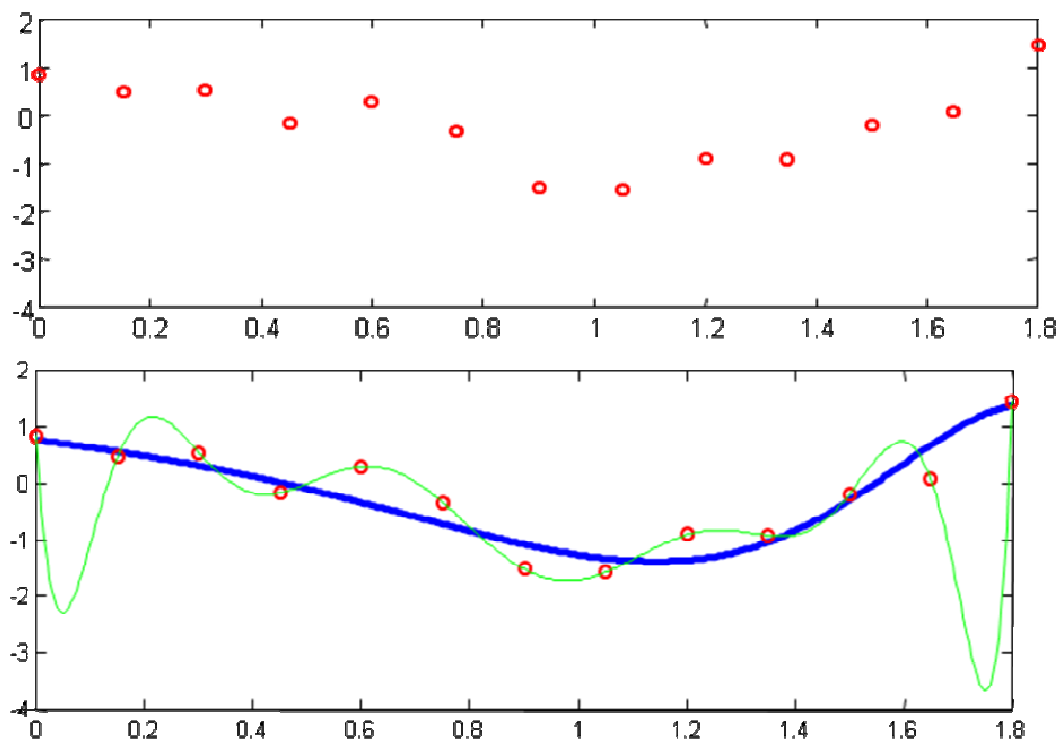
## **Chapitre 6 : Interpolation et Intégration numérique**

### **1. Introduction à l' interpolation numérique**

En analyse numérique, l'interpolation est une opération mathématique permettant de construire une courbe à partir de la donnée d'un nombre fini de points, ou une fonction à partir de la donnée d'un nombre fini de valeurs.

La solution du problème d'interpolation passe par les points prescrits, et, suivant le type d'interpolation, il lui est demandé de vérifier des propriétés supplémentaires.

Ainsi le type le plus simple d'interpolation est l'interpolation linéaire, qui consiste à "joindre les points" donnés.



L'interpolation doit être distinguée de l'approximation de fonction, qui consiste à chercher la fonction la plus proche possible, selon certains critères, d'une fonction donnée. Dans le cas de l'approximation, il n'est en général plus imposé de passer exactement par les points donnés initialement. Ceci permet de mieux prendre en compte le cas des erreurs de mesure, et c'est ainsi que l'exploitation de données expérimentales pour la recherche de lois empiriques relève plus souvent de la régression linéaire, ou plus généralement de la méthode des moindres carrés.

### **2. Interpolation polynomiale**

l'interpolation polynomiale est une technique d'interpolation d'un ensemble de données ou d'une fonction par un polynôme. En d'autres termes, étant donné un ensemble de points (obtenu, par exemple, à la suite d'une expérience), on cherche

un polynôme qui passe par tous ces points, et éventuellement vérifie d'autres conditions, de degré si possible le plus bas.

**Théorème (Polynôme d'interpolation)** Soient les  $n + 1$  points distincts

$$(a_0 < a_1 < \dots, a_n) \in \mathbb{R}^{n+1},$$

il existe un unique polynôme  $P$  de degré  $n$  qui coupe la fonction  $f$  sur ces points i.e. tel que :

$$\forall i \in \{0, \dots, n\}$$

$$P(a_i) = f(a_i)$$

C'est le polynôme interpolateur de LAGRANGE de  $f$  sur les points  $a_i$ .

Soient les  $n + 1$  points distincts  $(a_0 < a_1 < \dots, a_n) \in \mathbb{R}^{n+1}$ , on appelle **base de LAGRANGE** les polynômes de la forme :

$$\mathcal{L}_i(x) = \prod_{k=0, k \neq i}^n \frac{x - a_k}{a_i - a_k}$$

$$\forall i, j \quad \mathcal{L}_i(a_j) = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si } i \neq j \end{cases}$$

Le polynôme :

$$P(x) = f_0 \mathcal{L}_0(x) + f_1 \mathcal{L}_1(x) + \dots + f_n \mathcal{L}_n(x)$$

Convient. On l'appelle **polynôme d'interpolation de LAGRANGE**

### Exemple avec n=2

On connaît 3 points (0,1), (2,5) et (4,17)

Polynômes de Lagrange associés :

$$L_0(x) = \frac{(x-2)(x-4)}{8}$$

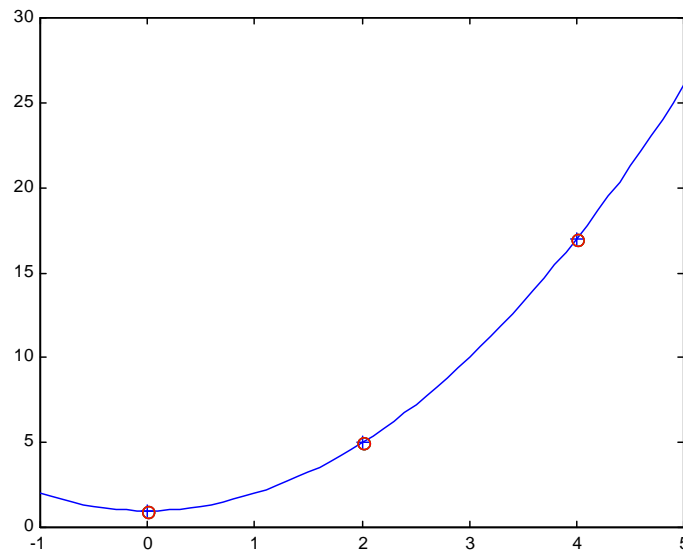
$$L_1(x) = \frac{x(x-4)}{-4}$$

$$L_2(x) = \frac{x(x-2)}{8}$$

Calcul du polynôme d'interpolation :

$$p(x) = L_0(x) + 5 L_1(x) + 17 L_2(x)$$

En simplifiant, on trouve  $p(x)=x^2+1$



### 3. Calcul de l'erreur d'interpolation

Si on interpole la fonction  $f \in C^{m+1}$  sur l'intervalle  $[a, b]$ , par le polynôme  $P(x)$  de degré  $n$ , grâce aux points d'interpolation  $a_0 < a_1 < \dots < a_n$ .

**Théorème**  $\forall x \in [a, b]$  il existe  $\eta \in [a_0, a_n]$  tel que

$$f(x) - P(x) = \frac{1}{(n+1)!} f^{(n+1)}(\eta) \phi(x)$$

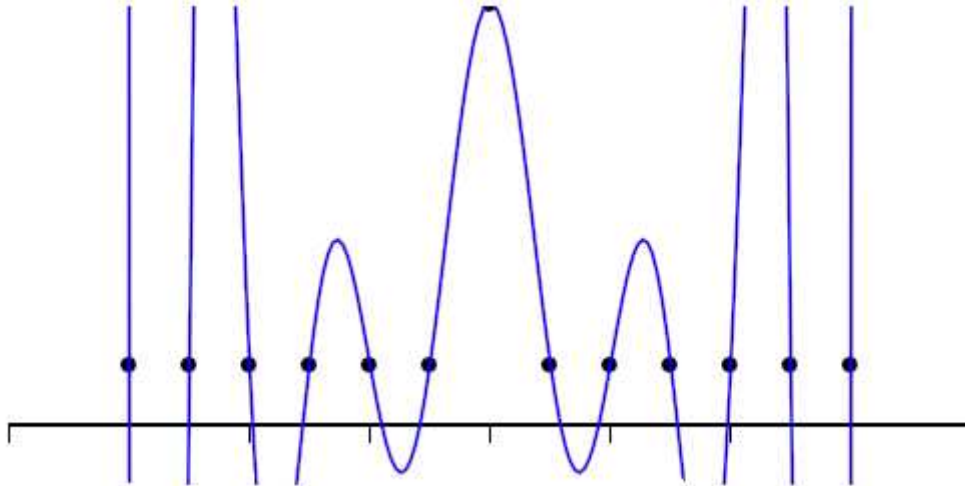
$$\text{avec} \quad \phi(x) = (x - a_0)(x - a_1) \cdots (x - a_n)$$

L'erreur dépend de :

- $\frac{1}{(n+1)!}$  qui tend vers 0 si  $n \rightarrow \infty$ .
- $\phi(x)$  qui tend vers  $\infty$  quand  $x \rightarrow \infty$ .  
 $\Rightarrow$  problèmes quand  $x \rightarrow \infty$
- $f^{(n+1)}(\eta)$  qui dépend de la fonction  $f$  et de l'intervalle  $[a_0, a_n]$ .  
 $\Rightarrow$  problèmes si un point est très différent des autres  
( $f^{(n+1)} \gg 1$ )  
 $\Rightarrow$  le polynôme a tendance à osciller entre les points d'interpolations

Certaines fonctions simples seront mal interpolées par un polynôme.

## Exemple :



Comment améliorer l'interpolation ?:

- On découpe l'intervalle en petits morceaux,
- On utilise des polynômes de petit degré pour approcher la fonction sur chaque sous-intervalle.

C'est ce qu'on appelle Interpolation polynomiale par morceau (splines )

Par exemple :

- Fonctions linéaires par morceaux
- Fonctions quadratiques par morceaux
- Les *splines cubiques* (polynôme de degré 3 par morceau).

## 4. Principe des méthodes d'intégration numérique

On cherche à approcher l'intégrale d'une fonction dont on ne connaît la valeur qu'en certains points mesures mais dont on ne peut pas calculer la primitive par une formule analytique.

Il faut donc approcher par des méthodes numériques l'intégrale :

$$I(f) = \int_a^b f(t)dt$$

On dispose de la valeur de  $f$  sur des points régulièrement espacés

$$f(y_0), f(y_1), \dots, f(y_m) \quad \text{avec} \quad y_0 = a < y_1 < \dots < y_{m-1} < y_m = b$$
$$\text{et} \quad y_{i+1} - y_i = \frac{b - a}{m}$$

- On sépare  $[a, b]$  en sous-intervalles  
i.e. on regroupe les points  $y_i$  par paquets de un, deux ( $[y_i, y_{i+1}]$ ) ou trois ( $[y_i, y_{i+1}, y_{i+2}]$ ) points consécutifs.
- On interpole la fonction sur chaque sous-intervalle par des polynômes  $g_i(t)$ .
- On calcule l'intégrale du polynôme d'interpolation de chaque sous intervalle, cela s'exprime simplement en fonction des valeurs  $f_i = f(y_i)$  :

$$\int_{y_i}^{y_{i+k}} g_i(t) dt = \sum_{l=0}^k \alpha_l f_{i+l}$$

La somme de ces valeurs est une approximation de l'intégrale de  $f$  sur  $[a, b]$ . Cette somme s'exprime aussi simplement en fonction des valeurs  $f_i$

$$I(f) \simeq \sum_{i=0}^m \beta_i f_i \quad (1)$$

La différence entre les méthodes vient du nombre de points d'interpolations dans les paquets :

- 1 point  $\Rightarrow$  approximation de degré 0  $\Rightarrow$  méthode des rectangles
- 2 points  $\Rightarrow$  approximation de degré 1  $\Rightarrow$  méthode des trapèzes
- 3 points  $\Rightarrow$  approximation de degré 2  $\Rightarrow$  méthode de SIMPSON
- $n$  points  $\Rightarrow$  approximation de degré  $n - 1 \Rightarrow$  méthode de NEWTON-CÔTES

Pour chaque méthode, il existe des *constantes*  $\beta_i$  qui permettent d'appliquer la formule (1) et une majoration de l'erreur que l'on va calculer.

### 5. Méthode des rectangles



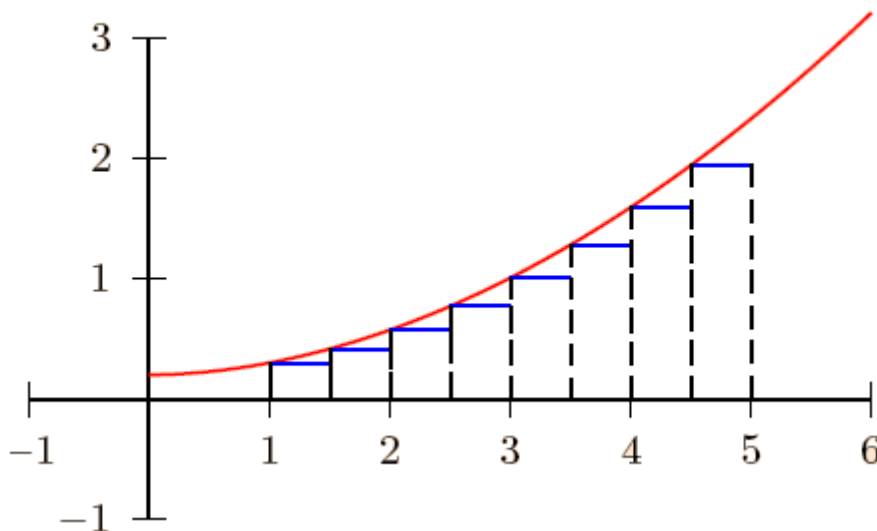
Les points  $(y_i) \ i = 0, \dots, m$  sont pris régulièrement espacés sur  $[a, b] : y_i = a + i \frac{b-a}{m}$ . Sur chaque intervalle  $I_i = [y_i, y_{i+1}]$ , la fonction  $f$  est approchée par la fonction constante  $g_i$  tel que  $g_i(t) = f(y_i)$ .

$$\begin{aligned} \int_{y_i}^{y_{i+1}} g_i(t) dt &= (y_{i+1} - y_i) f(y_i) = \frac{b-a}{m} f(y_i) \\ \int_a^b f(t) dt &= \int_{y_0}^{y_1} f(t) dt + \int_{y_1}^{y_2} f(t) dt + \dots + \int_{y_{m-1}}^{y_m} f(t) dt \end{aligned}$$

L'approximation de  $\int_a^b f(t) dt$  par la méthode des rectangles est donnée par :

$$I_R = \frac{b-a}{m} \sum_{i=0}^{m-1} f(y_i)$$

**Exemple :**



### Erreur commise

On applique la formule d'erreur de l'interpolation de Lagrange on obtient alors :

$$\left| \int_a^b f(t) dt - I_R \right| \leq M \frac{(b-a)^2}{2m}$$

$$M = \sup_{t \in [a,b]} |f'(t)|$$

Avec :



## 6. Méthode des trapèzes

Les points  $(y_i)$   $i = 0, \dots, m$  sont pris régulièrement espacés sur  $[a, b]$  :  $y_i = a + i \frac{b-a}{m}$ .

Sur chaque intervalle  $I_i = [y_i, y_{i+1}]$ , la fonction  $f$  est approchée par la fonction affine  $g_i$  coïncidant avec  $f$  en  $y_i$  et  $y_{i+1}$  soit :

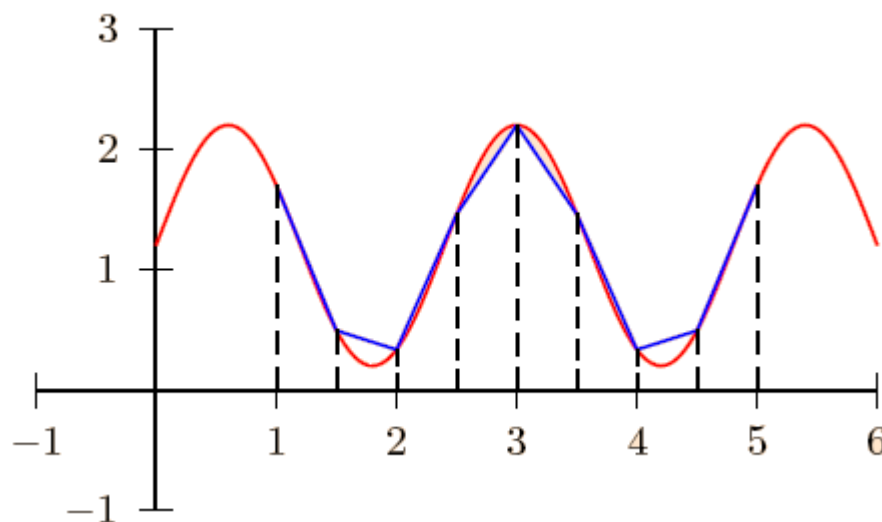
$$g_i(t) = f(y_i) + \frac{(t - y_i)}{(y_{i+1} - y_i)}(f(y_{i+1}) - f(y_i)).$$

Remarque :  $g_i$  est la fonction affine par morceaux reliant les points de coordonnées  $(y_i, f(y_i))$ .

L'approximation de  $\int_a^b f(t)dt$  par la méthode des trapèzes est donnée par :

$$I_T = \frac{b-a}{2m} (f(y_0) + 2f(y_1) + \dots + 2f(y_{m-1}) + f(y_m))$$

**Exemple :**



### Erreur commise

On applique la formule d'erreur de l'interpolation de Lagrange on obtient alors :

$$\left| \int_a^b f(t)dt - I_T \right| \leq M \frac{(b-a)^3}{12m^2}$$

$$M = \sup_{t \in [a,b]} |f''(t)|$$

Avec :

Remarques :

- Pour trouver une valeur approchée de  $\int_a^b f(t)dt$  à  $\varepsilon$  près, il suffit de prendre  $m$  plus grand que  $\sqrt{M \frac{(b-a)^3}{12\varepsilon}}$
- L'approximation est exacte si la dérivée seconde  $f''$  est nulle c'est-à-dire si la fonction  $f$  est affine.

## 7. Méthode de SIMPSON

On considère les  $2n + 1$  points  $z_i = a + i \frac{b-a}{2n}$  ( $i = 0, \dots, 2n$ )

Sur chaque intervalle  $I_i = [z_{2i}, z_{2i+2}]$ , la fonction  $f$  est approchée par la parabole  $g_i$  passant par les points

$$(z_{2i}, f(z_{2i})) \quad (z_{2i+1}, f(z_{2i+1})) \quad (z_{2i+2}, f(z_{2i+2}))$$

Donc

$$\begin{aligned} g_i(t) = & f(z_{2i}) \cdot \frac{(t - z_{2i+1})(t - z_{2i+2})}{(z_{2i} - z_{2i+1})(z_{2i} - z_{2i+2})} \\ & + f(z_{2i+1}) \cdot \frac{(t - z_{2i})(t - z_{2i+2})}{(z_{2i+1} - z_{2i})(z_{2i+1} - z_{2i+2})} \\ & + f(z_{2i+2}) \cdot \frac{(t - z_{2i})(t - z_{2i+1})}{(z_{2i+2} - z_{2i})(z_{2i+2} - z_{2i+1})} \end{aligned}$$

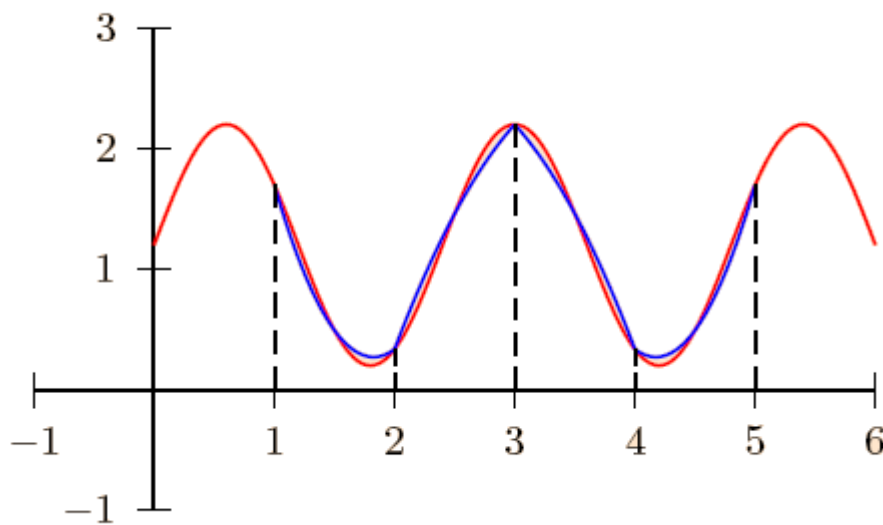
Ce qui donne, tout calcul fait :

$$\int_{z_{2i}}^{z_{2i+2}} g_i(t)dt = \frac{b-a}{6n} (f(z_{2i}) + 4f(z_{2i+1}) + f(z_{2i+2}))$$

L'approximation de l'intégrale par la méthode de SIMPSON est donc  $I_S$  avec

$$\begin{aligned} I_S = & \frac{b-a}{6n} \left( f(z_0) + 4f(z_1) + 2f(z_2) + 4f(z_3) + 2f(z_4) + \right. \\ & \left. \dots + 2f(z_{2n-2}) + 4f(z_{2n-1}) + f(z_{2n}) \right) \end{aligned}$$

**Exemple :**



## Erreur commise

Soit  $M = \max_{t \in [a,b]} |f^{(4)}(t)|$

$$\left| \int_a^b f(t)dt - I_S \right| \leq \frac{M(b-a)^5}{2880n^4}$$

Remarques :

- Pour trouver une valeur approchée de  $\int_a^b f(t)dt$  à  $\varepsilon$  près, il suffit de prendre  $m$  plus grand que  $4\sqrt[4]{M \frac{(b-a)^5}{2880\varepsilon}}$
- L'approximation est exacte si la dérivée  $f^{(4)}$  est nulle c'est-à-dire si la fonction  $f$  est un polynôme de degré inférieur ou égal à 3.

## Chapitre 7 : Méthodes numériques pour les équations différentielles

### 1. Problème de Cauchy

#### Définition

Soit  $f : [a,b] \times \mathbb{R} \rightarrow \mathbb{R}$  et  $y_0 \in \mathbb{R}$ . Le problème de Cauchy ou le problème de la condition initiale est le suivant :

Trouver une fonction  $y : [a,b] \rightarrow \mathbb{R}$  dérivable sur  $[a,b]$  qui vérifie :

$$\begin{cases} y(a) = y_0 \\ y'(x) = f(x, y(x)) \quad \forall x \in [a,b] \end{cases}$$

#### Théorème (Cauchy-Lipshitz)

On suppose que la fonction  $f$  vérifie les conditions suivantes :

- $f$  est continue par rapport à ses deux variables.
- $f(x,y)$  est Lipchitzienne en  $y$  uniformément par rapport à  $x$ , c'est-à-dire :  
 $\exists L > 0$  telle que :  $|f(x, y) - f(x, z)| \leq L|y - z|$  pour tout  $x \in [a,b]$  et  $\forall y, z \in \mathbb{R}$

Alors dans ce cas le problème de Cauchy admet une solution unique■.

### 2. Méthode d'Euler

La méthode d'Euler est la méthode numérique la plus simple mais pas la plus efficace qui est utilisée pour résoudre une équation différentielle.

On considère le problème de Cauchy suivant : 
$$\begin{cases} y(a) = y_0 \\ y'(x) = f(x, y(x)) \quad \forall x \in [a,b] \end{cases}$$

La méthode d'Euler consiste à subdiviser l'intervalle  $[a,b]$  en points équidistants :

$x_0=a, x_1, \dots, x_n=b$  avec  $x_{i+1}-x_i=(b-a)/n=h$  est le pas de la subdivision.

On en déduit  $x_i=a+ih$  pour  $i=0,1,\dots,n$ .

Supposons que la solution du problème soit deux fois dérivable alors d'après le développement de Taylor et pour chaque indice  $i=0\dots n-1$ , il existe  $c_i \in [x_i, x_{i+1}]$  tel que :

$$y(x_{i+1}) = y(x_i) + (x_{i+1} - x_i)y'(x_i) + \frac{(x_{i+1} - x_i)^2}{2} y''(c_i)$$

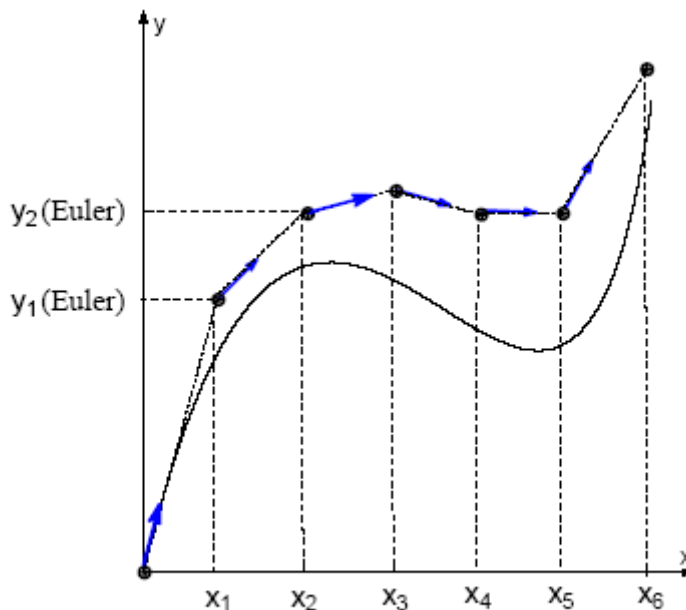
Puisque  $y(x)$  satisfait l'équation différentielle on peut écrire :

$$y(x_{i+1}) = y(x_i) + hf(x_i, y(x_i)) + \frac{h^2}{2} y''(c_i)$$

La méthode d'Euler consiste à négliger pour chaque indice  $i$  le terme  $\frac{h^2}{2} y''(c_i)$

On obtient ainsi une estimation de la solution sur les points de la subdivision :

$$\begin{cases} y_0 = y(a) \\ y_{i+1} = y_i + hf(x_i, y_i) \quad \text{pour } i = 0 \dots n-1 \end{cases}$$



La courbe en trait plein correspond à la solution analytique.

## Théorème

On peut montrer que suivant certaines conditions ( $f$  lipchitzienne et  $y''$  bornée sur  $[a,b]$ ) l'approximation  $(y_0, y_1, \dots, y_n)$  donné par la méthode d'Euler converge vers la solution exacte  $(y(x_0), y(x_1), \dots, y(x_n))$  lorsqu'on augmente le nombre des points de la subdivision c'est-à-dire lorsque  $h$  tends vers 0■.

## Définition

Une méthode numérique fournissant des valeurs approchées  $(y_0, y_1, \dots, y_n)$  de la solution exacte  $(y(x_0), y(x_1), \dots, y(x_n))$  est dite d'ordre  $p$  s'il existe une constante  $K$  telle que :

$$\max_{i=0, \dots, n} |y(x_i) - y_i| \leq K.h^p$$

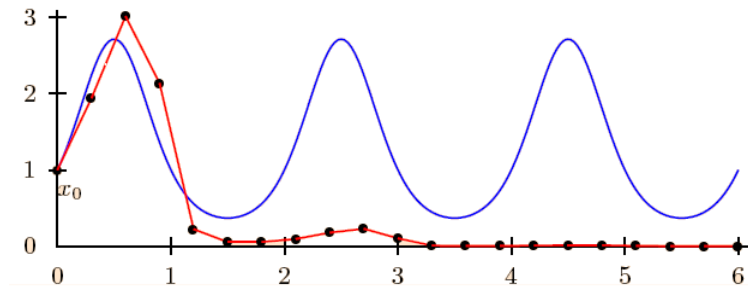
On peut vérifier que la méthode d'Euler est d'ordre 1 donc c'est une méthode qui ne converge pas rapidement. En général on utilise des méthodes d'ordre plus élevé comme les méthodes de Runge-Kutta qui peuvent atteindre l'ordre 4.

## Exemple

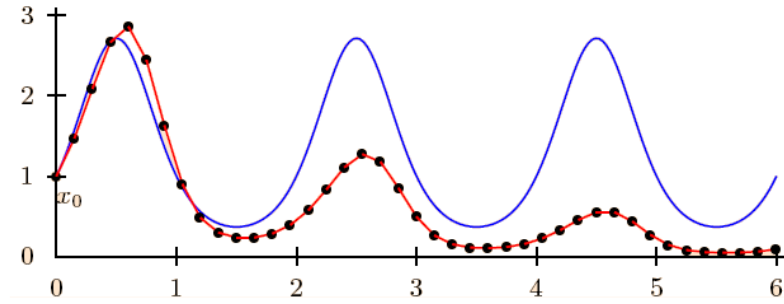
$$\begin{cases} y(0) = 0 \\ y' = \pi \cos(\pi x) y(x) \end{cases}$$

Solution exacte :  $y = e^{\sin(\pi x)}$

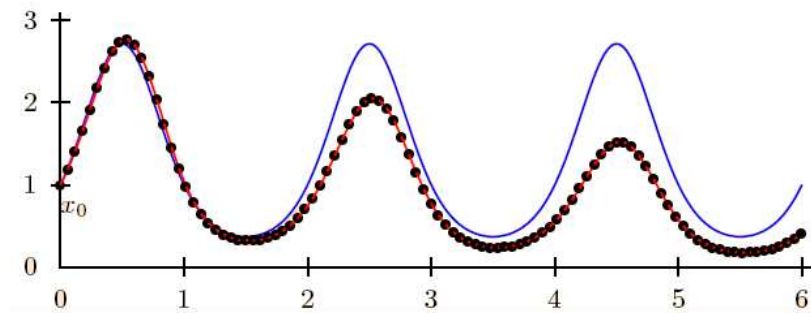
Solution avec la méthode d'Euler avec un pas de 0.3



Solution avec la méthode d'Euler avec un pas de 0.15 :



Solution avec la méthode d'Euler avec un pas de 0.06 :



### 3. Méthodes de RUNGE-KUTTA

Le principe des méthodes à pas séparés est de calculer

$$y_{k+1} = y_k + \varphi(x_k, y_k, h)$$

en choisissant  $\varphi$  de manière à ce que  $y_k$  soit le plus proche possible de  $y(x_k)$ .

Comme  $x_{k+1} = x_k + h$  une bonne méthode est de choisir  $\varphi$  afin d'annuler le plus de termes possibles dans le développement de TAYLOR de  $y(x + h)$

$$y(x + h) = y(x) + hy'(x) + \frac{h^2}{2}y''(x) + \dots + \frac{h^n}{n!}y^{(n)}(x) + O(h^{n+1})$$

C'est possible puisque

$$y'(x) = f(x, y(x))$$

$$y''(x) = \frac{\partial f}{\partial x}(x, y(x)) + \frac{\partial f}{\partial y}(x, y(x))f(x, y(x))$$

...

On peut par exemple choisir :

$$y_{n+1} = y_n + hf(x_n, y_n) + \frac{h^2}{2} \left( \frac{\partial f}{\partial x}(x_n, y_n) + \frac{\partial f}{\partial y}(x_n, y_n)f(x_n, y_n) \right)$$

Qui est d'ordre 2.

Mais cela oblige à calculer  $\frac{\partial f}{\partial x}$  et  $\frac{\partial f}{\partial y}$ .

Pour éviter de calculer ces dérivées partielles, on peut utiliser une approximation :

$$\frac{\partial f}{\partial x}(x_n, y_n) \simeq \frac{f(x_n + h, y_n) - f(x_n, y_n)}{h}$$

Au final,  $\varphi$  sera de la forme :

$$\varphi(x, y, h) = a_1 f(x, y) + a_2 f(x + a_3 h, y + a_4 h)$$

L'idée de RUNGE-KUTTA est d'utiliser le développement de TAYLOR de  $f(x + a_3h, y(x) + a_4h)$  pour identifier les valeurs des coefficients  $a_1, a_2, a_3$  et  $a_4$ .

$$f(x + a_3h, y(x) + a_4h) = f(x, y(x)) + a_3h \frac{\partial f}{\partial x}(x, y(x)) + a_4h \frac{\partial f}{\partial y}(x, y(x)) + O(h^2)$$

en remplaçant  $x$  par  $x_n$  et  $y(x)$  par  $y(x_n)$  cela donne

$$y_{n+1} = y_n + (a_1 + a_2)hf(x_n, y_n) + a_2a_3h^2 \frac{\partial f}{\partial x}(x_n, y_n) + a_2a_4h^2 \frac{\partial f}{\partial y}(x_n, y_n)$$

Qu'il faut comparer à

$$y_{n+1} = y_n + hf(x_n, y_n) + \frac{h^2}{2} \frac{\partial f}{\partial x}(x_n, y_n) + \frac{f(x_n, y_n)h^2}{2} \frac{\partial f}{\partial y}(x_n, y_n)$$

En identifiant terme à terme les deux itérations :

$$\begin{cases} a_1 + a_2 = 1 \\ a_2a_3 = \frac{1}{2} \\ a_2a_4 = \frac{f(x_n, y_n)}{2} \end{cases}$$

on peut choisir :

○  $a_1 = a_2 = \frac{1}{2}, a_3 = 1$  et  $a_4 = f(x_n, y_n)$ .

C'est la méthode d'EULER modifiée

$$\varphi(x, y, h) = \frac{f(x, y)}{2} + \frac{1}{2}f(x + h, y + hf(x_n, y_n))$$

○  $a_1 = 0, a_2 = 1, a_3 = \frac{1}{2}$  et  $a_4 = \frac{f(x_n, y_n)}{2}$ .

C'est la méthode de HEUN

$$\varphi(x, y, h) = f\left(x + \frac{h}{2}, y + \frac{h}{2}f(x_n, y_n)\right)$$

Par définition ces méthodes sont d'ordre 2



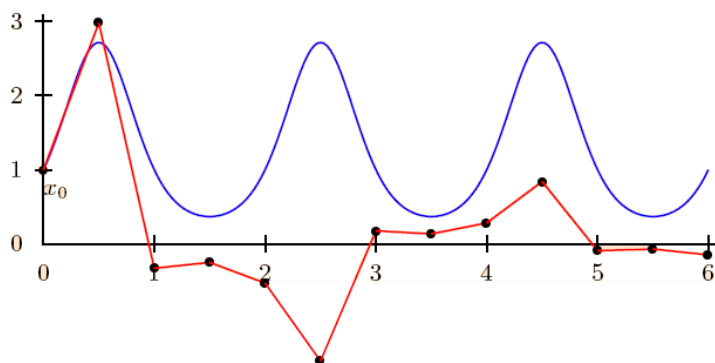
En poursuivant le même raisonnement à partir du développement de TAYLOR, d'ordre 5 on obtient la méthode de RUNGE-KUTTA d'ordre 4 :

$$\begin{cases} x_{n+1} = x_n + h \\ k_1 = hf(x_n, y_n) \\ k_2 = hf(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}) \\ k_3 = hf(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}) \\ k_4 = hf(x_n + h, y_n + k_3) \\ y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \end{cases}$$

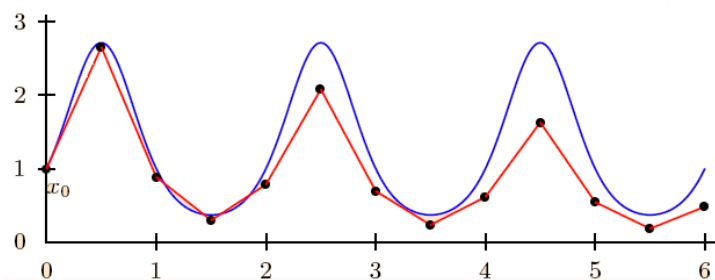
## Exemple

$$\begin{cases} y(0) = 0 \\ y' = \pi \cos(\pi x) y(x) \end{cases}$$

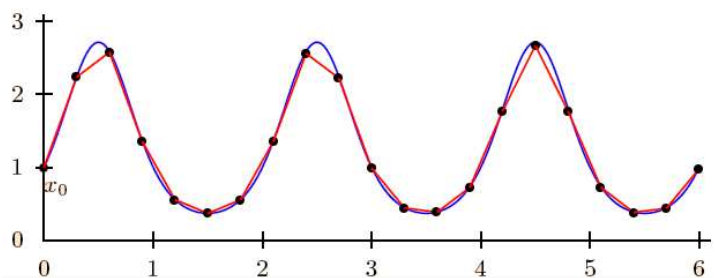
Solution exacte :  $y = e^{\sin(\pi x)}$



Solution avec la méthode de Heun avec un pas de 0.5



Solution avec la méthode de Runge-Kutta avec un pas de 0.5



Solution avec la méthode de Runge-Kutta avec un pas de 0.3

## **Chapitre 8 :Introduction aux différences finies et éléments finis**

### **1. Modélisation mathématique et Simulation numérique**

Un modèle mathématique et une interprétation abstraite de la réalité physique qui est accessible à l'analyse et au calcul.

La simulation numérique permet de calculer sur ordinateur les solutions de ces modèles.

Les modèles étudiées ici sont des équations différentielles à plusieurs variables (EDP) et qui sont déterministes.

Les domaines d'applications de la modélisation et la simulation numérique sont innombrables Quelques exemples:

- Sciences de l'ingénieur: aérodynamique, calcul des structures, électromagnétisme, énergie, automatique, signal...
- Autres sciences: physique, optique, chimie, biologie, économie...
- Météorologie, environnement, finance...

Le Principe des méthodes de résolution numérique des EDP consiste à « Obtenir des valeurs numériques discrètes (c.a.d en nombre fini) qui « approchent » en un sens convenable la solution exacte »

Donc dans ce cas on a les propriétés suivantes :

- On calcule des solutions approchées
- On discrétise le problème ( On passe du continue au discret): remplacement des fonction par un nombre fini de valeurs.

Il y a plusieurs méthodes de résolution numériques mais les deux principales méthodes sont :

- Méthode des Différences finies
- Méthode des Eléments finis

### **2. Quelques modèles classiques**

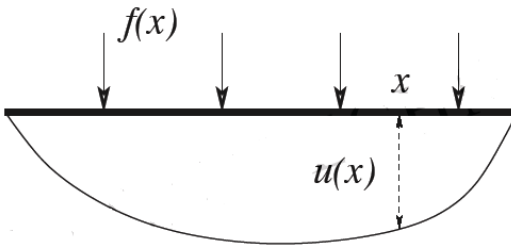
#### **Equation de la chaleur:**

Elle intervient, comme modèle, dans de nombreux problèmes des sciences de l'ingénieur :

$$\begin{cases} \frac{\partial u}{\partial t} - \Delta u = f & \text{dans } \Omega \times \mathbb{R}_+^+ \\ u = 0 & \text{sur } \partial\Omega \times \mathbb{R}_+^+ \\ u(t = 0) = u_0 & \text{dans } \Omega \end{cases}$$

Equation d'ordre 1 en temps 2 en espace (équation parabolique)

## Equation des ondes:

$$\left\{ \begin{array}{ll} \frac{\partial^2 u}{\partial t^2} - \Delta u = f & \text{dans } \Omega \times \mathbb{R}_*^+ \\ u = 0 & \text{sur } \partial\Omega \times \mathbb{R}_*^+ \\ u(t=0) = u_0 & \text{dans } \Omega \\ \frac{\partial u}{\partial t}(t=0) = u_1 & \text{dans } \Omega \end{array} \right. \quad \Omega$$


Modélisation des phénomènes de propagation des ondes ou de vibration (vibration d'une membrane élastique, corde vibrante ...)

Domaine  $\Omega$ , déplacement normale  $u$ . force normale  $f$ , condition aux limites de Dirichlet

Ordre 2 en temps donc il faut 2 conditions initiales pour  $u$  (équation hyperboliques)

## Equation de Laplace:

$$\left\{ \begin{array}{ll} -\Delta u = f & \text{dans } \Omega \\ u = 0 & \text{sur } \partial\Omega, \end{array} \right.$$

Pour certains choix de  $f$ : la solution de l'équation de la chaleur admet une limite quand le temps tend vers  $+\infty$ . On dit que la solution atteint un état stationnaire.

Cette équation est elliptique. C'est aussi la version stationnaire de l'équation des ondes.

Elle intervient dans beaucoup de problèmes par exemple: Modéliser le déplacement verticale  $u$  d'une membrane élastique fixée sur son contour et soumise à une force normale  $f$ .

## 3. Classification des équations aux dérivées partielles (EDP)

On appelle ordre d'une EDP, l'ordre de la plus grande dérivées présente dans l'équation

### Exemples:

- Le Laplacien est une équation de second ordre
- L'équation de chaleur est du premier ordre en temps et du second ordre en espace.
- L'équation des ondes est du second ordre en espace-temps.

On utilise une classification des EDP de second ordre à coefficient constants portant sur une variable  $u(x,y)$  de la manière suivante:

$$a \frac{\partial^2 u}{\partial x^2} + b \frac{\partial^2 u}{\partial x \partial y} + c \frac{\partial^2 u}{\partial y^2} + d \frac{\partial u}{\partial x} + e \frac{\partial u}{\partial y} + f u = g.$$

- On dit que l'équation est Elliptique si :  $b^2 - 4ac < 0,$

- On dit que l'équation est Parabolique si :  $b^2 - 4ac = 0$
- On dit que l'équation est Hyperbolique si :  $b^2 - 4ac > 0$

L'origine de cette classification vient de celle des coniques du plan.

En règle générale : les problèmes stationnaires (indépendants du temps) sont modélisés par des e.d.p. elliptiques tandis que les problèmes d'évolution sont modélisés par des e.d.p. paraboliques ou hyperboliques

## Exemples

⇒ Exemple d'équation **parabolique**: équation de la chaleur

$$\begin{cases} \frac{\partial \theta}{\partial t} - \Delta \theta = f & \text{dans } \Omega \times \mathbb{R}_*^+ \\ + \text{conditions aux limites} + \text{condition initiale} \end{cases}$$

⇒ Exemple d'équation **elliptique**: équation de Laplace

$$\begin{cases} -\Delta \theta = f & \text{dans } \Omega \\ + \text{conditions aux limites} \end{cases}$$

⇒ Exemple d'équation **hyperbolique**: équation des ondes

$$\begin{cases} \frac{\partial^2 \theta}{\partial t^2} - \Delta \theta = f & \text{dans } \Omega \times \mathbb{R}_*^+ \\ + \text{conditions aux limites} + \text{conditions initiales} \end{cases}$$

---

## 4. Méthode de différences finies

La méthode des différences finies est l'une des plus anciennes méthodes de simulation numérique. Elle est encore utilisée pour certaines applications comme la:

- Propagation des ondes (sismiques ou électromagnétiques)
- Mécanique des fluides compressibles

Pour d'autres applications on lui préfère la méthode des éléments finis:

- Mécanique des solides
- Mécanique des fluides incompressibles

Principe de la méthode des différences finies : Les fonctions et leurs dérivées qui interviennent dans les équations EDP sont remplacées par des approximations (des différences finies) :

$$u_j^n \approx u(t_n, x_j)$$

$$\frac{\partial u}{\partial x}(t_n, x_j) \approx \frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} \quad \text{ou bien} \quad \approx \frac{u_{j+1}^n - u_j^n}{\Delta x} \quad \text{ou bien} \quad \approx \frac{u_j^n - u_{j-1}^n}{\Delta x}$$

Ceci a pour effet de remplacer un problème de dimension infinie (calculer la fonction  $u(t, x)$ ) par un problème de dimension fini calculer le vecteur  $u_j^n$ . ceci permet à son tour de résoudre le problème par ordinateur.

Il y a plusieurs formule d'approximation par différences finies, en général on utilise les formules de Taylor pour les générer.

Par exemple de la formule de Taylor :

$$\begin{aligned} -u(t, x - \Delta x) + 2u(t, x) - u(t, x + \Delta x) = & -(\Delta x)^2 \frac{\partial^2 u}{\partial x^2}(t, x) \\ & - \frac{(\Delta x)^4}{12} \frac{\partial^4 u}{\partial x^4}(t, x) + \mathcal{O}((\Delta x)^6) \end{aligned}$$

On déduit une formule de différences finies pour la dérivée seconde  $u(t, x)$  :

$$-\frac{\partial^2 u}{\partial x^2}(t_n, x_j) \approx \frac{-u_{j-1}^n + 2u_j^n - u_{j+1}^n}{(\Delta x)^2}$$

Pour la dérivée première de  $u(t, x)$  on a trois possibilités :

⇒ Différence finie centrée en temps:

$$\frac{\partial u}{\partial t}(t_n, x_j) \approx \frac{u_j^{n+1} - u_j^{n-1}}{2\Delta t}$$

⇒ Différence finie décentrée (on avance dans le temps): **Euler explicite**

$$\frac{\partial u}{\partial t}(t_n, x_j) \approx \frac{u_j^{n+1} - u_j^n}{\Delta t}$$

⇒ Différence finie décentrée (on remonte dans le temps): **Euler implicite**

$$\frac{\partial u}{\partial t}(t_n, x_j) \approx \frac{u_j^n - u_j^{n-1}}{\Delta t}$$

## Application à l'équation de la chaleur

$$\begin{cases} \frac{\partial u}{\partial t} - \nu \Delta u = 0 & \text{dans } \Omega \times \mathbb{R}_*^+ \\ u = 0 & \text{sur } \partial\Omega \times \mathbb{R}_*^+ \\ u(t=0, x) = u_0(x) & \text{dans } \Omega \end{cases}$$

⇒ schéma centré: le plus "naturel"

$$\frac{u_j^{n+1} - u_j^{n-1}}{2\Delta t} + \nu \frac{-u_{j-1}^n + 2u_j^n - u_{j+1}^n}{(\Delta x)^2} = 0$$

⇒ schéma d'Euler explicite: le plus simple

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + \nu \frac{-u_{j-1}^n + 2u_j^n - u_{j+1}^n}{(\Delta x)^2} = 0$$

(explicite ⇔ formule immédiate pour trouver  $u^{n+1}$  en fonction de  $u^n$ )

⇒ schéma d'Euler implicite: plus compliqué

$$\frac{u_j^n - u_j^{n-1}}{\Delta t} + \nu \frac{-u_{j-1}^n + 2u_j^n - u_{j+1}^n}{(\Delta x)^2} = 0$$

(implicite ⇔ système linéaire pour trouver  $u^n$  en fonction de  $u^{n-1}$ )

Initialisation:  $u_j^0 = u_0(x_j)$  où  $u_0(x)$  est la condition initiale.

## 5. Méthode des éléments finis

La méthode des éléments finis est une méthode numérique pour résoudre des équations différentielles linéaires en utilisant l'approche variationnelle des équations d'état. L'idée de la méthode est de remplacer l'espace des solutions, dans la formulation variationnelle, par un espace de dimension fini bien choisi.

Le problème approché se ramène alors à la résolution d'un système linéaire.

A noter la différence de philosophie entre les différences finies et les éléments finis :

- Différences finies : On approche les opérateurs qui interviennent dans les EDP.
- Éléments finis : On approche l'espace dans le quel on cherche la solution.

## Exemple de l'équation de Laplace :

**Etape 1.** On transforme les équations en les écrivant de manière **variationnelle**. Principe (formel) de la **formulation variationnelle** :

$$-\frac{\partial^2 u}{\partial x^2} = f \quad x \in (a, b)$$

$$\Rightarrow - \int_a^b \frac{\partial^2 u}{\partial x^2}(x) v(x) dx = \int_a^b f(x) v(x) dx \quad \forall v.$$

En intégrant par parties, et en utilisant  $u(a) = u(b) = 0$  ;

$$\int_a^b \frac{\partial u}{\partial x}(x) \frac{\partial v}{\partial x}(x) dx = \int_a^b f(x) v(x) dx \quad \forall v.$$

- Cette équation requiert moins de régularité pour  $u$ .
- Même régularité est requise pour  $u$  que pour la fonction test  $v$ .

**Etape 2.** Principe de discrétisation (approximation de type **Galerkin**)

$\Rightarrow$  On remplace dans la formulation variationnelle l'espace  $\mathcal{V}$  par un **sous espace** de **dimension finie** noté  $\mathcal{V}_h : \mathcal{V}_h \xrightarrow{h \rightarrow 0} \mathcal{V}$ .

$$\text{On note } N_h = \dim \mathcal{V}_h \implies \mathcal{V}_h \hookrightarrow \mathbb{R}^{N_h}$$

Problème discrétisé

$$\left\{ \begin{array}{l} \text{Trouver } u_h \in \mathcal{V}_h \text{ tq.} \\ \int_a^b \frac{\partial u_h}{\partial x}(x) \frac{\partial v_h}{\partial x}(x) dx = \int_a^b f(x) v_h(x) dx, \quad \forall v_h \in \mathcal{V}_h. \end{array} \right.$$

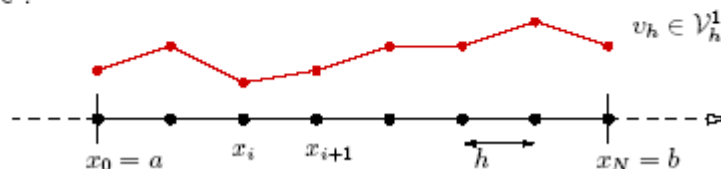
$$\iff \boxed{A_h U_h = F_h \text{ dans } \mathbb{R}^{N_h}}$$

Espaces d'éléments finis (de **Lagrange**) en 1-D

$$\mathcal{V}_h^k(a, b) = \{v_h \in C^0(a, b), \text{ tq. } v_h|_{[x_i, x_{i+1}]} \in P^k\}$$

$P^k$  : ensemble des polynômes de degrés  $\leq k$ .

Exemple :



## 6. Bibliographie

- [1] Analyse numérique pour ingénieurs ; Presses Internationales Polytechnique ; A. Fortin ; 2009
- [2] Introduction à l'analyse numérique matricielle et à l'optimisation. P.G Ciarlet, Dunod.
- [3] Analyse numérique des équations différentielles. M. Crouzeix, A. L. Mignot. Collection mathématiques appliquées pour la maîtrise. Masson, Paris 1984.
- [4] Calcul scientifique, Cours, exercices corrigés et illustrations en Matlab et Octave Alfio Quarteroni, austoSaleri, Springer.
- [5] Ciarlet, Miara et Thomas, Exercices d'analyse numérique matricielle et d'optimisation, Masson.
- [6] Lascaux-Théodor, Analyse numérique matricielle appliquée à l'art de l'ingénieur, Dunod.