ÉCOLE CENTRALE LYON

Analyse Numérique

Laurent Seppecher et Grégory Vial

Cours de tronc commun S5

Version 1.9 – 11 janvier 2023

Table des matières

Chapitre 1

Analyse numérique matricielle

1.1 Résolution numérique de systèmes linéaires

1.1.1 Normes subordonnées

Définition 1.1

Soit $\|\cdot\|$ une norme sur $\mathbb{R}^d$ et $A \in \mathbb{R}^{d\times d}$ (ou $\mathbb{C}^{d\times d}$) une matrice carrée. La norme subordonnée de A est définie comme

$$|||A||| = \sup_{x\in\mathbb{C}^d\setminus\{0\}} \frac{\|Ax\|}{\|x\|}.$$

Proposition 1.1

Soient $A, B \in \mathbb{R}^{d\times d}$ et $x \in \mathbb{C}^d$.

- $\|Ax\| \leqslant |||A||| \times \|x\|$.
- $|||I_d||| = 1$.
- $|||AB||| \leqslant |||A||| \times |||B|||$.

Définition 1.2 (*rayon spectral*)

Soit $A \in \mathbb{R}^{d\times d}$ (ou $\mathbb{C}^{d\times d}$). On appelle rayon spectral de A le nombre

$$\rho(A) = \max\Big\{|\lambda| \ ; \ \lambda \text{ valeur propre complexe de } M\Big\}.$$

Proposition 1.2

Soit $A \in \mathbb{R}^{d\times d}$. Alors, pour toute norme subordonnée, on a

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

On utilise fréquemment les normes suivantes sur $\mathbb{R}^d$ ($p \geqslant 1$) :

$$\|x\|_p = \left(\sum_{i=1}^{d} |x_i|^p\right)^{\frac{1}{p}}, \qquad \|x\|_\infty = \max_{i=1}^{d} |x_i|.$$

Les normes subordonnées associées sont notées $|||A|||_p$ et $|||A|||_\infty$. Les normes 1 et ∞ se calculent aisément en fonction des coefficients de la matrice, la norme 2 est reliée aux valeurs propres :

Proposition 1.3

$$|||A|||_\infty = \max_{i=1}^{d} \sum_{j=1}^{d} |A_{ij}|, \qquad |||A|||_1 = \max_{j=1}^{d} \sum_{i=1}^{d} |A_{ij}|.$$

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

En particulier, si A est symétrique, on a $|||A|||_2 = \rho(A)$.

1.1.2 Conditionnement d'un système linéaire

Définition 1.3

Soit $\|\cdot\|$ une norme sur $\mathbb{R}^d$ et $|||\cdot|||$ la norme subordonnée associée. On appelle conditionnement *d'une matrice $A \in \mathbb{R}^{d\times d}$ inversible le nombre*

$$\mathrm{cond}(A) = |||A||| \times |||A^{-1}|||.$$

Ce nombre dépend de la norme choisie. Pour les normes usuelles $\|\cdot\|_p$ et $\|\cdot\|_\infty$, il est noté $\mathrm{cond}_p(A)$, *et* $\mathrm{cond}_\infty(A)$, *respectivement.*

Le conditionnement s'interprète comme le facteur d'amplification des erreurs relatives lors de la résolution d'un système linéaire :

Proposition 1.4

Soient $A \in \mathbb{R}^{d\times d}$ inversible, $b, \delta b \in \mathbb{R}^d$. On note $x \in \mathbb{R}^d$ et $x + \delta x \in \mathbb{R}^d$ les solutions des systèmes linaires

$$Ax = b, \qquad A(x + \delta x) = b + \delta b.$$

Alors

$$\frac{\|\delta x\|}{\|x\|} \leqslant \mathrm{cond}(A) \frac{\|\delta b\|}{\|b\|}.$$

Propriétés 1.5

Soit $A \in \mathbb{R}^{d\times d}$ une matrice inversible.

- ◇ *Quelle que soit la norme subordonnée choisie,* $\mathrm{cond}(A) \geqslant 1$.
- ◇ *Si A est symétrique, alors*

$$\mathrm{cond}_2(A) = \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)}.$$

- ◇ *Si A est orthogonale, alors* $\mathrm{cond}_2(A) = 1$.

1.1.3 Autour de l'algorithme de Gauss

L'algorithme du pivot de Gauss

Le principe consiste à effectuer des opérations sur les lignes de la matrice A pour l'*échelonner*, et se ramener à un système triangulaire inférieur. Les différentes étapes sont détaillées ci-dessous.

- ◇ **Étape 0 :** on initialise l'algorithme à partir de la matrice A et du vecteur second membre b, selon les notations suivantes :

$$\left|\begin{array}{cccc} a_{1,1}^{[0]} & a_{1,2}^{[0]} & \dots & a_{1,d}^{[0]} \\ a_{2,1}^{[0]} & a_{2,2}^{[0]} & \dots & a_{2,d}^{[0]} \\ \vdots & & & \\ a_{d,1}^{[0]} & a_{d,2}^{[0]} & \dots & a_{d,d}^{[0]} \end{array}\right| \quad \left|\begin{array}{c} b_1^{[0]} \\ b_2^{[0]} \\ \vdots \\ b_d^{[0]} \end{array}\right|$$

- ◇ **Étape 1.1 :** si $a_{1,1}^{[0]} = 0$, on permute les lignes 1 et k, opération notée $\boxed{L_1 \leftrightarrow L_k}$ de telle sorte que $a_{k,1}^{[0]} \neq 0$.

$$\left|\begin{array}{cccc} \tilde{a}_{1,1}^{[0]} & \tilde{a}_{1,2}^{[0]} & \dots & \tilde{a}_{1,d}^{[0]} \\ \tilde{a}_{2,1}^{[0]} & \tilde{a}_{2,2}^{[0]} & \dots & \tilde{a}_{2,d}^{[0]} \\ \vdots & & & \\ \tilde{a}_{d,1}^{[0]} & \tilde{a}_{d,2}^{[0]} & \dots & \tilde{a}_{d,d}^{[0]} \end{array}\right| \quad \left|\begin{array}{c} \tilde{b}_1^{[0]} \\ \tilde{b}_2^{[0]} \\ \vdots \\ \tilde{b}_d^{[0]} \end{array}\right|$$

- ◇ **Étape 1.2 :** Maintenant $\tilde{a}_{1,1}^{[0]} \neq 0$; on effectue $\boxed{L_k \leftarrow L_k - r_k L_1}$ avec $r_k = \frac{\tilde{a}_{k,1}^{[0]}}{\tilde{a}_{1,1}^{[0]}}$.

$$\left|\begin{array}{cccc} \tilde{a}_{1,1}^{[0]} & \tilde{a}_{1,2}^{[0]} & \dots & \tilde{a}_{1,d}^{[0]} \\ 0 & a_{2,2}^{[1]} & \dots & a_{2,d}^{[1]} \\ \vdots & & & \\ 0 & a_{d,2}^{[1]} & \dots & a_{d,d}^{[1]} \end{array}\right| \quad \left|\begin{array}{c} \tilde{b}_1^{[0]} \\ b_2^{[1]} \\ \vdots \\ b_d^{[1]} \end{array}\right|$$

- ◇ **Étape 2 :** On recommence avec la sous-matrice correspondant aux indices $2 \leqslant i, j \leqslant d$:

$$\left|\begin{array}{cccc} \tilde{a}_{1,1}^{[0]} & \tilde{a}_{1,2}^{[0]} & \dots & \tilde{a}_{1,d}^{[0]} \\ 0 & \boxed{\begin{array}{ccc} a_{2,2}^{[1]} & \dots & a_{2,d}^{[1]} \\ \vdots & & \\ a_{d,2}^{[1]} & \dots & a_{d,d}^{[1]} \end{array}} & & \\ 0 & & & \\ \vdots & & & \\ 0 & & & \end{array}\right| \quad \left|\begin{array}{c} \tilde{b}_1^{[0]} \\ \boxed{\begin{array}{c} b_2^{[1]} \\ \vdots \\ b_d^{[1]} \end{array}} \end{array}\right|$$

À l'issue des $d-1$ étape de l'algorithme, la matrice est devenue triangulaire, et on est ramené au système linéaire $Uy = c$: U est la matrice triangulaire obtenue à la fin de l'algorithme, c est le vecteur second membre final, et y contient les composantes de la solution recherchée x, dans un ordre éventuellement différent si des permutations ont eu lieu.

La décomposition LU

Définition 1.4

Soit $A \in \mathbb{R}^{d\times d}$ une matrice. On appelle mineur fondamental *d'ordre $k \leqslant d$ le déterminant*

$$\det\left[(A_{ij})_{1\leqslant i,j\leqslant k}\right].$$

La décomposition LU est une interprétation matricielle de l'algorithme du pivot de Gauss :

Théorème 1.6 (*Décomposition LU*)

Soit $A \in \mathbb{R}^{d\times d}$ une matrice ayant tous ses mineurs fondamentaux non nuls. Alors il existe un unique couple $(L, U) \in (\mathbb{R}^{d\times d})^2$ tel que

- ◇ *L est triangulaire inférieure avec des 1 sur la diagonale,*
- ◇ *U est triangulaire supérieure,*
- ◇ *$A = LU$.*

Lorsque A admet une décomposition LU on peut, pour résoudre le système linéaire $Ax = b$, résoudre successivement les systèmes triangulaires

$$Ly = b \quad \text{et} \quad Ux = y.$$

Le coût de résolution d'un système quelconque de taille $d \times d$ par l'algorithme de Gauss est de l'ordre $\mathcal{O}(d^3)$. Il est bien moins coûteux de résoudre un système triangulaire,par descente ou remontée, en $\mathcal{O}(d^2)$. Toutefois le coût du calcul de L et U est également $\mathcal{O}(d^3)$.

L'intérêt de la décomposition LU réside dans la résolution de nombreux systèmes linéaires avec la même matrice et des seconds membres différents. On peut, en effet, calculer (et stocker) une fois pour toutes les matrices L et U et résoudre les p systèmes linéaires triangulaires ensuite. Le coût de calcul est $\mathcal{O}(d^3 + pd^2)$, inférieur à $\mathcal{O}(pd^3)$ requis par la méthodes de Gauss appliquée aux p systèmes linéaires. Observons toutefois que cela n'a d'intérêt que lorsque les seconds membres ne sont pas connus tous à l'avance (car sinon, on peut appliquer l'algorithme de Gauss une seule fois avec comme second membre la matrice concaténant les p seconds membres). Un exemple d'une telle situation se rencontre lorsqu'on utilise la méthode de la puissance inverse (voir §1.2).

Remarque 1.1

- ◇ *Dans le cas, plus général, où A est seulement inversible, on peut montrer qu'il existe une matrice de permutation (non unique) P telle que PA admette une décomposition LU.*

Corollaire 1.7 (*Décomposition de Choleski*)

Soit A une matrice symétrique définie positive. Il existe une unique matrice B, triangulaire inférieure avec des entrées diagonales strictement positives, telle que

$$A = BB^T.$$

1.1.4 Quelques méthodes itératives

En vue de résoudre un système linéaire $Ax = b$, les méthodes itératives construisent une suite de vecteurs $(x^{(n)})_n$ qui converge vers la solution x. L'idée générale consiste à

décomposer la matrice A en

$$A = M - N,$$

avec M inversible et « simple ». Le système linéaire est alors équivalent au problème de point fixe

$$Mx = Nx + b.$$

Il est alors naturel de construire la suite récurrente $(x^{(n)}$ par la relation

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

Théorème 1.8

Soit $B \in \mathbb{R}^{d\times d}$ et $c \in \mathbb{R}^d$. La suite $(x^{(n)})_n$ définie par

$$x^{(n+1)} = Bx^{(n)} + c$$

converge pour tout choix de $x^{(0)}$ et tout choix de c si et seulement si

$$\rho(B) < 1.$$

Dans ce cas, la convergence est géométrique : si x désigne la limite, alors

$$\|x - x^{(n)}\| = \mathcal{O}\left(\rho(B)^n\right).$$

La méthode de Jacobi correspond au choix $M = D$. Elle est bien définie sitôt que la matrice A n'a pas de 0 sur la diagonale. La méthode de Gauss-Seidel, quant à elle, revient à choisir pour M la partie triangulaire (diagonale comprise) de A. Elle est bien définie dans les mêmes conditions. La question de la convergence de ces méthodes n'est pas simple dans le cas général.

1.2 Calcul d'éléments propres : méthodes de puissance

Théorème 1.9 (*Méthode de la puissance*)

Soit $A \in \mathbb{R}^{d\times d}$ une matrice dont on note $\lambda_1, \dots, \lambda_p$, avec $p \leqslant d$, les valeurs propres complexes distinctes. On suppose que

$$|\lambda_1| \leqslant |\lambda_2| \leqslant \cdots \leqslant |\lambda_{p-1}| < |\lambda_p|,$$

et on construit la suite

$$\begin{cases} y^{(n+1)} = Ax^{(n)}, \\ x^{(n+1)} = \dfrac{y^{(n+1)}}{\|y^{(n+1)}\|_2}, \\ x^{(0)} \in \mathbb{R}^d. \end{cases}$$

Si $x^{(0)}$ n'est pas choisi dans le sous-espace engendré par les vecteurs propres associées à $\{\lambda_1, \dots, \lambda_{p-1}\}$, alors la suite

$$(Ax^{(n)}|x^{(n)}) \quad \text{converge vers} \quad \lambda_p.$$

Remarque 1.2

- ◇ *En général la suite $(x^{(n)})$ ne converge pas bien qu'asymptotiquement le vecteur $x^{(n)}$ est proche d'un vecteur propre associé à λ_p.*
- ◇ *L'hypothèse $|\lambda_{p-1}| < |\lambda_p|$ est essentielle pour assurer la convergence de l'algorithme. Des phénomènes d'oscillation peuvent avoir lieu dans le cas où cette hypothèse n'est pas satisfaite. On peut alors étudier la matrice $A + \varepsilon I$ si cette situation arrive.*
- ◇ *Si l'on applique la méthode de la puissance à la matrice A^{-1} (on n'inverse pas explicitement la matrice, mais on résout un système linéaire à chaque itération), on peut approcher la plus petite valeur propre (en module) de A. On parle de méthode de la puissance inverse. De même, si on applique la méthode à $(A - \mu I_d)^{-1}$, on approchera la valeur propre la plus proche du nombre complexe μ (méthode de la puissance inverse avec translation).*
- ◇ *Il est possible, notamment dans le cas où A est symétrique, d'adapter la méthode pour les quelques plus grandes valeurs propres de la matrice A. Il s'agit de la méthode dite de* déflation.

PREUVE. (Pour les matrices diagonalisables) Soit $x^{(0)} \in \mathbb{R}^d$, $x \neq 0$. Comme A est diagonalisable, on écrit $A = P^{-1}DP$ avec P inversible et $D = diag(\lambda_p, \dots, \lambda_p, \lambda_{p-1}, \dots, \lambda_1)$. Posons maintenant la suite $z^{(n)}$ définie par

$$z^{(n)} = A^n x^{(0)}, \quad \forall n \in \mathbb{N}.$$

Remarquons que par récurrence $\frac{z^{(n)}}{\left\|z^{(n)}\right\|_2} = x_n$ pour $n \geqslant 1$. En effet $z^{(1)} = y^{(1)}$ et $x^{(1)} = \frac{y^{(1)}}{\left\|y^{(1)}\right\|_2}$, si c'est vrai au rang n, on a

$$\begin{aligned}
z^{(n)} &= \left\|z^{(n)}\right\|_2 x^{(n)}, \\
Az^{(n)} &= \left\|z^{(n)}\right\|_2 Ax^{(n)} = \left\|z^{(n)}\right\|_2 y^{(n+1)} \\
z^{(n+1)} &= \left\|z^{(n)}\right\|_2 y^{(n+1)}, \\
\frac{z^{(n+1)}}{\left\|z^{(n+1)}\right\|_2} &= \frac{y^{(n+1)}}{\left\|y^{(n+1)}\right\|_2} = x^{(n+1)}.
\end{aligned}$$

On réécrit alors $D = \lambda_p \times diag(1, \dots, 1, \alpha_1, \dots, \alpha_r)$ ou les α_i sont des complexes de module strictement inférieurs à 1. Ainsi $D^n = \lambda_p^n \times diag(1, \dots, 1, \alpha_1^n, \dots, \alpha_r^n)$ et donc

$$\frac{1}{\lambda_p^n} D^n \to J,$$

avec $J = \begin{bmatrix} I_m & 0 \\ 0 & 0 \end{bmatrix}$ avec m la multiplicité de la valeur propre λ_p. Ainsi $\frac{1}{\lambda_p^n} A^n \to P^{-1}JP$ et donc

$$\frac{1}{\lambda_p^n} z^{(n)} \to P^{-1}JPx^{(0)}.$$

On pose $\xi = P^{-1}JPx^{(0)} \neq 0$. Supposons que $\xi \neq 0$. Alors la suite $z^{(n)}$ ne peut jamais s'annuler, (sinon nulle à partir d'une certain rang). On a

$$\frac{1}{|\lambda_p|^n}\left\|z^{(n)}\right\|_2 \to \|\xi\|,$$

et donc

$$\frac{z^{(n)}}{\left\|z^{(n)}\right\|_2} \to \frac{\xi}{\|\xi\|_2}, \quad \text{i.e.} \quad x^{(n)} \to \frac{\xi}{\|\xi\|_2}$$

De plus,

$$A\xi = P^{-1}DPP^{-1}JPx^{(0)} = P^{-1}DJPx^{(0)} = \lambda_p P^{-1}JPx^{(0)} = \lambda_p \xi$$

donc $\frac{(A\xi|\xi)}{\|\xi\|_2^2} = \lambda_p$ et on a bien $(Ax^{(n)}|x^{(n)}) \to \lambda_p$. ∎

Chapitre 2

Intégration numérique

Le principe général des méthodes d'intégration numérique (dites aussi méthodes de quadrature) pour le calcul approché de l'intégrale d'une fonction continue f

$$I = \int_a^b f(x)\,\mathrm{d}x,$$

consiste à subdiviser l'intervalle d'intégration $[a, b]$ en

$$x_i = a + ih, \quad \text{avec} \quad h = \frac{b-a}{N},$$

où N est un entier donné. On alors, d'après la relation de Chasles,

$$I = \int_a^b f(x)\,\mathrm{d}x = \sum_{i=0}^{N-1} \int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x.$$

La construction d'une méthode d'intégration numérique consiste donc à approcher l'intégrale sur chacun des sous-intervalles $[x_i, x_{i+1}]$.

2.1 La méthode des rectangles

La méthode des rectangles consiste à utiliser l'approximation suivante :

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x \simeq h f(x_i).$$

L'approximation de I est alors définie par

$$I_N = h \sum_{i=0}^{N-1} f(x_i).$$

Théorème 2.1

Si f est de classe $\mathscr{C}^1$ sur l'intervalle $[a, b]$, alors on a l'estimation d'erreur suivante pour la méthode des rectangles :

$$|I - I_N| \leqslant \frac{(b-a)^2}{2N} \sup_{x\in[a,b]} |f'(x)|.$$

En particulier, on a

$$|I - I_N| = \mathcal{O}\left(\frac{1}{N}\right).$$

2.2 La méthode des trapèzes

On peut améliorer l'approximation en utilisant

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x \simeq h\,\frac{f(x_i) + f(x_{i+1})}{2}.$$

La méthode obtenue, dite *méthode des trapèzes* définit alors l'approximation

$$I_N = \frac{h}{2}\sum_{i=0}^{N-1}\left(f(x_i) + f(x_{i+1})\right) = \frac{h}{2}\left(f(x_0) + f(x_N)\right) + h\sum_{i=1}^{N-1} f(x_i).$$

Ainsi, la méthode des trapèzes ne nécessite qu'une seule évaluation supplémentaire par rapport à la méthode des rectangle. Elle est, en revanche, plus précise.

Théorème 2.2

Si f est de classe $\mathscr{C}^1$ sur l'intervalle $[a, b]$, alors on a l'estimation d'erreur suivante pour la méthode des trapèzes :

$$|I - I_N| \leqslant \frac{(b-a)^3}{N^2}\sup_{x\in[a,b]}|f''(x)|.$$

En particulier, on a

$$|I - I_N| = \mathcal{O}\left(\frac{1}{N^2}\right).$$

2.3 Généralisation : les méthodes composées

2.3.1 Introduction

On décrit ici un procédé qui permet de construire d'autres méthodes d'intégration numérique, dites *composées*. Pour approcher l'intégrale, on la découpe selon une subdivision (ici encore, de pas uniforme h) :

$$I = \int_a^b f(x)\,\mathrm{d}x = \sum_{i=0}^{N-1}\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x.$$

Sur chaque intervalle $[x_i, x_{i+1}]$, on va approcher l'intégrale par une combinaison linéaire de valeurs de f :

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x \simeq \sum_{q=0}^{k_i} w_q^i f(\xi_q^i).$$

Il reste à expliquer comment on choisit les *points* (ou *nœuds*) ξ_q^i et les *poids* w_q^i.

Une manière simple de procéder consiste à choisir k_i indépendant de i, de même que les w_q^i. Pour les points, on se donne un *motif* :

$$\xi_q^i = x_i + t_q h,$$

où $t_0, t_1, \ldots, t_q$ sont donnés. Autrement dit, on utilise l'approximation suivante :

$$\int_{x_i}^{x_{i+1}} f(x)\,\mathrm{d}x = h \int_0^1 \underbrace{f(x_i + th)}_{\varphi_i(t)}\,\mathrm{d}t \simeq h \sum_{q=0}^{k} w_q \varphi_i(t_q).$$

Lorsque l'on utilise l'approximation

$$\int_0^1 \varphi(t)\,\mathrm{d}t \simeq \sum_{q=0}^{k} w_q \varphi(t_q),$$

on parle de *modèle élémentaire*.

Remarque 2.1

Il est important de comprendre qu'un modèle élémentaire ne donne, en général, pas une bonne approximation de l'intégrale de φ sur $[0,1]$. Le seul paramètre sur lequel on pourrait jouer est k, et il n'est pas assuré que lorsque $k \to \infty$, on ait convergence vers l'intégrale.

2.3.2 Cadre formel

Définition 2.1

On appelle modèle élémentaire *toute formule du type*

$$\sum_{q=0}^{k} w_q \varphi(t_q), \tag{2.1}$$

où les points t_q *et les* poids w_q *sont donnés.*

On dit que ce modèle est exact $\mathbb{P}_\ell$ (ou exact au degré ℓ) lorsque, pour tout polynôme $p \in \mathbb{P}_\ell$, on a

$$\sum_{q=0}^{k} w_q p(t_q) = \int_0^1 p(t)\,\mathrm{d}t.$$

Définition 2.2

On appelle méthode composée *à partir du modèle élémentaire* (2.1) *pour le calcul de l'intégrale*

$$I = \int_a^b f(x)\,\mathrm{d}x$$

la formule d'approximation

$$I_N = h \sum_{i=0}^{N-1} \sum_{q=0}^{k} w_q f(x_i + t_q h). \tag{2.2}$$

Théorème 2.3

Si le modèle élémentaire (2.1) *est exacte* $\mathbb{P}_\ell$ *et si* $f \in \mathscr{C}^{\ell+1}([a,b])$*, alors on a l'estimation d'erreur*

$$|I - I_N| \leqslant \frac{(b-a)^{\ell+2}}{N^{\ell+1}(\ell+1)!} \sup_{x\in[a,b]} \left|f^{(\ell+1)}(x)\right|.$$

Remarque 2.2

La méthode des rectangles est la méthode composée à partir du modèle élémentaire correspondant à

$$k = 0, \quad t_0 = 0, \quad w_0 = 1.$$

Ce modèle est exact $\mathbb{P}_0$*, donc si* $f \in \mathscr{C}^1([a,b])$*, le théorème 2.3 nous assure l'estimation d'erreur*

$$|I - I_N| \leqslant \frac{(b-a)^2}{N} \sup_{x\in[a,b]} \left|f'(x)\right|.$$

On voit que l'estimation du théorème 2.1 est plus précise (en raison du facteur 2 *au dénominateur). Toutefois, l'ordre de convergence* $\mathcal{O}(N^{-1})$ *est le même dans les deux cas.*

2.3.3 Calcul des poids

On suppose que les points d'intégration (encore appelés *points de quadrature* sont donnés. On a intérêt à ajuster les poids pour que le modèle élémentaire soit exacte $\mathbb{P}_\ell$ pour le plus grand ℓ possible.

Proposition 2.4

Pour tout choix des points $t_0 < t_1 < \cdots < t_k$*, il existe un unique choix de poids* $w_0, w_1, \ldots, w_k$ *tel que le modèle élémentaire soit exact* $\mathbb{P}_k$*. Le vecteur* $W = (w_0, w_1, \ldots, w_k)^T$ *est solution du système linéaire de Vandermonde*

$$\mathbb{V}W = b,$$

avec

$$\forall 0 \leqslant i, j \leqslant k, \quad \mathbb{V}_{ij} = t_j^i \quad \text{et} \quad b_i = \frac{1}{i+1}.$$

Notons qu'il n'est, en général, pas possible de calculer les poids pour que la formule soit exacte $\mathbb{P}_\ell$ avec $\ell > k$.

2.3.4 Méthode de Gauss-Legendre

Si l'on suppose que les points de quadrature peuvent également être ajustés, il est possible d'augmenter le degré d'exactitude d'un modèle élémentaire au delà de k.

Théorème 2.5 (*Méthode de Gauss-Legendre*)

Soit $(P_0, P_1, \ldots)$ *la famille de polynômes orthogonaux (avec* $d^o P_i = i$*) pour le produit scalaire*

$$(\varphi, \psi) = \int_0^1 \varphi(t)\psi(t)\,\mathrm{d}t.$$

On note $(t_q)_{q=0,\dots,k}$ les racines de P_{k+1}. On détermine les poids $(w_q)_{q=0,\dots,k}$ tel que le modèle élémentaire soit exact $\mathbb{P}_k$, i.e. solution de

$$\forall i = 0, 1, \dots, k, \quad \sum_{q=0}^{k} w_q t_q^i = \frac{1}{i+1}.$$

Alors la formule

$$\int_0^1 \varphi(t)\,\mathrm{d}t \simeq \sum_{q=0}^{k} w_q \varphi(t_q)$$

est exacte $\mathbb{P}_{2k+1}$.

2.4 La méthode de Monte-Carlo pour le calcul d'intégrales

Théorème 2.6 (*Loi forte des grands nombres pour une loi uniforme*)

Si $X \hookrightarrow \mathbb{U}([a,b])$ et $f : [a,b] \to \mathbb{R}$, alors

$$\mathbb{E}\left[f(X)\right] = \frac{1}{b-a}\int_a^b f(x)\,\mathrm{d}x.$$

Si $(X_0, X_1, \dots)$ est une suite de variables aléatoires indépendantes de même loi que X, alors

$$\frac{b-a}{N}\sum_{i=0}^{N-1} f(X_i) \quad \textit{converge presque sûrement vers} \quad \int_a^b f(x)\,\mathrm{d}x.$$

Remarque 2.3

- ◇ *Le théorème 2.6 n'est autre que la loi forte des grands nombres, dans le cas particulier de la loi uniforme.*
- ◇ *En pratique, on dispose d'une réalisation (issue d'observations) de l'échantillon, notée (x_0, x_1, x_{N-1}), et la quantité calculable*

$$\frac{b-a}{N}\sum_{i=0}^{N-1} f(x_i)$$

  approche l'intégrale.
- ◇ *Le théorème centrale limite fournit une information sur la vitesse de convergence. Sans rentrer dans les détails, on peut retenir que l'approximation obtenue est de l'ordre $\mathcal{O}(N^{-\frac{1}{2}})$. Cette convergence est plus lente que pour la plus simple des méthodes déterministes (la méthode des rectangles). Toutefois, la méthode de Monte Carlo trouve son intérêt pour le calcul d'intégrales en grande dimension, là où les méthodes déterministes sont complètement inopérantes (en dimension d, mettre 10 points dans chaque direction pour un calcul déterministe requiert 10^d évaluations de la fonction à intégrer).*

Chapitre 3

Optimisation numérique

3.1 Quelques définitions

Définition 3.1

Une fonction $f : \mathbb{R} \to \mathbb{R}$ est dite unimodale *lorsqu'il existe $x^* \in \mathbb{R}$ tel que f soit strictement décroissante sur $]-\infty, x^*[$ et strictement croissante sur $[x^*, +\infty[$.*

Définition 3.2

Une fonction $f : \mathbb{R} \to \mathbb{R}$ est dite coercive *lorsque*

$$\lim_{|x| \to +\infty} f(x) = +\infty.$$

Définition 3.3

Une fonction $f : \mathbb{R} \to \mathbb{R}$ de classe $\mathscr{C}^2$ est dite fortement convexe *lorsqu'il existe un réel $\alpha > 0$ tel que pour tout $x \in \mathbb{R}$, on a $f''(x) \geqslant \alpha$.*

Proposition 3.1

- ◇ *Une fonction fortement convexe est coercive.*
- ◇ *Une fonction coercice et fortement convexe est unimodale.*

3.2 Méthodes de dichotomie en dimension 1

Proposition 3.2 (*Méthode de bissection*)

Soit $f : \mathbb{R} \to \mathbb{R}$ unimodale, dérivable, minimale en $x^* \in [a_0, b_0]$. On définit les suites (a_n), (b_n), (x_n) comme suit :

$$x_n = \frac{a_n + b_n}{2}.$$

- ◇ *Si $f'(x_n) < 0$, alors*
$$a_{n+1} = x_n, \quad \text{et} \quad b_{n+1} = b_n,$$
- ◇ *Si $f'(x_n) > 0$, alors*
$$a_{n+1} = a_n, \quad \text{et} \quad b_{n+1} = x_n.$$

La suite (x_n) converge vers x^*, et on a l'estimation d'erreur

$$|x_n - x^*| \leqslant \frac{b-a}{2^{n+1}}.$$

Cette méthode est très facile à mettre en œuvre si l'on peut évaluer facilement la dérivée f'. Dans le cas contraire, on lui préfère souvent la méthode dite *du nombre d'or*.

Proposition 3.3 (*Méthode du nombre d'or*)

Soit $f : \mathbb{R} \to \mathbb{R}$ unimodale, minimale en $x^* \in [a_0, b_0]$. On fixe aussi c_0, d_0 tels que $a_0 < c_0 < d_0 < b_0$ avec

$$\frac{b_0 - c_0}{b_0 - a_0} = \frac{b_0 - d_0}{b_0 - c_0} = \gamma.$$

On définit les suites (a_n), (b_n), (c_n), (d_n), (x_n) comme suit :

$$x_n = \frac{a_n + b_n}{2}.$$

- *Si $f(c_n) < f(d_n)$, alors*

$$\left|\begin{array}{lcl} a_{n+1} & = & a_n \\ b_{n+1} & = & d_n \\ c_{n+1} & = & b_{n+1} - \frac{b_{n+1}-a_{n+1}}{\gamma} \\ d_{n+1} & = & c_n \end{array}\right.$$

- *Si $f(c_n) > f(d_n)$, alors*

$$\left|\begin{array}{lcl} a_{n+1} & = & c_n \\ b_{n+1} & = & b_n \\ c_{n+1} & = & d_n \\ d_{n+1} & = & a_{n+1} + \frac{b_{n+1}-a_{n+1}}{\gamma} \end{array}\right.$$

La suite (x_n) converge vers x^*, et on a l'estimation d'erreur

$$|x_n - x^*| \leqslant \frac{b-a}{2\gamma^n}.$$

3.3 Optimisation libre dans $\mathbb{R}^d$

Les méthodes de dichotomie présentées dans le paragraphe précédent ne se généralisent pas en dimension supérieure. En effet, elles sont basées sur le théorème des valeurs intermédiaires qui n'a pas d'équivalent à partir de la dimension 2.

3.3.1 Méthodes de descente

Proposition 3.4 (*Méthode du gradient à pas fixe*)

Soit $\mathbf{f} \in \mathscr{C}^1(\mathbb{R}^d, \mathbb{R})$ strictement convexe, coercive, et telle que

$$\exists M > 0, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d, \quad \|\nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{y})\|_2 \leqslant M \|\mathbf{x} - \mathbf{y}\|_2.$$

(on dit que $\nabla \mathbf{f}$ est M-lipschitzien). Si $0 < \rho < \frac{2}{M}$, alors la suite définie par

$$\mathbf{x}^{(0)} \in \mathbb{R}^d, \quad et \quad \mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} - \rho \nabla \mathbf{f}(\mathbf{x}^{(n)})$$

converge vers le point de minimum global de $\mathbf{f}$ sur $\mathbb{R}^d$.

Remarque 3.1

*Si $\mathbf{f}$ est de classe $\mathscr{C}^2$ et si l'on ajoute l'hypothèse (dite de α-*convexité*)*

$$\exists \alpha > 0, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d, \quad (H\mathbf{f}(\mathbf{x})\mathbf{y}|\mathbf{y}) \geqslant \alpha \|\mathbf{y}\|_2^2,$$

alors si $0 < \rho < \frac{2\alpha}{M^2}$, alors la convergence est géométrique.

Proposition 3.5 (*Méthode du gradient à pas optimal*)

*La méthode du gradient à pas optimal (*steepest descent method *en anglais) est définie par l'itération*

$$\begin{cases} \mathbf{x}^{(0)} \in \mathbb{R}^d, \\ \rho_n = \operatorname{argmin} \left\{ \mathbf{f}(\mathbf{x}^{(n)} - \rho \nabla \mathbf{f}(\mathbf{x}^{(n)})) \ ; \ \rho \in \mathbb{R} \right\}, \\ \mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} - \rho_n \nabla \mathbf{f}(\mathbf{x}^{(n)}). \end{cases}$$

Si $\mathbf{f}$ est de classe $\mathscr{C}^2$, α-convexe, coercive, et

$$\exists M > 0, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d, \quad \|\nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{y})\|_2 \leqslant M \|\mathbf{x} - \mathbf{y}\|_2,$$

alors la méthode du gradient à pas optimal converge vers le point de minimum global de $\mathbf{f}$ sur $\mathbb{R}^d$. La convergence est géométrique.

Remarque 3.2

Dans le cas d'une fonctionnelle quadratique

$$\mathbf{f}(\mathbf{x}) = \|A\mathbf{x} - \mathbf{b}\|_2^2,$$

où A est une matrice symétrique définie positive, alors le pas optimal ρ_n peut être calculé explicitement :

$$\rho_n = \frac{\|\nabla \mathbf{f}(x^{(n)})\|_2^2}{2\|A \nabla \mathbf{f}(x^{(n)})\|_2^2}.$$

3.3.2 Méthode de Newton

Définition 3.4 (*Méthode de Newton pour la recherche de zéros*)

Soit $\mathbf{F} : \mathbb{R}^d \to \mathbb{R}^d$ une fonction de classe $\mathscr{C}^1$. La méthode de Newton consiste à construire la suite $x^{(n)}$ par la récurrence

$$\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} - \left[J\mathbf{F}(\mathbf{x}^{(n)}) \right]^{-1} \mathbf{F}(\mathbf{x}^{(n)}).$$

Lorsque l'on souhaite minimiser une fonction $\mathbf{f} : \mathbb{R}^d \to \mathbb{R}$, on peut recherche ses points critiques, i.e. les zéros de son gradient. La méthode de Newton s'écrit alors, pour $\mathbf{f} \in \mathscr{C}^2$:

$$\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} - \left[H\mathbf{f}(\mathbf{x}^{(n)})\right]^{-1} \nabla \mathbf{f}(\mathbf{x}^{(n)}).$$

Proposition 3.6 (*Convergence de la méthode de Newton*)

Soit $\mathbf{F} \in \mathscr{C}^2(\mathbb{R}^d, \mathbb{R}^d)$ et $\mathbf{x}^* \in \mathbb{R}^d$ tel que $\mathbf{F}(\mathbf{x}^*) = \mathbf{0}$. On suppose $J\mathbf{F}(\mathbf{x}^*)$ inversible. Alors il existe $\eta > 0$ tel que pour tout $\mathbf{x}^{(0)} \in B(\mathbf{x}^*, \eta)$, la méthode de Newton est bien définie et converge vers $\mathbf{x}^*$.

La convergence est quadratique, i.e. il existe $C > 0$ tel que

$$\forall n \geqslant 0, \quad \|\mathbf{x}^{(n+1)} - \mathbf{x}^*\| \leqslant C\|\mathbf{x}^{(n)} - \mathbf{x}^*\|^2.$$

Remarque 3.3

On n'est assuré de la convergence de la méthode de Newton que si le vecteur initial est suffisamment proche d'un zero de F. D'un point de vue pratique, il est très difficile de faire un tel choix. Dans le cas où le problème ne laisse apparaître aucun choix naturel, une stratégie consiste à faire des essais jusqu'à ce que la méthode converge. En cas de convergence, celle-ci est très rapide et la précision machine est obtenue après quelques itérations.

3.3.3 Critères d'arrêt

Les méthodes présentées ci-dessus sont de nature itérative. D'un point de vue pratique, il est essentiel de disposer de *critères d'arrêt* pour décider de stopper l'algorithme. Un choix naturel consiste en le *contrôle de l'incrément* : on arrête l'algorithme lorsque

$$\|\mathbf{x}^{(n+1)} - \mathbf{x}^{(n)}\| < \varepsilon,$$

pour une précision ε donnée. Cela correspond à stopper l'algorithme lorsque les itérés ne sont presque plus modifiés d'une itération à l'autre.

Dans le cas des méthodes de gradient, ce contrôle coïncide (à multiplication par le pas près) à contrôler la norme du gradient. Cela s'interprète en stoppant l'algorithme lorsque l'itéré se trouve dans une zone de très faible pente de la fonction à minimiser.

3.4 Optimisation sous contraintes

3.4.1 Méthode du gradient projeté

Définition 3.5 (*Projection sur on convexe fermé*)

Soit $K \subset \mathbb{R}^d$ un convexe fermé non vide. Pour tout $\mathbf{x} \in \mathbb{R}^d$, il existe un unique point $\pi_K(\mathbf{x}) \in K$ qui minimise la distance à K :

$$\|\mathbf{x} - \pi_K(\mathbf{x})\|_2 = \min_{\mathbf{y} \in K} \|\mathbf{x} - \mathbf{y}\|_2.$$

Proposition 3.7 (*Méthode du gradient projeté*)

Soit $\mathbf{f} \in \mathscr{C}^1(\mathbb{R}^d, \mathbb{R})$ strictement convexe et coercive, et K convexe fermé non vide, avec

$$\exists M > 0, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d, \quad \|\nabla \mathbf{f}(\mathbf{x}) - \nabla \mathbf{f}(\mathbf{y})\|_2 \leqslant M\|\mathbf{x} - \mathbf{y}\|_2,$$

alors la méthode du gradient projeté définie par $\mathbf{x}^{(0)} \in \mathbb{R}^d$ *et*

$$\mathbf{x}^{(n+1)} = \pi_K \left(\mathbf{x}^{(n)} - \rho \nabla \mathbf{f}(\mathbf{x}^{(n)}) \right)$$

converge vers le point de minimum global de $\mathbf{f}$ *sur* K*, dès que* $0 < \rho < \frac{2}{M}$.

3.4.2 Méthode de pénalisation

Définition 3.6 (*Fonction de pénalisation*)

Soit $K \subset \mathbb{R}^d$*. On appelle* fonction de pénalisation *de* K *toute fonction* $\beta : \mathbb{R}^d \to \mathbb{R}$ *satisfaisant*

- $\diamond$ β *est continue,*
- $\diamond$ $\beta \geqslant 0$ *sur* $\mathbb{R}^d$*,*
- $\diamond$ $\beta(\mathbf{x}) = 0 \Longleftrightarrow \mathbf{x} \in K$.

La stratégie de pénalisation consiste à remplacer le problème avec contrainte

$$\min_{\mathbf{x} \in K} \mathbf{f}(\mathbf{x}) \tag{3.1}$$

par le problème sans contrainte

$$\min_{\mathbf{x} \in \mathbb{R}^d} \mathbf{f}(\mathbf{x}) + \frac{1}{\varepsilon} \beta(\mathbf{x}). \tag{3.2}$$

Théorème 3.8

Soient $f \in \mathscr{C}^1(\mathbb{R}^d, \mathbb{R})$ *strictement convexe et coercive, et* $K \subset \mathbb{R}^d$ *un convexe fermé non vide. Soit* β *une fonction de pénalisation de* K*. Alors, pour tout* $\varepsilon > 0$*, le problème pénalisé* (3.2) *admet une unique solution* $\mathbf{x}_\varepsilon$*, qui satisfait*

$$\lim_{\varepsilon \to 0} \mathbf{x}_\varepsilon = \mathbf{x},$$

où $\mathbf{x}$ *est l'unique solution du problème avec contrainte* (3.1).

Chapitre 4

Résolution numérique des équations différentielles ordinaires

4.1 Problème de Cauchy

On appelle problème de Cauchy un problème différentiel avec condition initiale :

$$\begin{cases} \mathbf{u}'(t) = \mathbf{f}(t, \mathbf{u}(t)), \quad t \in [0, T], \\ \mathbf{u}(0) = \mathbf{u}_0, \end{cases} \tag{4.1}$$

où $f : [0, T] \times \mathbb{R}^d \to \mathbb{R}^d$ est une fonction donnée, et $u_0 \in \mathbb{R}^d$. Lorsque $d = 1$, le problème est *scalaire*, il s'agit d'une équation différentielle avec condition initiale. Dans le cas $d \geqslant 2$, on a affaire à un système d'équations différentielles avec conditions initiales : c'est un problème *vectoriel*.

Si l'on a

$$\exists \tilde{\mathbf{f}} : \mathbb{R}^d \to \mathbb{R}^d, \quad \forall t \in [0, T], \quad \forall \mathbf{v} \in \mathbb{R}^d, \quad \mathbf{f}(t, \mathbf{v}) = \tilde{\mathbf{f}}(\mathbf{v}),$$

alors on dit que le problème est *autonome*.

On rappelle ici le théorème de Cauchy-Lipschitz global, qui fournit un cadre de résolution pour le problème (4.1).

Théorème 4.1 (*Cauchy-Lipschitz global*)

On suppose que f est continue et qu'il existe $L > 0$ tel que

$$\forall t \in [0, T], \quad \forall \mathbf{v}, \mathbf{w} \in \mathbb{R}^d, \quad \|\mathbf{f}(t, v) - \mathbf{f}(t, w)\| \leqslant L\|\mathbf{v} - \mathbf{w}\|.$$

Alors le problème (4.1) *admet une unique solution $\mathbf{u} \in \mathscr{C}^1([0, T], \mathbb{R}^d)$.*

4.2 Principe des méthodes d'approximation

Afin d'approcher numériquement la solution u du problème (4.1), on introduit un entier $N_h \geqslant 1$ lié au pas de discrétisation h par $h = \frac{T}{N_h}$, et la subdivision

$$\forall n = 0, 1, \ldots, N_h, \quad t_n = nh.$$

Une méthode (ou schéma) numérique consiste en la construction des valeurs

$$(\mathbf{U}_0, \mathbf{U}_1, \ldots, \mathbf{U}_{N_h})$$

dont on espère qu'elles approcheront les évaluations de la solution sur la subdivision

$$(\mathbf{u}(t_0), \mathbf{u}(t_1), \ldots, \mathbf{u}(t_{N_h})).$$

Bien sûr, sauf cas exceptionnel, $\mathbf{U}_n \neq \mathbf{u}(t_n)$. Par ailleurs, insistons sur le fait que, lorsque N_h change, les valeurs des temps auxquels ont approche la solution changent également. Par exemple, $t_1 = h = \frac{T}{N_h}$.

4.3 La méthode d'Euler

La méthode d'Euler consiste à construire la suite $(\mathbf{U}_n)$ grâce à la relation de récurrence

$$\mathbf{U}_{n+1} = \mathbf{U}_n + h\mathbf{f}(t_n, \mathbf{U}_n),$$

l'initialisation étant naturellement effectuée par $\mathbf{U}_0 = \mathbf{u}_0$ (condition initiale donnée par le problème de Cauchy).

La méthode d'Euler consiste à suivre la tangente à la courbe intégrale passant par le point $(t_n, \mathbf{U}_n)$ pendant un pas de temps h, comme le représente la figure (4.1).

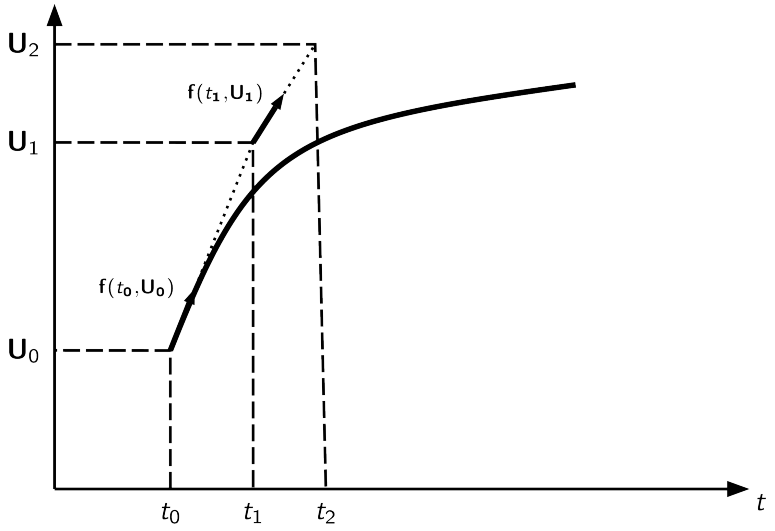

FIGURE 4.1 – La méthode d'Euler en dimension 1.

4.4 Méthodes classiques

Définition 4.1 (*Schéma numérique à un pas*)

Un schéma général à un pas et une relation de récurrence de la forme

$$\mathbf{U}_{n+1} = \mathbf{U}_n + h\mathbf{G}_h(t_n, \mathbf{U}_n, \mathbf{U}_{n+1}), \quad n \in \mathbb{N}.$$

Si $\mathbf{G}$ *ne dépend pas de* $\mathbf{U}_{n+1}$ *le schéma est explicite, sinon il est implicite.*

Voici une liste non exhaustives des schémas à un pas les plus classiques.

◇ Méthode d'Euler rétrograde (implicite, ordre 1)

$$\mathbf{U}_{n+1} = \mathbf{U}_n + h\mathbf{f}(\mathbf{U}_{n+1}, t_{n+1}).$$

◇ Méthode de Heun (explicite, ordre 2)

$$\mathbf{U}_{n+1} = \mathbf{U}_n + \frac{h}{2}\left[\mathbf{f}(\mathbf{U}_n, t_n) + \mathbf{f}(\mathbf{U}_n + h\mathbf{f}(\mathbf{U}_n, t_n), t_{n+1})\right].$$

◇ La méthode de Crank-Nicolson (implicite, ordre 2)

$$\mathbf{U}_{n+1} = \mathbf{U}_n + \frac{h}{2}\left[\mathbf{f}(\mathbf{U}_n, t_n) + \mathbf{f}(\mathbf{U}_{n+1}, t_{n+1})\right].$$

◇ La méthode de Runge-Kutta 2 (RK2) (explicite, ordre 2)

$$\mathbf{U}_{n+1} = \mathbf{U}_n + h\mathbf{f}\left(\mathbf{U}_n + \frac{h}{2}\mathbf{f}(\mathbf{U}_n, t_n), t_n + \frac{h}{2}\right).$$

◇ La méthode de Runge-Kutta 4 (RK4) (explicite, ordre 4)

$$\begin{cases} \mathbf{k}_1 = \mathbf{f}(\mathbf{U}_n, t_n) \\ \mathbf{k}_2 = \mathbf{f}\left(\mathbf{U}_n + \frac{h}{2}\mathbf{k}_1, t_n + \frac{h}{2}\right) \\ \mathbf{k}_3 = \mathbf{f}\left(\mathbf{U}_n + \frac{h}{2}\mathbf{k}_2, t_n + \frac{h}{2}\right) \\ \mathbf{k}_4 = \mathbf{f}(\mathbf{U}_n + h\mathbf{k}_3, t_{n+1}) \\ \mathbf{U}_{n+1} = \mathbf{U}_n + \frac{h}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4). \end{cases}$$

4.5 Consistance, stabilité, convergence

Considérons un schéma numérique sous la forme

$$\forall n = 0, 1, \ldots, N_h - 1, \quad \mathbf{U}_{n+1} = \mathbf{U}_n + h\mathbf{G}(t_n, \mathbf{U}_n, \mathbf{U}_{n+1}). \tag{4.2}$$

Si la fonction G ne dépend pas de la variable $\mathbf{U}_{n+1}$, on dit que le schéma est *explicite*, sinon on dit qu'il est *implicite*.

Définition 4.2 (*Erreur de consistance*)

L'erreur de consistance (locale) du schéma (4.2) *est définie par*

$$\forall n = 0, 1, \ldots, N_h - 1, \quad \varepsilon_n^h = \frac{\mathbf{u}(t_{n+1}) - \mathbf{u}(t_n)}{h} - \mathbf{G}(t_n, \mathbf{u}(t_n), \mathbf{u}(t_{n+1})),$$

où u désigne la solution du problème de Cauchy (4.1).

Définition 4.3 (*Consistance, ordre de consistance*)

On dit que la méthode (4.2) *est consistante avec le problème* (4.1) *lorsque l'erreur de*

consistance locale converge vers 0 *lorsque* N_h *tend vers l'infini (i.e.* h *tend vers* 0*) :*

$$\lim_{h\to 0} \max_{n=0}^{N_h-1} \|\varepsilon_n^h\| = 0.$$

On dit que la méthode est consistante d'ordre k *lorsque*

$$\max_{n=0}^{N_h-1} \|\varepsilon_n^h\| = \mathcal{O}(h^k).$$

Définition 4.4 (*Stabilité*)

Soit le schéma perturbé

$$\forall n = 0, 1, \ldots, N_h - 1, \quad \mathbf{V}_{n+1} = \mathbf{V}_n + h\mathbf{G}(t_n, \mathbf{V}_n, \mathbf{V}_{n+1}) + \mu_n,$$

initialisé avec $V_0 = u_0$*.*

On dit que le schéma (4.2) *est stable lorsqu'il existe une constante* C*, telle que pour tout choix de* (μ_n)*, on ait*

$$\max_{n=0}^{N_h} \|\mathbf{U}_n - \mathbf{V}_n\| \leqslant C \sum_{n=0}^{N_h-1} \|\mu_n\|.$$

Définition 4.5 (*Convergence*)

On dit que la méthode (4.2) *est convergente lorsque*

$$\lim_{h\to 0} \max_{n=0}^{N_h} \|\mathbf{U}_n - \mathbf{u}(t_n)\| = 0.$$

La convergence est dite d'ordre k *lorsque*

$$\max_{n=0}^{N_h} \|\mathbf{U}_n - \mathbf{u}(t_n)\| = \mathcal{O}(h^k).$$

Théorème 4.2 (*Lax*)

Une méthode consistante et stable est convergente. Si la consistance est d'ordre k*, la convergence est également d'ordre* k*.*

4.6 Schémas classiques à un pas

Méthode d'Euler (explicite)

$$\mathbf{U}_{n+1} = \mathbf{U}_n + h\mathbf{f}(t_n, \mathbf{U}_n).$$

Méthode d'Euler implicite

$$\mathbf{U}_{n+1} = \mathbf{U}_n + h\mathbf{f}(t_{n+1}, \mathbf{U}_{n+1}).$$

Méthode de Heun

$$\mathbf{U}_{n+1} = \mathbf{U}_n + \frac{h}{2}\Big[(\mathbf{f}(t_n, \mathbf{U}_n) + \mathbf{f}\big(t_{n+1}, \mathbf{U}_n + h\mathbf{f}(t_n, \mathbf{U}_n)\big)\Big].$$

Méthode de Crank-Nicolson

$$\mathbf{U}_{n+1} = \mathbf{U}_n + \frac{h}{2}\Big[(\mathbf{f}(t_n, \mathbf{U}_n) + \mathbf{f}(t_{n+1}, \mathbf{U}_{n+1})\Big].$$

Méthode de Runge-Kutta 2

$$\left|\begin{array}{l} \mathbf{k}_1 = \mathbf{f}\left(t_n, \mathbf{U}_n\right). \\ \mathbf{k}_2 = \mathbf{f}\left(t_n + \dfrac{h}{2}, \mathbf{U}_n + \dfrac{h}{2}\mathbf{k}_1\right). \\ \mathbf{U}_{n+1} = \mathbf{U}_n + h\mathbf{k}_2. \end{array}\right.$$

Méthode de Runge-Kutta 4

$$\left|\begin{array}{l} \mathbf{k}_1 = \mathbf{f}\left(t_n, \mathbf{U}_n\right). \\ \mathbf{k}_2 = \mathbf{f}\left(t_n + \dfrac{h}{2}, \mathbf{U}_n + \dfrac{h}{2}\mathbf{k}_1\right). \\ \mathbf{k}_3 = \mathbf{f}\left(t_n + \dfrac{h}{2}, \mathbf{U}_n + \dfrac{h}{2}\mathbf{k}_2\right). \\ \mathbf{k}_4 = \mathbf{f}\left(t_n + h, \mathbf{U}_n + h\mathbf{k}_3\right). \\ \mathbf{U}_{n+1} = \mathbf{U}_n + \dfrac{h}{6}\left(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4\right). \end{array}\right.$$

Chapitre 5

Discrétisation de l'équation de Laplace par différences finies

5.1 Un problème mono-dimensionnel

On considère le problème

$$\begin{cases} -u''(x) = f(x) \quad \text{pour } x \in]0,1[, \\ u(0) = \alpha, \\ u(1) = \beta, \end{cases} \tag{5.1}$$

où la fonction f est donnée, et α, β sont deux réels fixés.

Un entier $N \geqslant 1$ étant fixé, on introduit le pas de discrétisation

$$h = \frac{1}{N+1},$$

et la subdivision

$$\forall i = 0, 1, \ldots, N+1, \quad x_i = ih.$$

Définition 5.1

L'approximation de la dérivée seconde à 3 points est donnée par

$$u''(x) \simeq \frac{u(x-h) - 2u(x) + u(x+h)}{h^2}.$$

Si u est de classe $\mathscr{C}^4$, on a l'estimation d'erreur

$$\left| u''(x) - \frac{u(x-h) - 2u(x) + u(x+h)}{h^2} \right| \leqslant \frac{h^2}{12} \max_{\xi \in [x-h, x+h]} |f^{(4)}(\xi)|.$$

En utilisant l'approximation de la dérivée seconde à 3 points en chaque x_i pour $1 \leqslant i \leqslant N$, on obtient le problème discret suivant, où U_i fournira une approximation de $u(x_i)$

$$\forall 1 \leqslant i \leqslant N, \quad \frac{-U_{i-1} + 2U_i - U_{i+1}}{h^2} = f(x_i),$$

et U_0, U_{N+1} sont données par les conditions aux limites :

$$U_0 = \alpha, \quad U_{N+1} = \beta.$$

On obtient le système linéaire $\mathbb{A}\mathbb{U} = \mathbb{F}$, avec

$$\mathbb{A} = \frac{1}{h^2}\begin{pmatrix} 2 & -1 & & & \\ -1 & \ddots & \ddots & & \\ & \ddots & \ddots & -1 & \\ & & -1 & 2 & \end{pmatrix}, \quad \mathbb{U} = \begin{pmatrix} U_1 \\ U_2 \\ \vdots \\ U_N \end{pmatrix}, \quad \mathbb{F} = \begin{pmatrix} f(x_1) + \alpha/h^2 \\ f(x_2) \\ \vdots \\ f(x_{N-1}) \\ f(x_N) + \beta/h^2 \end{pmatrix}.$$

Remarque 5.1

Il est possible de choisir pour vecteur d'inconnues

$$\mathbb{V} = \begin{pmatrix} U_0 \\ U_1 \\ U_2 \\ \vdots \\ U_{N+1} \end{pmatrix} \in \mathbb{R}^{N+2}.$$

Le système linéaire obtenu peut alors s'écrire $\mathbb{B}\mathbb{V} = \mathbb{G}$,

$$\mathbb{B} = \frac{1}{h^2}\begin{pmatrix} 1 & 0 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & 0 & 1 \end{pmatrix}, \quad \mathbb{G} = \begin{pmatrix} \alpha/h^2 \\ f(x_1) \\ f(x_2) \\ \vdots \\ f(x_N) \\ \beta/h^2 \end{pmatrix}$$

Théorème 5.1

La matrice $\mathbb{A}$ *est symétrique définie positive. Ses valeurs propres sont données par*

$$\lambda_\ell = \frac{4}{h^2}\sin^2\left(\frac{\ell\pi h}{2}\right), \quad \ell = 1, 2, \ldots, N.$$

Théorème 5.2

Si la solution du problème (5.1) *est de classe* $\mathscr{C}^4$ *sur l'intervalle fermé* $[0,1]$, *alors*

$$\max_{i=1}^{N} |u(x_i) - U_i| \leqslant \frac{h^2}{96} \sup_{x\in[0,1]} \left|u^{(4)}(x)\right|$$

Remarque 5.2

Si f *est de classe* $\mathscr{C}^2$, *alors* u *est de classe* $\mathscr{C}^4$.

5.2 En dimension supérieure

On considère le problème modèle

$$\begin{cases} -\Delta u = f & \text{dans } \Omega, \\ u = 0 & \text{sur } \partial\Omega. \end{cases} \tag{5.2}$$

Définition 5.2

L'approximation du laplacien à 5 points est donnée par

$$\Delta u(x,y) \simeq \Delta_h u(x,y) = \frac{u(x-h,y)+u(x+h,y)+u(x,y-h)+u(x,y+h)-4u(x,y)}{h^2}.$$

Si u est de classe $\mathscr{C}^4$, on a l'estimation d'erreur

$$|\Delta u(x,y) - \Delta_h u(x,y)| \leqslant \frac{h^2}{6} \max_{\xi\in[x-h,x+h],\eta\in[y-h,y+h]} \left| \frac{\partial^4 f}{\partial x^{(4)}}(\xi,\eta) \right| + \left| \frac{\partial^4 f}{\partial y^{(4)}}(\xi,\eta) \right|.$$

Si Ω est le carré $[0,1]^2$, on introduit le pas

$$h = \frac{1}{N+1},$$

où $N \geqslant 1$ est un entier fixé, et la subdivision

$$\forall 0 \leqslant i,j \leqslant N+1, \quad x_i = ih, \quad y_j = jh.$$

Alors, l'approximation du laplacien à 5 points conduit au schéma discret

$$\forall 1 \leqslant i,j \leqslant N, \quad \frac{4U_{i,j} - U_{i-1,j} - U_{i+1,j} - U_{i,j-1} - U_{i,j+1}}{h^2} = f(x_i,y_j),$$

les $U_{i,j}$ représentant les approximations construites pour $u(x_i,y_j)$. Bien sûr, les conditions aux limites fournissent

$$\forall 0 \leqslant i,j \leqslant N+1, \quad U_{0,j} = U_{N+1,j} = U_{i,0} = U_{i,N+1} = 0.$$

Si l'on ré-ordonne les $(U_{i,j})_{1\leqslant i,j\leqslant N}$ selon l'ordre lexicographique, on obtient le système linéaire

$$\mathbb{A}\mathbb{U} = \mathbb{F},$$

où

$$\mathbb{A} = \frac{1}{h^2} \left(\begin{array}{c|c|c|c} H & -I & & \\ \hline -I & \ddots & \ddots & \\ \hline & \ddots & \ddots & -I \\ \hline & & -I & H \end{array} \right) \in \mathbb{R}^{N^2\times N^2},$$

avec

$$H = \begin{pmatrix} 4 & -1 & & \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 4 \end{pmatrix} \in \mathbb{R}^{N\times N}, \quad I = I_N \in \mathbb{R}^{N\times N}.$$

Le vecteur $\mathbb{F}$ est donné par les valeurs $f(x_i, y_j)$, ré-ordonnées de la même manière que les $U_{i,j}$.

Exemple. Considérons le cas où $\Omega = [0,1]^2$, et $N = 4$. La numérotation des nœuds par ordre lexicographique est indiquée sur la figure 5.1. La matrice $\mathbb{A}$ est donnée par

$$\mathbb{A} = \begin{pmatrix}
4 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 4 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 4 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 4 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 & 4 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & -1 & 4 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & 0 & -1 & 4 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & -1 & 4 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 4 & -1 & 0 & 0 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 4 & -1 & 0 & 0 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 4 & -1 & 0 & 0 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 4 & 0 & 0 & 0 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 4 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 4 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 4 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 4
\end{pmatrix}$$

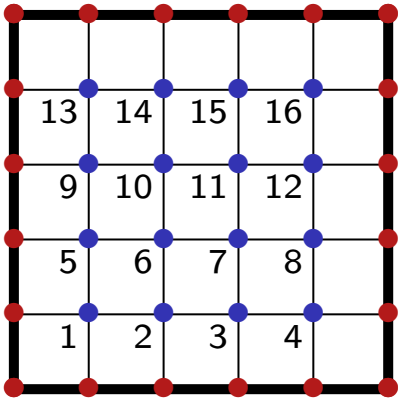

FIGURE 5.1 – Numérotation des nœuds pour $N = 4$.

Dans le vecteur solution $\mathbb{U}$, la huitième composante, par exemple, correspond à l'approximation de $u(4h, 2h)$.

Si l'on change la manière de numéroter les nœuds, la structure de la matrice change (le nombre de coefficients non nuls reste le même, mais leur position est modifiée).

♢

On a une estimation d'erreur similaire à celle obtenue en dimension 1 :

Théorème 5.3

Si la solution u du problème (5.2) *est de classe $\mathscr{C}^4$ sur $\overline{\Omega}$, alors*

$$\max_{i=1}^{N} |u(x_i, y_j) - U_{i,j}| \leqslant \frac{h^2}{48} \sup_{x \in [0,1]} \left| \frac{\partial^4 f}{\partial x^{(4)}}(x, y) \right| + \left| \frac{\partial^4 f}{\partial y^{(4)}}(x, y) \right|.$$

Remarque 5.3

- ◇ *La prise en compte de géométries complexes par cette méthode (dite de* différences finies*) n'est pas aisée.*
- ◇ *Dès la dimension* 2*, l'hypothèse* $f \in \mathscr{C}^2(\overline{\Omega})$ *n'implique* $u \in \mathscr{C}^4(\overline{\Omega})$ *que lorsque le domaine* Ω *a une frontière régulière.*

Chapitre 6

Discrétisation de l'équation de transport

6.1 L'équation de transport linéaire mono-dimensionnelle

Définition 6.1

On appellera équation de transport *(ou d'advection) linéaire une équation aux dérivées partielles de la forme*

$$\begin{cases} \partial_t u(x,t) + v(x,t)\partial_x u(x,t) + a(x,t)u(x,t) = f(x,t), & x \in \mathbb{R},\ t \in [0,T], \\ u(x,0) = u_0(x), & x \in \mathbb{R}, \end{cases} \tag{6.1}$$

où

- u *est la fonction inconnue,*
- v *est la vitesse (connue),*
- a *est le facteur d'amortissement (connu),*
- f *est le terme source (connu),*
- u_0 *est la donnée (ou condition) initiale (connue).*

Dans le cas particulier où $a = f = 0$, et $v(x,t) = c$, vitesse indépendante de l'espace et du temps, on obtient le problème simple

$$\begin{cases} \partial_t u(x,t) + c\partial_x u(x,t) = 0, & x \in \mathbb{R},\ t \in [0,T], \\ u(x,0) = u_0(x), & x \in \mathbb{R}. \end{cases} \tag{6.2}$$

Pour cette équation, on peut déterminer l'unique solution explicitement.

Théorème 6.1

Soit $u_0 : \mathbb{R} \to \mathbb{R}$, de classe $\mathscr{C}^1$. Le problème (6.2) *admet une unique solution, donnée par*

$$\forall x \in \mathbb{R}, \quad \forall t > 0, \quad u(x,t) = u_0(x - ct).$$

6.2 La méthode des caractéristiques

Définition 6.2

On appelle caractéristique *du problème* (6.1) *une fonction* $t \mapsto X(t)$ *solution de l'équation différentielle*

$$X'(t) = v(X(t), t).$$

Proposition 6.2

Si $v : \mathbb{R}\times]0, +\infty[$ *est k-lipschitzienne c'est-à-dire si*

$$\exists k > 0, \quad \forall x, y \in \mathbb{R}, \quad \forall t > 0, \quad |v(x,t) - v(y,t)| \leqslant k|x - y|,$$

alors le problème aux caractéristiques

$$\begin{cases} X'(t) = v(X(t), t), & t \in [0, T], \\ X(t_0) = x_0, \end{cases}$$

(où $t_0 \in [0, T]$ *et* $x_0 \in \mathbb{R}$ *sont fixé) admet une unique solution.*

Dans le cas particulier ou a et f sont nulle on remarque aisément que la fonction définie par $\varphi(t) = u(X(t), t)$ est constante. En effet,

$$\varphi'(t) = \partial_x u(X(t), t)X'(t) + \partial_t u(x, t) = \partial_x u(X(t), t)v(X(t), t) + \partial_t u(x, t) = 0.$$

Cela implique notamment l'églité

$$u(x_0, t_0) = u(X(0), 0) = u_0(X(0)).$$

Autrement dit, on peut trouver la valeur de $u(x_0, t_0)$ en déterminant l'unique caractéristique X qui passe par x_0 à $t = t_0$ et en évaluant u_0 sur la position d'origine $X(0)$ de la caractéristique.

Dans le cas général, φ n'est pas constante mais elle est solution d'une EDO simple. En dérivant,

$$\varphi'(t) = v(X(t), t)\partial_x u(X(t), t) + \partial_t u(X(t), t).$$

et en exploitant l'équation (6.1), on peut écrire

$$\varphi'(t) = f(X(t), t) - a(X(t), t)\varphi(t),$$

qui n'est autre qu'une équation différentielle linéaire satisfaite par la fonction φ. On peut écrire la formule explicite grâce à la formule de Duhamel (variation de la constante) :

$$\varphi(t) = \int_0^t f(X(s), s) \exp\left(-\int_s^t a(X(\tau), \tau)\,\mathrm{d}\tau\right) \mathrm{d}s + \varphi(0) \exp\left(-\int_0^t a(X(\tau), \tau)\,\mathrm{d}\tau\right).$$

Or $\varphi(0) = u(X(0), 0) = u_0(X(0))$ est connu grâce à la donnée initiale. Ainsi, φ est connue pour tout temps $t > 0$, et on en déduit immédiatement l'évaluation de la solution en (x_0, t_0) grâce à l'expression

$$u(x_0, t_0) = \varphi(t_0).$$

On peut montrer que cette approche offre un résultat d'existence et d'unicité pour l'équation de transport.

Théorème 6.3

On se place sous les hypothèses de la proposition 6.2, et on suppose a, et f continues, et u_0 de classe $\mathscr{C}^1$. Alors le problème (6.1) *admet une unique solution de classe $\mathscr{C}^1$ sur $\mathbb{R} \times [0, T]$.*

D'un point numérique, la méthodes des caractéristiques peut donner lieu à la mise en place d'une méthode par résolution numérique des équations différentielles définissant X et φ. Toutefois, il faut répéter l'opération pour chaque nouveau couple (x_0, t_0).

6.3 Approximation par différences finies

On revient au cas particulier de l'équation sans second membre, sans amortissement, et à vitesse constante (6.2).

Pour la discrétisation, on fixe un intervalle d'espace $[a, b]$ sur lequel on va travaillé, et N_x, N_t deux entiers non nuls. On introduit des pas d'espace et de temps

$$\Delta x = \frac{b-a}{N_x}, \qquad \Delta t = \frac{T}{N_t},$$

et on pose

$$x_i = i\Delta x \quad (0 \leqslant i \leqslant N_x) \quad \text{et} \quad t_n = n\Delta t \quad (0 \leqslant n \leqslant N_t).$$

La méthode des différences finies consiste à construire U_i^n qui approche $u(x_i, t_n)$.

Définition 6.3

Un schéma d'approximation (ou de discrétisation) est une relation (homogène à $\partial_t u$) du type

$$G_{\Delta x, \Delta t}\left((U_i^{n+1})_{i=1}^{N_x}, (U_i^n)_{i=1}^{N_x}\right) = 0, \qquad 0 \leqslant n \leqslant N_t,$$

permettant de calculer les U_i^n à partir de la donnée initiale $U_i^0 = u_0(x_i)$. Si $G_{\Delta t, \Delta x}$ ne dépend pas des $(U_j^{n+1})_{j \leqslant i+1}$ le schéma est dit explicite, sinon il est implicite.

Exemple. Un premier schéma s'écrit

$$\frac{U_i^{n+1} - U_i^n}{\Delta t} + c\frac{U_i^n - U_{i-1}^n}{\Delta x} = 0, \tag{6.3}$$

soit encore

$$U_i^{n+1} = (1-\beta)U_i^n + \beta U_{i-1}^n,$$

où l'on a posé

$$\beta = c\frac{\Delta t}{\Delta x}.$$

Ce schéma permet de construire les $(U_i^{n+1})_i$ à partir des $(U_i^n)_i$ seulement pour $i < N_x$. On ajoute la condition aux limites (non présente dans le problème initial) pour régler ce problème :

$$\forall n, \quad U_{N_x}^n = 0.$$

On observe que ce schéma fournit une approximation raisonnable de la solution lorsque $\beta \in [0, 1]$. Ce résultat peut être montré mathématiquement grâce aux notions de consistance et de stabilité. ♢

Définition 6.4

L'erreur de consistance est donnée par

$$\varepsilon_i^n = G_{\Delta x,\Delta t}\big(u(x_i,t_{n+1})\big)_{i=1}^{N_x}, (u(x_i,t_n))_{i=1}^{N_x}\big),$$

où u désigne la solution exacte du problème (6.1).

Exemple. Pour le schéma (6.3), l'erreur de consistance vaut

$$\varepsilon_i^n = \frac{u(x_i,t_{n+1})-u(x_i,t_n)}{\Delta t} + c\frac{u(x_i,t_n)-u(x_{i-1},t_n)}{\Delta x}.$$

♢

Définition 6.5

Le schéma est consistant si

$$\max_{0\leqslant i\leqslant N_x,\ 0\leqslant n\leqslant N_t} |\varepsilon_i^n| \longrightarrow 0 \quad \textit{lorsque} \quad \Delta x, \Delta t \to 0.$$

Il est dit d'ordre q en espace et p en temps, lorsque

$$\max_{0\leqslant i\leqslant N_x,\ 0\leqslant n\leqslant N_t} |\varepsilon_i^n| = \mathcal{O}(\Delta t^p + \Delta x^q).$$

Exemple. Le schéma (6.3) est d'ordre 1 en espace et en temps. ♢

Pour définir la notion de stabilité, on considère un schéma perturbé

$$G_{\Delta x,\Delta t}\left((V_i^{n+1})_{i=1}^{N_x}, (V_i^n)_{i=1}^{N_x}\right) = \mu_i^n, \qquad 0\leqslant i\leqslant N_x, \quad 0\leqslant n\leqslant N_t, \tag{6.4}$$

Définition 6.6

Le schéma est dit stable pour la norme infinie lorsqu'il existe une constante $C>0$ telle que, pour toute perturbation (μ_i^n), on ait

$$\max_{0\leqslant i\leqslant N_x,\ 0\leqslant n\leqslant N_t} |U_i^n - V_i^n| \leqslant C \sum_{n=0}^{N_t-1} \max_{0\leqslant i\leqslant N_x} |\mu_i^n|.$$

Exemple. Le schéma (6.3) est stable pour $\beta\in[0,1]$. ♢

Définition 6.7

Le schéma est dit convergent pour la norme infinie lorsque

$$\max_{0\leqslant i\leqslant N_x,\ 0\leqslant n\leqslant N_t} |U_i^n - u(x_i,t_n)| \longrightarrow 0 \quad \textit{lorsque} \quad \Delta x, \Delta t \to 0.$$

Il est dit d'ordre q en espace et p en temps, lorsque

$$\max_{0\leqslant i\leqslant N_x,\ 0\leqslant n\leqslant N_t} |U_i^n - u(x_i,t_n)| = \mathcal{O}(\Delta t^p + \Delta x^q).$$

En général les schémas peuvent s'écrire simplement à l'aide de matrices. En posant $\mathbf{U}^n = (U_i^n)_{i=0}^{N_x}$ on obtient une relation de récurrence de la forme

$$\mathbf{U}^{n+1} = \mathcal{A}\mathbf{U}^n, \quad n \in \mathbb{N},$$

où $\mathcal{A}$ est une matrice carrée de taille $N_x + 1$.

Proposition 6.4

Le schéma numérique

$$\mathbf{U}^{n+1} = \mathcal{A}\mathbf{U}^n, \quad n \in \mathbb{N},$$

est stable en norme infinie si et seulement si $|||\mathcal{A}|||_\infty \leq 1$.

Théorème 6.5 (*Lax*)

Un schéma numérique consistant et stable est convergent.