

Chapitre 4

Équations aux dérivées partielles (EDP)

IBRAHIM ALAME

ESTP

21/02/2024

Approximations des dérivées d'une fonction régulière

L'idée fondamentale consiste à approcher les dérivées d'une fonction :

$$\frac{du}{dx} = \lim_{h \rightarrow 0} \frac{u(x+h) - u(x)}{h} \simeq \frac{u(x+h) - u(x)}{h}$$

pour un h petit fixé. Si la fonction u est C^2 :

$$u(x+h) = u(x) + hu'(x) + \frac{h^2}{2}u''(\theta)$$

où $\theta \in]x, x+h[$, l'inégalité

$$\left| \frac{u(x+h) - u(x)}{h} - u'(x) \right| \leq Ch$$

où $C = \sup_{y \in [0,1]} |u''(y)|$.

Définition

Si l'erreur est majorée par Ch^p pour $p > 0$ fixé, on dit plus généralement que l'approximation est consistante d'ordre p .

Approximations des dérivées d'une fonction régulière

On vérifie facilement que $u'(x) \simeq \frac{u(x) - u(x-h)}{h}$ est également consistante d'ordre 1.

Si la fonction u est C^2 :

$$u(x-h) = u(x) - hu'(x) + \frac{h^2}{2}u''(\theta)$$

où $\theta \in]x-h, x[$, l'inégalité

$$\left| \frac{u(x) - u(x-h)}{h} - u'(x) \right| \leq Ch$$

où $C = \sup_{y \in [0,1]} |u''(y)|$.

Si la fonction u est C^3 :

$$u(x+h) = u(x) + hu'(x) + \frac{h^2}{2}u''(x) + \frac{h^3}{6}u'''(\xi_1),$$

$$u(x-h) = u(x) - hu'(x) + \frac{h^2}{2}u''(x) - \frac{h^3}{6}u'''(\xi_2),$$

où $\xi_1 \in]x, x+h[$ et $\xi_2 \in]x-h, x[$. On obtient alors :

$$\left| \frac{u(x+h) - u(x-h)}{2h} - u'(x) \right| = \frac{h^2}{2} |u'''(\xi_1) + u'''(\xi_2)| \leq Ch^2$$

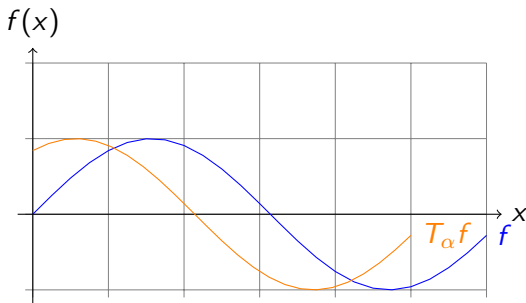
L'approximation

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

est donc consistante d'ordre 2.

Opérateur de translation

$$(T_\alpha f)(x) = f(x + \alpha)$$



- $T_0 = I$
- $T_\alpha \cdot T_\beta = T_{\alpha+\beta}$
- $T_\alpha^n = T_{n\alpha}$

$$u'_d(x) \simeq \frac{u(x+h) - u(x)}{h}$$

$$u'_g(x) \simeq \frac{u(x) - u(x-h)}{h}$$

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

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

On pose

$$\Delta_d = \frac{1}{h}(T_h - I), \quad \Delta_g = \frac{1}{h}(I - T_{-h}), \quad \Delta = \frac{1}{2h}(T_h - T_{-h})$$

Nous avons aussi

$$\Delta = \frac{1}{h}(T_{h/2} - T_{-h/2})$$

On a donc

$$u' = \Delta u, \quad u'_g = \Delta_g u, \quad u'_d = \Delta_d u$$

Plus généralement

$$u'' = \Delta^2 u, \quad u''' = \Delta^3 u, \dots, u^{(n)} = \Delta^n u$$

Approximation de la dérivée seconde

On a

$$\Delta = \frac{1}{h}(T_{h/2} - T_{-h/2})$$

Donc

$$\Delta^2 = \left(\frac{1}{h}(T_{h/2} - T_{-h/2})\right)^2$$

$$\Delta^2 = \frac{1}{h^2}(T_{h/2}^2 - 2T_{h/2}T_{-h/2} + T_{-h/2}^2)$$

$$\Delta^2 u = \frac{1}{h^2}(T_h - 2I + T_{-h})$$

$$\frac{d^2 u(x_i)}{dx^2} = \frac{u(x_i + h) - 2u(x_i) + u(x_i - h)}{h^2}$$

$$u_i'' = \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2}$$

$$\begin{aligned}
\Delta^4 &= \left(\frac{1}{h} (T_{h/2} - T_{-h/2}) \right)^4 \\
&= \frac{1}{h^4} (T_{h/2}^4 - 4T_{h/2}^3 T_{-h/2} + 6T_{h/2}^2 T_{-h/2}^2 - 4T_{h/2} T_{-h/2}^3 + T_{-h/2}^4) \\
&= \frac{1}{h^4} (T_{2h} - 4T_{3h/2} T_{-h/2} + 6T_h T_{-h} - 4T_{h/2} T_{-3h/2} + T_{-2h}) \\
&= \frac{1}{h^4} (T_{2h} - 4T_h + 6I - 4T_{-h} + T_{-2h}) \\
\Delta^4 u(x_i) &= \frac{1}{h^2} (u(x_i + 2h) - 4u(x_i + h) + 6u(x_i) - 4u(x_i - h) + u(x_i - 2h)) \\
\Delta^4 u(x_i) &= \frac{1}{h^2} (u(x_{i+2}) - 4u(x_{i+1}) + 6u(x_i) - 4u(x_{i-1}) + u(x_{i-2}))
\end{aligned}$$

$$\frac{\partial^4 u_i}{\partial x^4} = \frac{1}{h^2} (u_{i+2} - 4u_{i+1} + 6u_i - 4u_{i-1} + u_{i-2})$$

Si u est de classe C^4 au voisinage de x :

$$u(x+h) = u(x) + hu'(x) + \frac{h^2}{2}u''(x) + \frac{h^3}{6}u'''(x) + \frac{h^4}{24}u^{(4)}(\xi_1),$$

$$u(x-h) = u(x) - hu'(x) + \frac{h^2}{2}u''(x) - \frac{h^3}{6}u'''(x) + \frac{h^4}{24}u^{(4)}(\xi_2),$$

Donc

$$u(x+h) + u(x-h) = 2u(x) + h^2u''(x) + \frac{h^4}{24} \left(u^{(4)}(\xi_1) + u^{(4)}(\xi_2) \right),$$

D'où

$$\left| \frac{u(x+h) - 2u(x) + u(x-h)}{h^2} - u''(x) \right| \leq \frac{h^2}{12} \sup_{y \in [0,1]} |u^{(4)}(y)|$$

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

est donc consistante d'ordre 2.

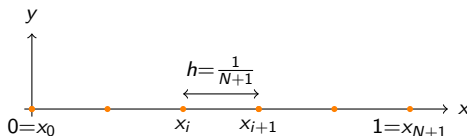
Différences finies ou Problème de Cauchy ?

le problème aux limites d'ordre 2 en dimension 1 On considère le problème aux limites

$$\begin{cases} -\frac{d^2 u}{dx^2} + c(x)u(x) = f(x), & \text{pour } x \in]0, 1[\\ u(0) = u_0, & u(1) = u_1, \end{cases} \quad (1)$$

où f et c sont des fonctions données sur $[0, 1]$ avec $c \geq 0$. Les deux ingrédients principaux d'une approximation par différences finies sont le schéma d'approximation des dérivées et la grille de discrétisation.

Approximation par différences finies



Soit $N \in \mathbb{N}$ fixé. On définit les points de discrétisation du maillage par

$$x_i = ih, \quad i \in 0, 1, \dots, N+1, \quad \text{où } h = \frac{1}{N+1}$$

Les points $x_0 = 0$ et $x_{N+1} = 1$ constituent le bord du domaine (les extrémités de l'intervalle de définition de u), et les points x_1, \dots, x_N sont appelés points internes du maillage.

On cherche en chaque point x_i une valeur approchée $u_i \simeq u(x_i)$. La première et la dernière valeurs de la suite sont données par les conditions aux limites : $u_0 = u(0)$ et $u_{N+1} = u(1)$.

Pour les sommets internes, on utilise l'approximation (2) de la dérivée seconde décrite plus haut. On a

$$\begin{cases} -u''(x_i) + c(x_i)u_i = f(x_i), & \text{pour } i \in \{1, \dots, N\}, \\ u(0) = g_0, \quad u(1) = g_1 \end{cases} \quad (2)$$

Or

$$u''(x_i) \simeq \frac{u(x_i + h) - 2u(x_i) + u(x_i - h))}{h^2} \simeq \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2}$$

D'où le problème approché

$$\begin{cases} -\frac{u_{i-1} - 2u_i + u_{i+1}}{h^2} + c_i u_i = f_i, & \text{pour } i \in \{1, \dots, N\}, \\ u_0 = g_0, \quad u_{N+1} = g_1 \end{cases} \quad (3)$$

où $c_i = c(x_i)$ et $f_i = f(x_i)$.

On obtient N équations à N inconnues u_1, \dots, u_N .

$$\begin{aligned} i = 1 : \quad & -\frac{u_0 - 2u_1 + u_2}{h^2} + c_1 u_1 = f_1, \\ i = 2 : \quad & -\frac{u_1 - 2u_2 + u_3}{h^2} + c_2 u_2 = f_2, \\ & \dots \\ i = N : \quad & -\frac{u_{N-1} - 2u_N + u_{N+1}}{h^2} + c_N u_N = f_N, \end{aligned}$$

D'où le système

$$\left\{ \begin{array}{llll} -\frac{-2u_1 + u_2}{h^2} & + & c_1 u_1 & = & f_1 + \frac{u_0}{h^2}, \\ -\frac{u_1 - 2u_2 + u_3}{h^2} & + & c_2 u_2 & = & f_2, \\ \dots & \dots & \dots & \dots & \\ -\frac{u_{N-1} - 2u_N}{h^2} & + & c_N u_N & = & f_N + \frac{u_{N+1}}{h^2}, \end{array} \right.$$

Matriciellement, le problème s'écrit :

$$A_h u_h = b_h$$

(4)

avec

$$A_h = \frac{1}{h^2} \begin{pmatrix} 2 + h^2 c_1 & -1 & 0 & \cdots & 0 \\ -1 & 2 + h^2 c_2 & -1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -1 & 2 + h^2 c_{N-1} & -1 \\ 0 & \cdots & 0 & -1 & 2 + h^2 c_N \end{pmatrix}$$

$$u_h = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{n-1} \\ u_n \end{pmatrix}, \quad b_h = \begin{pmatrix} f_1 + \frac{g_0}{h^2} \\ f_2 \\ \vdots \\ f_{n-1} \\ f_n + \frac{g_1}{h^2} \end{pmatrix}$$

Exemple numérique

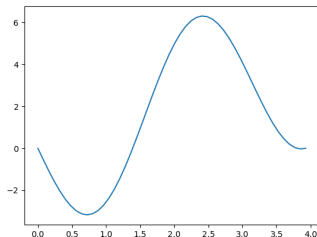
Soit à résoudre l'équation différentielle

$$\begin{cases} \frac{d^2 u(x)}{dx^2} + 4u(x) = 4x, & \text{pour } x \in]0, \frac{5\pi}{4}[\\ u(0) = 0, & u(\frac{5\pi}{4}) = 0, \end{cases} \quad (5)$$

La solution exacte est $y(x) = x - \frac{5\pi}{4} \sin(x)$.

$$\frac{1}{4h^3} \begin{bmatrix} -2 + 4h^2 & 1 & 0 & \cdots & 0 \\ 1 & -2 + 4h^2 & 1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & 1 & -2 + 4h^2 & 1 \\ 0 & \cdots & 0 & 1 & -2 + 4h^2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ \vdots \\ \vdots \\ N \end{bmatrix}$$

```
# Courbe
import numpy as np
import matplotlib.pyplot as plt
a = 0
b = 5 * np.pi / 4
t = np.linspace(a, b, 50)
y = t - 5 * np.pi / 4 * np.sin(2 * t)
plt.plot(t, y)
```



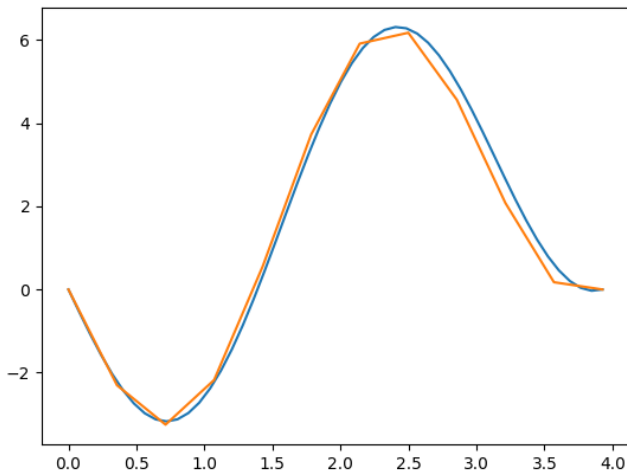

```

import numpy as np
import matplotlib.pyplot as plt

a = 0; b = 5 * np.pi / 4; t = np.linspace(a, b, 50)
y = t - 5 * np.pi / 4 * np.sin(2 * t)
plt.plot(t, y)

N = 10; h = (b - a) / (N + 1)
x = np.linspace(a, b, N + 2)
M = ((-2 + 4 * h ** 2) * np.diag(np.ones(N))
      + np.diag(np.ones(N - 1), -1)
      + np.diag(np.ones(N - 1), 1)) / (4*h**3)
F = np.arange(1, N+1, 1)
U = np.linalg.solve(M, F)
U = np.insert(U, N, 0)
U = np.insert(U, 0, 0)
plt.plot(x, U)
plt.show()

```



La matrice A_h est tridiagonale, on le décompose en deux matrices :

$$A_h = \frac{1}{h^2} A_h^{(0)} + C_h$$

$$A_h^{(0)} = \begin{pmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -1 & 2 & -1 \\ 0 & \cdots & 0 & -1 & 2 \end{pmatrix}$$

et

$$C_h = \begin{pmatrix} c_1 & 0 & \cdots & 0 \\ 0 & c_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & c_n \end{pmatrix}$$

On note que les conditions au bord $u_0 = g_0$ et $u_{n+1} = g_1$ n'apparaissent que dans le vecteur b_h .

proposition

Supposons $c \geq 0$. La matrice A_h est symétrique définie positive.

Démonstration La matrice A_h est clairement symétrique. Soit z un vecteur de \mathbb{R}^n , de composantes z_1, \dots, z_n . On a

$$\begin{aligned} z^t A_h z &= z^t A_h^{(0)} z + \sum_{i=1}^n c_i z_i^2 \geq z^t A_h^{(0)} z \\ &= \frac{z_1^2 + (z_1 - z_2)^2 + \dots + (z_{n-1} - z_n)^2 + z_n^2}{h^2} \end{aligned}$$

Cette quantité est positive, et non nulle si $z \neq 0$ (car au moins l'un des termes au numérateur est non nul).

Définition

$$A_h^{(0)} = \begin{pmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -1 & 2 & -1 \\ 0 & \cdots & 0 & -1 & 2 \end{pmatrix}$$

est la matrice du Laplacien discret.

Théorème (Principe du maximum discret)

$A_h V_h = F_h$ admet une solution unique V_h de plus si $f \geq 0$ alors $V_h \geq 0$

Lemme

$$\forall V \in \mathbb{R}^n \quad AV \geq 0 \implies V \geq 0$$

$$AV \geq 0 \implies \begin{cases} 2V_1 - V_2 \geq 0 \\ -V_{j-1} + 2V_j - V_{j+1} \geq 0 \\ -V_{n-1} + 2V_n \geq 0 \end{cases}$$

Soit $V_m = \inf_i(V_i)$

- $m = 1 : 2V_1 - V_2 \geq 0 \implies V_1 \geq V_2 - V_1 \geq 0$
- $m = n : -V_{n-1} + 2V_n \geq 0 \implies V_n \geq V_{n-1} - V_n \geq 0$
- $m \neq n$ et $m \neq 1 :$

$$\begin{aligned} -V_{m-1} + 2V_m - V_{m+1} \geq 0 &\implies (V_m - V_{m+1}) \geq (V_{m-1} - V_m) \geq 0 \\ \implies V_m = V_{m+1} = V_{m-1} &\implies V_{m-1} = \inf_i(V_i) \implies V_1 = \inf_i(V_i) \end{aligned}$$

$$AV = 0 \implies AV \geq 0 \text{ et } AV \leq 0 \implies V \geq 0 \text{ et } V \leq 0 \implies V = 0$$

Les coefficients de A sont tous positifs : si $A^{-1} = [C_1 \ C_2 \ \cdots \ C_n]$ alors

$$AA^{-1} = I \implies AC_i = e_i \implies C_i \geq 0 \implies A^{-1} \geq 0$$

$$\implies V = A^{-1}F \geq 0$$

Consistance

Définition

Supposons $f \in C^2([0, 1])$, donc $u \in C^4([0, 1])$. On pose

$$\begin{aligned} \Pi_h : C^0([0, 1]) &\rightarrow \mathbb{R}^n \\ \omega &\mapsto (\omega(x_1), \omega(x_2), \dots, \omega(x_N))^T \end{aligned}$$

On définit l'erreur de consistance par

$$\varepsilon_h = A_h \Pi_h u - F_h$$

On dit que le schéma est consistant si

$$\lim_{n \rightarrow \infty} \|\varepsilon_h\|_{\infty} = 0$$

Si

$$\|\varepsilon_h\|_{\infty} = O(h^p)$$

On dit que la méthode est d'ordre p .

Consistance

Proposition

Le schéma $A_h V_h = F_h$ est consistant d'ordre 2.

$$\begin{aligned}(\varepsilon_h)_j &= \frac{u(x_{j-1}) - 2u(x_j) + u(x_{j+1}))}{h^2} - f(x_j) \\ &= \frac{u(x_{j-1}) - 2u(x_j) + u(x_{j+1}))}{h^2} - u''(x_j) = O(h^2)\end{aligned}$$

$$\begin{aligned}&\left| u(x_j + h) - u(x_j) - hu'(x_j) - \frac{h^2}{2} u''(x_j) - \frac{h^3}{6} u'''(x_j) \right| \leq \frac{h^4}{24} \|u^{(4)}\|_\infty \\&-\frac{h^4}{24} \|u^{(4)}\|_\infty \leq u(x_j + h) - u(x_j) - hu'(x_j) - \frac{h^2}{2} u''(x_j) - \frac{h^3}{6} u'''(x_j) \leq \frac{h^4}{24} \|u^{(4)}\|_\infty \\&-\frac{h^4}{24} \|u^{(4)}\|_\infty \leq u(x_j - h) - u(x_j) + hu'(x_j) - \frac{h^2}{2} u''(x_j) + \frac{h^3}{6} u'''(x_j) \leq \frac{h^4}{24} \|u^{(4)}\|_\infty \\&\implies \|\varepsilon_h\|_\infty \leq \frac{h^2}{12} \|u^{(4)}\|_\infty\end{aligned}$$

Stabilité

Définition

On dit que le schéma numérique $A_h V_h = F_h$ est stable s'il existe une constante C telle que

$$\|V_h\| \leq C \|F_h\| \quad \forall F_h$$

Théorème

$$\|A_h^{-1}\| \leq \frac{1}{8} \quad \forall n \geq 1$$

Soit le problème $\begin{cases} -u'' = 1 \\ u(0) = u(1) = 0 \end{cases}$ dont la solution est $u(x) = \frac{x(1-x)}{2}$

Alors

$$F_h = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \Rightarrow A_h^{-1} F_h = \begin{pmatrix} \frac{h(1-h)}{2} \\ \frac{2h(1-2h)}{2} \\ \vdots \\ \frac{nh(1-nh)}{2} \end{pmatrix}$$

Donc

$$\|A_h^{-1}\| = \sup_{\|F_h\|=1} \|A_h^{-1}F_h\| = \frac{1}{8}$$

Définition

Soit $E_h = V_h - \Pi_h u$ On dit que le schéma numérique $A_h V_h = F_h$ converge si pour tout $f \in C^0([0, 1])$

$$\|E_h\|_{\infty} \rightarrow 0 \quad \text{quand } n \rightarrow \infty$$

Théorème

La méthode converge à l'ordre 2.

Démo : On note u la solution exacte

$$\begin{aligned} A_h E_h &= A_h V_h - A_h \Pi_h u \\ &= F_h - (\varepsilon_h + F_h) \\ &= -\varepsilon_h \end{aligned}$$

$$\text{Donc } E_h = -A_h^{-1} \varepsilon_h \implies \|E_h\| \leq \|A_h^{-1}\| \|\varepsilon_h\| \leq \frac{1}{8} \times \frac{h^2}{24} \|u^{(4)}\| \rightarrow 0$$

Convergence de la méthode

Afin d'étudier la convergence de la solution approchée u_h vers la solution exacte u lorsque $h \rightarrow 0$, on commence par étudier l'erreur de consistance :

definition

On appelle erreur de consistance du schéma $A_h u_h = b_h$, le vecteur $\epsilon_h(u)$ de \mathbb{R}^n défini par

$$\epsilon_h(u) = A_h(\pi_h(u)) - b_h, \quad \text{où } \pi_h = \begin{pmatrix} u(x_1) \\ u(x_2) \\ \vdots \\ u(x_n) \end{pmatrix}$$

$\pi_h(u)$ représente la projection de la solution exacte sur le maillage. On dit que le schéma est consistant pour la norme $\|\cdot\|$ de \mathbb{R}^n si $\lim_{h \rightarrow 0} \|\epsilon_h(u)\| = 0$. Si de plus il existe C indépendante de h telle que

$$\|\epsilon_h(u)\| \leq Ch^p$$

En utilisant (3), on obtient immédiatement

$$\|\epsilon_h(u)\|_\infty \leq \frac{h^2}{12} \sup_{y \in [0,1]} |u^{(4)}(y)|,$$

et donc

proposition

Supposons que la solution u du problème (1) est C^4 sur $[0, 1]$. Alors le schéma (4) est consistant d'ordre 2 pour la norme $\|\cdot\|_\infty$.

L'erreur de convergence est l'erreur entre la solution approchée u_h est la solution exacte aux points du maillage $\pi_h(u)$. Afin de majorer cette erreur, il suffit d'observer que, par définition de l'erreur de consistance et puisque $A_h u_h = b_h$,

$$u_h - \pi_h(u) = -(A_h)^{-1} \epsilon_h(u).$$

On peut montrer que $\|(A_h)^{-1}\| \leq \frac{1}{8}$ (admis), ce qui donne le résultat de convergence :

Theorem

On suppose $c \geq 0$. Si la solution u du problème (1) est C^4 sur $[0, 1]$, alors le schéma (4) est convergent d'ordre 2 pour la norme $\|\cdot\|_\infty$. Plus précisément,

$$\|u_h - \pi_h(u)\|_\infty \leq \frac{h^2}{96} \sup_{y \in [0,1]} |u^{(4)}(y)|.$$

On notera que, dans cet exemple, la consistance du schéma permet immédiatement d'obtenir la convergence. Ceci est particulier à cet exemple simple. En général, un autre ingrédient est nécessaire à la convergence : la stabilité du schéma (voir section 2.3).

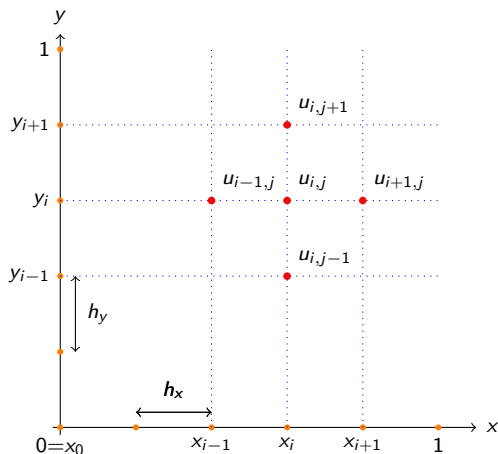
Le problème aux limites d'ordre 2 en dimension 2

Equation de Poisson $-\Delta u = f$ On considère le problème aux limites

$$\left\{ \begin{array}{l} -\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} = f(x, y), \quad (x, y) \in \Omega \\ u(x, y) = 0, \quad (x, y) \in \partial\Omega \end{array} \right. \quad (6)$$

où f est une fonction donnée sur $\Omega = [0, 1] \times [0, 1]$. Les deux ingrédients principaux d'une approximation par différences finies sont le schéma d'approximation des dérivées et la grille de discrétisation.

Approximation par différences finies



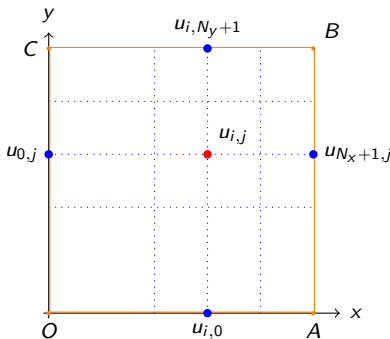
Les points de discrétisation du maillage : $M_{i,j} = (x_i, y_j) = (ih_x, jh_y)$

$$(i, j) \in \llbracket 0, N_x + 1 \rrbracket \times \llbracket 0, N_y + 1 \rrbracket, \quad h_x = \frac{1}{N_x + 1}, h_y = \frac{1}{N_y + 1}$$

Les points

$$\begin{cases} M_{0,j} \text{ et } M_{N_x+1,j} \text{ pour } j = 0, 1, \dots, N_y + 1 \\ M_{i,0} \text{ et } M_{i,N_y+1} \text{ pour } i = 0, 1, \dots, N_x + 1 \end{cases}$$

constituent le bord du domaine (les cotés du rectangle $OABC$), et les points $M_{i,j}$ où $(i,j) \in \llbracket 1, N_x \rrbracket \times \llbracket 1, N_y \rrbracket$ sont les points internes du maillage.



On cherche en chaque point $M_{i,j}$ une valeur approchée $u_{i,j} \simeq u(x_i, y_j)$.
Les valeurs $u_{i,j}$ sur les bord sont nulles (ou données par hypothèse).

Pour les sommets internes, on utilise l'approximation de la dérivée seconde décrite plus haut. On a

$$\left\{ \begin{array}{l} -\frac{\partial^2 u(x_i, y_j)}{\partial x^2} - \frac{\partial^2 u(x_i, y_j)}{\partial y^2} = f(x_i, y_j), \quad (x_i, y_j) \in]0, 1[\times]0, 1[\\ u(x_i, 0) = u(x_i, 1) = 0, \quad 0 \leq x_i \leq 1 \\ u(0, y_j) = u(1, y_j) = 0, \quad 0 \leq y_j \leq 1 \end{array} \right. \quad (7)$$

Or

$$\frac{\partial^2 u(x_i, y_j)}{\partial x^2} \simeq \frac{u(x_i - h, y_j) - 2u(x_i, y_j) + u(x_i + h, y_j)}{h_x^2} \simeq \frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{h_x^2}$$

$$\frac{\partial^2 u(x_i, y_j)}{\partial y^2} \simeq \frac{u(x_i, y_j - h) - 2u(x_i, y_j) + u(x_i, y_j + h)}{h_y^2} \simeq \frac{u_{i,j-1} - 2u_{i,j} + u_{i,j+1}}{h_y^2}$$

et on pose où $f_{i,j} = f(x_i, y_j)$.

D'où le schéma, pour $i \in \llbracket 0, N_x + 1 \rrbracket$ et $j \in \llbracket 0, N_y + 1 \rrbracket$:

$$\left\{ \begin{array}{l} \text{Pour } (i, j) \in \llbracket 1, N_x \rrbracket \times \llbracket 1, N_y \rrbracket \\ - \frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{h_x^2} - \frac{u_{i,j-1} - 2u_{i,j} + u_{i,j+1}}{h_y^2} = f_{i,j} \\ \text{et} \\ u_{i,0} = u_{i,N_y+1} = 0, \quad i \in \llbracket 0, N_x + 1 \rrbracket \\ u_{0,j} = u_{N_x+1,j} = 0, \quad j \in \llbracket 0, N_y + 1 \rrbracket \end{array} \right. \quad (8)$$

On obtient $N_x \times N_y$ équations à $N_x \times N_y$ inconnues

$$u_{1,1}; u_{1,2}; \dots, u_{1,N_y}; u_{2,1}; u_{2,2} \dots u_{2,N_y} \dots \dots u_{N_x,1}; u_{N_x,2} \dots u_{N_x,N_y}$$

$$i = 1, j = 1 : \quad -\frac{u_{0,1} - 2u_{1,1} + u_{2,1}}{h_x^2} - \frac{u_{1,0} - 2u_{1,1} + u_{1,2}}{h_y^2} = f_{1,1}$$

$$i = 1, j = 2 : \quad -\frac{u_{0,2} - 2u_{1,2} + u_{2,2}}{h_x^2} - \frac{u_{1,1} - 2u_{1,2} + u_{1,3}}{h_y^2} = f_{1,2}$$

...

$$i = 1, j = N_y : \quad -\frac{u_{0,N_y} - 2u_{1,N_y} + u_{2,N_y}}{h_x^2} - \frac{u_{1,N_y-1} - 2u_{1,N_y} + u_{1,N_y+1}}{h_y^2} = f_{1,N_y}$$

$$i = 2, j = 1 : \quad -\frac{u_{1,1} - 2u_{2,1} + u_{3,1}}{h_x^2} - \frac{u_{2,0} - 2u_{2,1} + u_{2,2}}{h_y^2} = f_{2,1}$$

$$i = 2, j = 2 : \quad -\frac{u_{1,2} - 2u_{2,2} + u_{3,2}}{h_x^2} - \frac{u_{2,1} - 2u_{2,2} + u_{2,3}}{h_y^2} = f_{2,2}$$

...

$$i = 2, j = N_y : \quad -\frac{u_{1,N_y} - 2u_{2,N_y} + u_{3,N_y}}{h_x^2} - \frac{u_{2,N_y-1} - 2u_{2,N_y} + u_{2,N_y+1}}{h_y^2} = f_{2,N_y}$$

...

$$i = N_x, j = N_y : \quad -\frac{u_{N_x-1,N_y} - 2u_{N_x,N_y} + u_{N_x+1,N_y}}{h_x^2} - \frac{u_{N_x,N_y-1} - 2u_{N_x,N_y} + u_{N_x,N_y+1}}{h_y^2} = f_{N_x,N_y}$$

D'où le système, en faisant $N_x = N_y = n$ et donc $h_x = h_y = h$:

$$\left\{ \begin{array}{l} 4u_{1,1} - u_{1,2} - u_{2,1} = h^2 f_{1,1} \\ -u_{1,1} + 4u_{1,2} - u_{1,3} - u_{2,2} = h^2 f_{1,2} \\ \dots \\ -u_{1,n-1} + 4u_{1,n} - u_{2,n} = h^2 f_{1,n} \\ \hline -u_{1,1} + 4u_{2,1} - u_{2,2} - u_{3,1} = h^2 f_{2,1} \\ -u_{1,2} + 4u_{2,2} - u_{2,3} - u_{3,2} = h^2 f_{2,2} \\ \dots \\ -u_{1,n} - u_{2,n-1} + 4u_{2,n} - u_{3,n} = h^2 f_{2,n} \\ \hline \vdots \\ \hline -u_{n-1,1} + 4u_{n,1} - u_{n,2} = h^2 f_{n,1} \\ -u_{n-1,2} + 4u_{n,2} - u_{n,3} = h^2 f_{n,2} \\ \dots \\ -u_{n-1,n} - u_{n,n-1} + 4u_{n,n} = h^2 f_{n,n} \end{array} \right.$$

Matriciellement, le problème s'écrit :

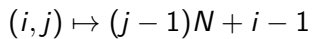
$$A_h u_h = b_h$$

(9)

avec

$$\begin{bmatrix} 4 & -1 & 0 & 0 & -1 & 0 & 0 & 0 \\ -1 & 4 & -1 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 4 & -1 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 4 & 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 & 4 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & -1 & 4 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & -1 & 4 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & -1 & 4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\left[\begin{array}{ccc|ccc|ccc}
 4 & -1 & 0 & -1 & 0 & 0 & & & \\
 -1 & 4 & -1 & 0 & -1 & 0 & & & \\
 0 & -1 & 4 & 0 & 0 & -1 & & & \\
 \hline
 -1 & & & 4 & -1 & 0 & -1 & 0 & 0 \\
 & -1 & & -1 & 4 & -1 & 0 & -1 & 0 \\
 & & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\
 \hline
 & & & -1 & & & 4 & -1 & 0 \\
 & & & & -1 & & -1 & 4 & -1 \\
 & & & & & -1 & 0 & -1 & 4
 \end{array} \right] \begin{bmatrix} u_{11} \\ u_{12} \\ u_{13} \\ \hline u_{21} \\ u_{22} \\ u_{23} \\ \hline u_{31} \\ u_{32} \\ u_{33} \end{bmatrix} = h^2 \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ \hline f_{21} \\ f_{22} \\ f_{23} \\ \hline f_{31} \\ f_{32} \\ f_{33} \end{bmatrix}$$




```

import matplotlib.pyplot as plt
from matplotlib import cm
import numpy as np

N = 3; h = 1 / (N + 1)
# La matrice

M = np.diag(4 * np.ones(N * N)) \
    - np.diag(np.ones(N * N - 1), -1) \
    - np.diag(np.ones(N * N - 1), 1) \
    - np.diag(np.ones(N * N - N), -N) \
    - np.diag(np.ones(N * N - N), N)

for k in range(N - 1):
    M[(k + 1) * N, (k + 1) * N - 1] = 0
    M[(k + 1) * N - 1, (k + 1) * N] = 0

F = h * h * np.ones(N * N)
U = np.linalg.solve(M, F)

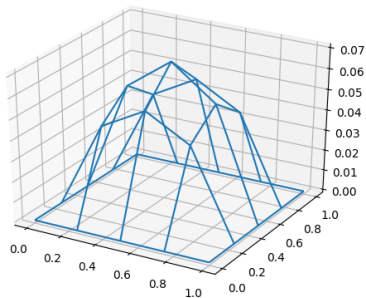
```

```

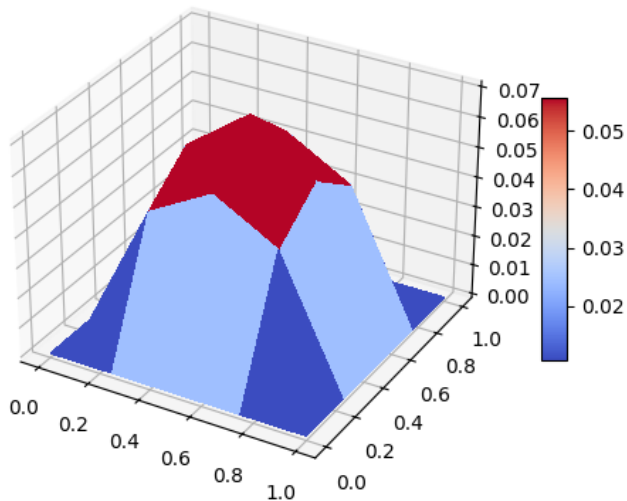
fig = plt.figure()
ax = fig.add_subplot(projection = '3d')
# Make data.
X = np.linspace(0, 1, N + 2)
Y = np.linspace(0, 1, N + 2)
X, Y = np.meshgrid(X, Y)
Z = np.zeros((N + 2, N + 2))
for i in range(1, N + 1):
    for j in range(1, N + 1):
        Z[i, j] = U[(i - 1) * N + j - 1]
# Plot the surface.
surf = ax.plot_surface(X, Y, Z, rstride=1, cstride=1,
                        cmap=cm.coolwarm,linewidth=0, antialiased=False)
fig.colorbar(surf, shrink=0.5, aspect=10)
plt.show()

```

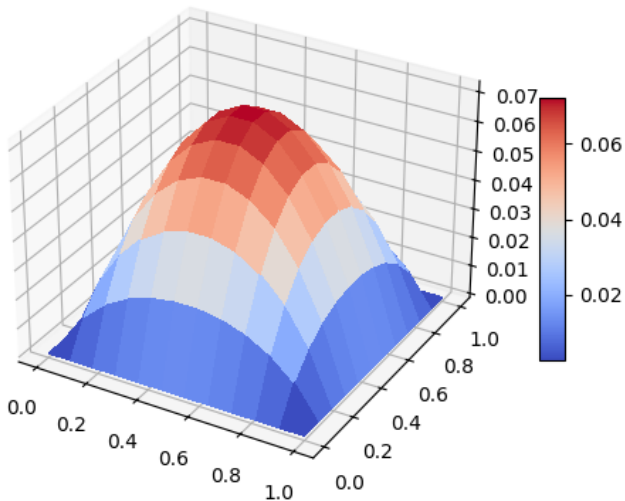
$$N = 3$$



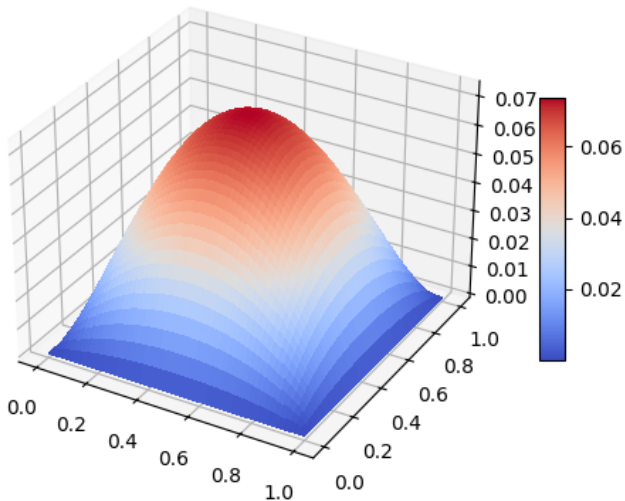
$$N = 3$$



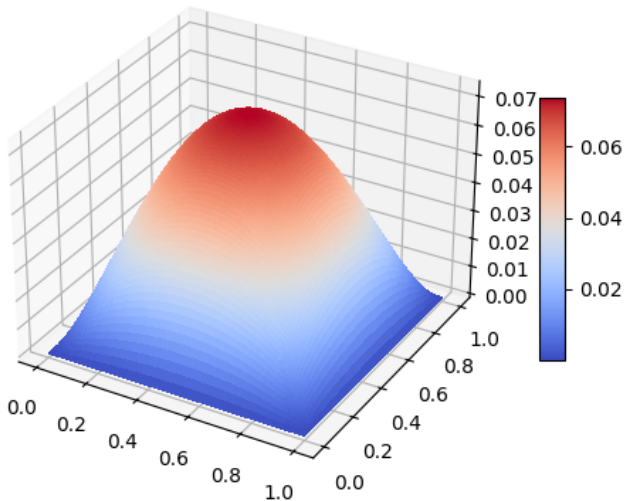
$N = 10$



$N = 50$



$N = 100$



les schémas implicites et explicites

On s'intéresse au problème

$$\begin{cases} \frac{\partial u}{\partial t}(t, x) - \frac{\partial^2 u}{\partial x^2}(t, x) = f(t, x), \text{ pour } (t, x) \in [0, T] \times [0, 1] \\ u(0, x) = u_0(x), \text{ pour } x \in [0, 1] \\ u(t, 0) = 0 \text{ et } u(t, 1) = 0, \text{ pour } t \geq 0, \end{cases}$$

où l'on a pris pour simplifier des conditions au bord nulles. Il s'agit de l'équation de la chaleur en dimension 1 (d'espace), qui est un problème parabolique en dimension 2 (temps et espace). Cet exemple est typique de la situation générale des problèmes paraboliques. On distingue deux grandes familles d'approximations par différences finies : les schémas explicites et les schémas implicites.

La discétisation

On va se limiter au cas le plus simple du maillage régulier : soient n , m deux entiers fixés. On pose

$$x_i = ih, \quad \forall i \in \{0, 1, \dots, n+1\}, \quad \text{où } h = \frac{1}{n+1}$$

$$t_j = j\Delta t, \quad \forall j \in \{0, 1, \dots, m\}, \quad \text{où } \Delta t = \frac{1}{m}.$$

En particulier, $x_0 = 0$, $x_{n+1} = 1$, $t_0 = 0$ et $t_m = T$. Les points (t_j, x_i) sont alors les points d'intersection d'une "grille" régulière en espace-temps.

L'approximation par différences finies consiste alors à chercher une approximation, notée $u_i^{(j)}$, de $u(t_j, x_i)$ (notez que l'indice en temps est en exposant, et l'indice en espace en indice, comme précédemment).

Les valeurs approchées aux points de maillage au bord du domaine et en $t = 0$ sont données par la valeur exacte (donnée) de la fonction u :

$$u_0^{(j)} = u_{n+1}^{(j)} = 0, \quad \forall j \in \{0, \dots, m\} \quad (10)$$

$$u_i^{(0)} = u_0(x_i), \quad \forall i \in \{0, \dots, n+1\} \quad (11)$$

Ceci laisse nm inconnues à déterminer (les $u_i^{(j)}$ pour $1 \leq i \leq n$ et $1 \leq j \leq m$).

On a déjà vu que le terme de dérivée seconde pouvait être approché avec

$$\frac{\partial^2 u}{\partial x^2}(x_i, t_j) \simeq \frac{u(x_{i+1}, t_j) - 2u(x_i, t_j) + u(x_{i-1}, t_j))}{h^2} \simeq \frac{u_{i+1}^{(j)} - 2u_i^{(j)} + u_{i-1}^{(j)}}{h^2}$$

(schéma à trois points pour la dérivée seconde). Plusieurs choix sont possibles pour l'approximation de la dérivée en temps. Chacun de ces choix conduit à une famille de schémas distincte.

Le schéma explicite

La première possibilité est d'utiliser l'approximation décentrée à droite

$$\frac{\partial u}{\partial t}(x_i, t_j) \simeq \frac{u(x_i, t_{j+1}) - u(x_i, t_j)}{\Delta t} \simeq \frac{u_i^{(j+1)} - u_i^{(j)}}{\Delta t}$$

On obtient alors le schéma suivant :

$$\frac{u_i^{(j+1)} - u_i^{(j)}}{\Delta t} - \frac{u_{i+1}^{(j)} - 2u_i^{(j)} + u_{i-1}^{(j)}}{h^2} = f(x_i, t_j), \quad \forall i \in \{1, \dots, n\}, \forall j \in \{0, \dots, m\}$$

soit $n \times m$ équations pour $n \times m$ inconnues. On note que les conditions aux limites (8) et (9) doivent être connues pour résoudre ces équations, et que l'indice de temps j doit varier entre 0 et m .

Il est commode de réécrire ce système vectoriellement : on introduit la notation

$$U^{(j)} = \begin{pmatrix} u_1^{(j)} \\ u_2^{(j)} \\ \vdots \\ u_n^{(j)} \end{pmatrix}, \quad \forall j \in \{0, \dots, m\}$$

Le vecteur $U^{(0)}$ est donné par les conditions initiales, et le schéma précédent s'écrit :

$$\frac{U^{(j+1)} - U^{(j)}}{\Delta t} + A_h^{(0)} U^{(j)} = C^{(j)}, \quad \forall j \in \{0, \dots, m\} \quad (12)$$

où la matrice $A_h^{(0)}$ a été définie dans (6), et

$$C^{(j)} = \begin{pmatrix} f(x_1, t^{(j)}) \\ f(x_2, t^{(j)}) \\ \vdots \\ f(x_n, t^{(j)}) \end{pmatrix}, \quad \forall j \in \{0, \dots, m\}$$

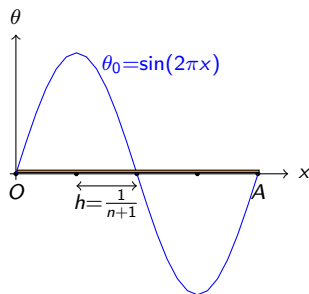
Ce système peut également se réécrire sous la forme

$$U^{(j+1)} = (I_d - \Delta t A_h^{(0)}) U^{(j)} + \Delta t C^{(j)}, \quad \forall j \in \{0, \dots, m-1\} \quad (13)$$

où on utilise la notation I_d pour la matrice identité.

Cette équation justifie le nom explicite pour ce schéma, puisqu'il permet de calculer la valeur approchée au temps t_{j+1} par simple produit de matrice avec la valeur approchée au temps t_j . En particulier, aucune inversion de matrice ou résolution de système linéaire n'est nécessaire pour le calcul.

Exemple



$$\Delta t = 0.01$$

$$\begin{cases} U^{(j+1)} = \left[I_n - \frac{\Delta t}{h^2} \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix} \right] U^{(j)} \\ U^{(0)} = \begin{pmatrix} 100 \\ 0 \\ -100 \end{pmatrix} \end{cases}$$

```
import matplotlib.pyplot as plt
import numpy as np
```

```
N = 50;
```

```
M = 5;
```

```
h = 1 / (N + 1);
```

```
dt = 0.01
```

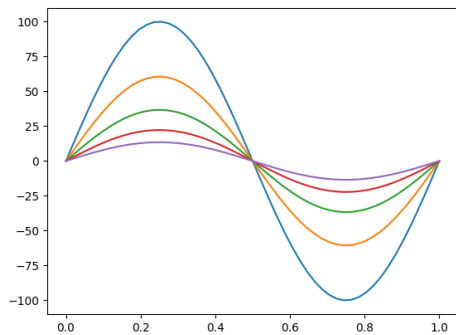
```
x = np.linspace(0, 1, N + 2)
```

```
# La matrice
```

```
A = np.diag(2 * np.ones(N)) \
    - np.diag(np.ones(N - 1), -1) \
    - np.diag(np.ones(N - 1), 1)
```

```
# Theta initiale
U = 100 * np.sin(2*np.pi*x)
plt.plot(x, U)
for k in range(1, M):
    U[1:-1] = np.dot((np.eye(N) - dt / (h ** 2) * A), U[1:-1])
    plt.plot(x, U)

plt.savefig('chaleurExplicite.png')
plt.show()
```

Le schéma implicite

On aurait pu approcher la dérivée partielle en temps par l'approximation décentrée à gauche

$$\frac{\partial u}{\partial t}(x_i, t_j) \simeq \frac{u(x_i, t_j) - u(x_i, t_{j-1})}{\Delta t} \simeq \frac{u_i^{(j)} - u_i^{(j-1)}}{\Delta t}$$

On obtient alors le schéma suivant :

$$\frac{u_i^{(j)} - u_i^{(j-1)}}{\Delta t} - \frac{u_{i+1}^{(j)} - 2u_i^{(j)} + u_{i-1}^{(j)}}{h^2} = f(x_i, t_j), \quad \forall i \in \{1, \dots, n\}, \forall j \in \{1, \dots, m\}$$

couplé aux mêmes conditions initiales que le schéma explicite (noter que cette fois l'indice j varie de 1 à m). Vectoriellement, on obtient

$$\frac{U^{(j)} - U^{(j-1)}}{\Delta t} + A_h^{(0)} U^{(j)} = C^{(j)}, \quad \forall j \in \{1, \dots, m\} \quad (14)$$

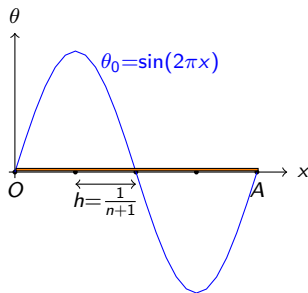
où les $C^{(j)}$ sont définis comme plus haut, ce qui se réécrit

$$U^{(j)} = (I_d + \Delta t A_h^{(0)})^{-1} \left(U^{(j-1)} + \Delta t C^{(j)} \right), \quad \forall j \in \{1, \dots, m\} \quad (15)$$

On remarque que la matrice $(I_d + \Delta t A_h^{(0)})$ est symétrique définie positive, et donc inversible, puisque $A_h^{(0)}$ est symétrique définie positive.

Ce schéma est dit implicite puisque, contrairement au schéma explicite, sa résolution nécessite la résolution d'un système linéaire à chaque pas de temps (ou le calcul initial de l'inverse de la matrice $(I_d + \Delta t A_h^{(0)})$, utilisé ensuite à chaque pas de temps). La résolution du schéma implicite est donc plus coûteuse que le schéma explicite. Cependant, comme on le verra plus loin, ce coût en temps de calcul est largement compensé par la meilleure stabilité du schéma.

Exemple



$$\Delta t = 0.01$$

$$\left\{ \begin{array}{l} U^{(j+1)} = \left[I_n + \frac{\Delta t}{h^2} \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix} \right]^{-1} U^{(j)} \\ U^{(0)} = \begin{pmatrix} 100 \\ 0 \\ -100 \end{pmatrix} \end{array} \right.$$

```
import matplotlib.pyplot as plt
import numpy as np
```

```
N = 50;
```

```
M = 5;
```

```
h = 1 / (N + 1);
```

```
dt = 0.05
```

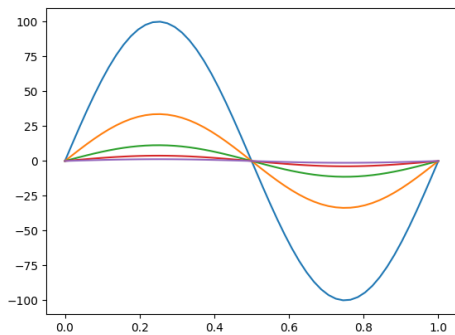
```
x = np.linspace(0, 1, N + 2)
```

```
# La matrice
```

```
A = np.diag(2 * np.ones(N)) \
    - np.diag(np.ones(N - 1), -1) \
    - np.diag(np.ones(N - 1), 1)
```

```
# Theta initiale
U = 100 * np.sin(2*np.pi*x)
plt.plot(x, U)
for k in range(1, M):
    U[1:-1] = np.linalg.solve((np.eye(N) + dt / (h ** 2) * A),
                               U[1:-1])
    plt.plot(x, U)

plt.savefig('chaleurImplicite.png')
plt.show()
```



Le schéma saute-mouton

Tous les schémas imaginables ne sont pas nécessairement bons, comme le montre l'exemple suivant : on pourrait chercher à améliorer la précision en temps en utilisant l'approximation d'ordre 2 (voir section 2.1.1) suivante de la dérivée en temps :

$$\frac{\partial u}{\partial t}(x_i, t_j) \simeq \frac{u(x_i, t_{j+1}) - u(x_i, t_{j-1})}{2\Delta t} \simeq \frac{u_i^{(j+1)} - u_i^{(j-1)}}{2\Delta t}$$

On obtiendrait alors le schéma

$$\frac{U^{(j+1)} - U^{(j-1)}}{2\Delta t} + A_h^{(0)} U^{(j)} = C^{(j)}, \quad \forall j \in \{2, \dots, m-1\} \quad (16)$$

qui est explicite en temps, puisque $U^{(j+1)}$ s'exprime directement en fonction de $U^{(j)}$ et $U^{(j-1)}$. On note qu'il est nécessaire pour ce schéma de connaître $U(0)$ et $U(1)$. Le précalcul de $U^{(1)}$ peut par exemple être fait en utilisant un pas en temps de l'une des méthodes précédentes.

Ce schéma, appelé schéma saute-mouton ou schéma de Richardson, est à la fois explicite et a priori d'ordre 2 en temps. Cependant, comme on le verra plus loin, il est toujours instable, et donc numériquement inutilisable.

Consistance, stabilité et convergence

Les trois schémas précédents peuvent s'écrire sous la forme générale

$$B_1 U^{(j+1)} + B_0 U^{(j)} + B_{-1} U^{(j-1)} = C^{(j)}$$

avec B_1 inversible ou $B_1 = 0$ et B_0 inversible. Pour le schéma explicite, on a

$$B_1 = \frac{1}{\Delta t} Id, \quad B_0 = -\frac{1}{\Delta t} Id + A_h^{(0)}, \quad B_{-1} = 0.$$

Pour le schéma implicite, on a

$$B_1 = 0, \quad B_0 = \frac{1}{\Delta t} Id + A_h^{(0)}, \quad B_{-1} = -\frac{1}{\Delta t} Id.$$

Pour le schéma suate-mouton, on a

$$B_1 = \frac{1}{\Delta t} Id, \quad B_0 = A_h^{(0)}, \quad B_{-1} = -\frac{1}{\Delta t} Id.$$

L'étude de la convergence de ces schémas est basée sur les propriétés de consistance et stabilité, définies ci-dessous pour le schéma général . On pourrait bien sûr imaginer des schémas faisant intervenir des indices en temps plus grands que $j + 1$ et plus petits que $j - 1$. Les définitions

Définition

On appelle erreur de consistance à l'instant t_j :

$$\varepsilon_h(u)^{(j)} = B_1(\pi_h(u))(t_{j+1}) + B_0(\pi_h(u))(t_j) + B_{-1}(\pi_h(u))(t_{j-1}) - C^{(j)},$$

où

$$\pi_h(u)(t) = \begin{pmatrix} u(t, x_1) \\ \vdots \\ u(t, x_N) \end{pmatrix}$$

On dit que le schéma est consistant pour la norme $\|\cdot\|$ de \mathbb{R}^N si

$$\sup_{0 \leq j \leq M+1} \|\varepsilon_h(u)^{(j)}\| \rightarrow 0, \quad \text{quand} \quad \Delta t \rightarrow 0 \text{ et } h \rightarrow 0.$$

Si de plus il existe $C > 0$, $p > 0$ et $q > 0$ indépendants de Δt et h tels que

$$\sup_{0 \leq j \leq M+1} \|\varepsilon_h(u)^{(j)}\| \leq C [(\Delta t)^p + h^q],$$

On dit que le schéma est consistant d'ordre p en temps et q en espace pour la norme $\|\cdot\|$.

Proposition

Supposons que la solution u au problème de la chaleur est C^2 par rapport à la variable t et C^4 par rapport à la variable x . Alors les schémas explicites et implicites sont consistants d'ordre 1 en temps et 2 en espace. Si de plus u est C^3 par rapport à la variable t , alors le schéma saute-mouton est consistant d'ordre 2 en temps et en espace.

On a $f(t_j, x_i) = \frac{\partial u}{\partial t}(t_j, x_i) - \frac{\partial^2 u}{\partial x^2}(t_j, x_i)$, On pose

$$\varepsilon_h(u)^{(j)} = E_i - F_i$$

où

$$E_i = \frac{u(t_{j+1}, x_i) - u(t_j, x_i)}{\Delta t} - \frac{\partial u}{\partial t}(t_j, x_i),$$
$$F_i = \frac{u(t_j, x_{i+1}) - 2u(t_j, x_i) + u(t_j, x_{i-1}))}{h^2} - \frac{\partial^2 u}{\partial x^2}(t_j, x_i)$$

Par développement de Taylor :

$$u(t_{j+1}, x_i) = u(t_j, x_i) + \Delta t \frac{\partial u}{\partial t}(t_j, x_i) + \frac{\Delta t^2}{2} \frac{\partial^2 u}{\partial t^2}(\theta, x_i)$$

D'où

$$E_i = \frac{\Delta t}{2} \frac{\partial^2 u}{\partial t^2}(\theta, x_i)$$

De même

$$F_i = \frac{h^2}{2} \left(\frac{\partial^4 u}{\partial t^4}(t_j, \xi_1) + \frac{\partial^4 u}{\partial t^4}(t_j, \xi_2) \right)$$

Définition

On dit que le schéma

$$B_1 U^{(j+1)} + B_0 U^{(j)} + B_{-1} U^{(j-1)} = C^{(j)}$$

est convergent ssi

$$\sup_{0 \leq j \leq M+1} \|U^{(j)} - (\pi_h u)(t_j)\| \rightarrow 0 \quad \text{quand } \Delta t \text{ et } h \rightarrow 0$$

Définition

On dit que le schéma

$$B_1 U^{(j+1)} + B_0 U^{(j)} + B_{-1} U^{(j-1)} = C^{(j)}$$

est stable ssi il existe deux constante $C_1(T)$ et $C_2(T)$ telles que

$$\max_{0 \leq j \leq M+1} \|U^{(j)}\| \leq C_1(T) \|U^{(0)}\| + C_1(T) \max_{0 \leq j \leq M+1} \|C^{(j)}\|$$

Théorème de Lax

Le schéma

$$B_1 U^{(j+1)} + B_0 U^{(j)} + B_{-1} U^{(j-1)} = C^{(j)}$$

converge ssi il est stable et consistant.

Proposition

Sous la condition

$$\frac{\Delta t}{h^2} \leq \frac{1}{2}$$

Le schéma explicite converge.

Équation d'advection

$$\begin{cases} \frac{\partial u}{\partial t} + V \frac{\partial u}{\partial x} = 0 & \forall (x, t) \in [0, 1] \times \mathbb{R}^+ \\ u(x+1, t) = u(x, t) & \forall (x, t) \in \mathbb{R} \times \mathbb{R}^+ \\ u(x, t=0) = u_0(x) & \forall (x, t) \in [0, 1] \end{cases}$$

On va étudier l'équation aux dérivées partielles suivante : où V est un réel non nul.

Remarque

Cette équation, aussi appelée équation de transport, intervient par exemple dans l'étude du transport d'un polluant dans un courant d'un fluide.

V est alors à interpréter comme la vitesse du courant. Noter qu'on peut avoir $V > 0$ ou $V < 0$ en fonction du sens du courant.

Équation d'advection

Dans le cas de ce problème simple, l'expression de la solution est connue. La solution s'écrit

$$u(x, t) = u_0(x - Vt)$$

En effet, on aura $\frac{\partial u}{\partial x} = u'_0(x - Vt)$ et $\frac{\partial u}{\partial t} - Vu'_0(x - Vt) = 0$ et donc bien $\frac{\partial u}{\partial t} + V \frac{\partial u}{\partial x} = 0$

Remarque

On a choisi ici des conditions aux limites de périodicité, mais on aurait pu envisager d'autres types de conditions aux limites. Il faudrait en tenir compte pour le traitement des équations aux bords de l'intervalle d'espace.

Équation d'advection

Discrétisation Posons $h = \frac{1}{N}$ pour discrétiser l'intervalle d'espace.

On a donc $x_j = jh$ et en particulier $x_0 = 0$, $x_N = 1$.

On pose $t_n = n\tau$ avec τ le pas de temps.

Ainsi, on posera comme inconnue discrète à chaque pas de temps le vecteur

$$u^{(n)} = \begin{pmatrix} u_0^{(n)} \\ u_1^{(n)} \\ \vdots \\ u_{N-1}^{(n)} \end{pmatrix}$$

La condition de périodicité se traduira par $u_N^{(n)} = u_0^{(n)}$.

On considère aussi $u_{N-1}^{(n)} = u_{-1}^{(n)}$.

Équation d'advection : Discrétisation

Posons $h = \frac{1}{N}$ pour discrétiser l'intervalle d'espace.

On a donc $x_j = jh$ et en particulier $x_0 = 0$, $x_N = 1$.

On pose $t_n = n\tau$ avec τ le pas de temps.

Ainsi, on posera comme inconnue discrète à chaque pas de temps le vecteur

$$u^{(n)} = \begin{pmatrix} u_0^{(n)} \\ u_1^{(n)} \\ \vdots \\ u_{N-1}^{(n)} \end{pmatrix}$$

La condition de périodicité se traduira par $u_N^{(n)} = u_0^{(n)}$.

On considère aussi $u_{N-1}^{(n)} = u_{-1}^{(n)}$.

Équation d'advection : Schéma explicite centré

$$\frac{u_j^{(n+1)} - u_j^{(n)}}{\tau} - \frac{u_{j+1}^{(n)} - u_{j-1}^{(n)}}{2h} = 0 \quad (n \geq 0, 0 \leq j \leq N-1)$$

Un schéma possible serait le schéma explicite centré :

$$u_j^{(n+1)} = \frac{c}{2} u_{j-1}^{(n)} + u_j^{(n)} - \frac{c}{2} u_{j+1}^{(n)} \quad \text{avec } c = \frac{V\tau}{h}$$

Ce schéma pourrait se formuler ainsi :

Équation d'advection : Schéma explicite centré

Et donc une formulation matricielle de la forme

$$u^{(n+1)} = Mu^{(n)}$$

avec

$$M = \begin{pmatrix} 1 & -\frac{c}{2} & 0 & 0 & 0 & \cdots & \frac{c}{2} \\ \frac{c}{2} & 1 & -\frac{c}{2} & 0 & 0 & \cdots & 0 \\ 0 & \frac{c}{2} & 1 & -\frac{c}{2} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ -\frac{c}{2} & 0 & 0 & 0 & 0 & \cdots & 1 \end{pmatrix}$$

en tenant de la condition de périodicité.

Propriétés du schéma explicite centré

Lemme

Le schéma explicite centré est consistant avec l'équation d'advection ci-dessus, précis à l'ordre 1 en temps et 2 en espace,

Équation d'advection : Exemple

On considère l'équation d'advection suivante :

$$\begin{cases} \frac{\partial u}{\partial t} + 3 \frac{\partial u}{\partial x} = 0, & 0 \leq x \leq 1, \quad t \in \mathbb{R}^+ \\ u(x+1, t) = u(x, t) \\ u(x, 0) = u_0(x) = \cos(\pi x)^8 \end{cases}$$

```
import matplotlib.pyplot as plt
import numpy as np

# Initialisation des constantes
h=0.005; tau=0.0002
nu = 3.0;c = nu * tau / h
N = int(1 / h)
nmax = int(1 / tau)

# Coordonnées en espace
X = np.linspace(0, 1, N + 2)[: -1]

# Initialisation temporelle
t = np.linspace(0, 1, nmax + 1)
```

```

# Initialisation de la solution
U = np.zeros((nmax + 1, N + 1))
U[0, :] = (np.cos(np.pi * X)) ** 8
# Matrice de taille N-N
M=np.eye(N+1)+c/2*np.diag(np.ones(N),1)
                                -c/2*np.diag(np.ones(N),-1)
M[0,N]=-c/2
M[N,0]=c/2
for n in range(nmax):
    U[n + 1,:] = np.dot(M, U[n,:])
K=nmax//10
for i in range(nmax//K):
    plt.plot(X, U[K*i,:], label = 'explicite centre')
plt.show()

```