

Ingénieurs Energétique 1ère année

Méthodes numériques (S6) - E.D.P.

François Cuvelier

Laboratoire d'Analyse Géométrie et Applications
Institut Galilée
Université Paris XIII.

2022/02/21

Chapitre IV

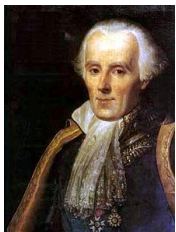
Résolution numérique des E.D.P.

- 1 Exemples d'E.D.P.
- 2 Méthodes de résolution numérique d'EDP

- 3 Méthode des différences finies 1D
- 4 Problème modèle évolutif

- 1 Exemples d'E.D.P.
 - Equation de Laplace/Poisson
 - Equation de la chaleur
 - Equation des ondes
- 2 Méthodes de résolution numérique d'EDP
- 3 Méthode des différences finies 1D
 - EDP stationnaire 1D + Dirichlet
 - EDP stationnaire + CL mixtes
- 4 Problème modèle évolutif
 - Schéma explicite
 - Schéma implicite

Equation de Laplace et équation de Poisson



Pierre-Simon Laplace 1749-1827, mathématicien, astronome, physicien et homme politique français



Siméon Denis Poisson 1781-1840, mathématicien, géomètre et physicien français

$$-\Delta u = f, \quad \text{dans } \Omega \subset \mathbb{R}^n \quad (4.1)$$

où Δ est l'opérateur laplacien : $\Delta u = \frac{\partial^2 u}{\partial x_1^2} + \dots + \frac{\partial^2 u}{\partial x_n^2}$.

Equation de Laplace si $f = 0$, sinon équation de Poisson.

$$-\Delta u = f, \quad \text{dans } \Omega \subset \mathbb{R}^n$$

- **Dirichlet** si on impose, sur une partie de $\partial\Omega$,

$$u = g, \quad \text{sur } \Gamma_D \subset \partial\Omega. \quad (4.2)$$

- **Neumann** si on impose sur une partie de $\partial\Omega$,

$$\frac{\partial u}{\partial n} = g, \quad \text{sur } \Gamma_N \subset \partial\Omega. \quad (4.3)$$

où $\frac{\partial u}{\partial n} = \langle \mathbf{grad} u, \mathbf{n} \rangle$ avec \mathbf{n} normale extérieure unitaire à Ω

- **Robin** si on impose sur une partie de $\partial\Omega$

$$\frac{\partial u}{\partial n} + \alpha u = g, \quad \text{sur } \Gamma_R \subset \partial\Omega. \quad (4.4)$$



Problème de condensateur en 2D

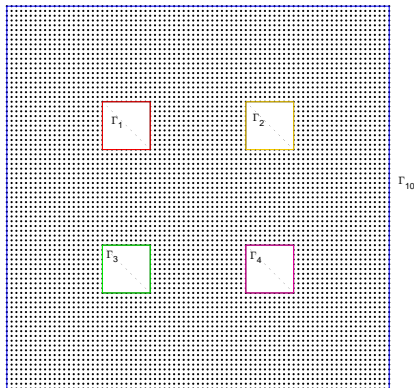
Find $u : \Omega \rightarrow \mathbb{R}$ such that

$$-\Delta u = 0 \text{ in } \Omega \subset \mathbb{R}^2,$$

$$u = 0 \text{ on } \Gamma_{10},$$

$$u = -1 \text{ on } \Gamma_2 \cup \Gamma_3,$$

$$u = 1 \text{ on } \Gamma_1 \cup \Gamma_4,$$





Problème de condensateur en 2D

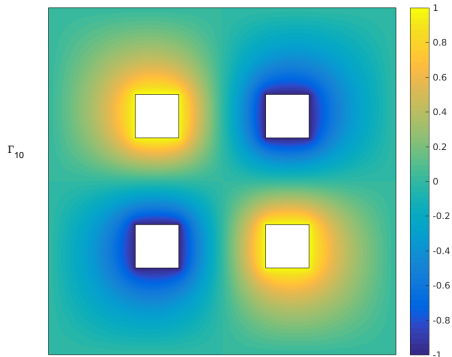
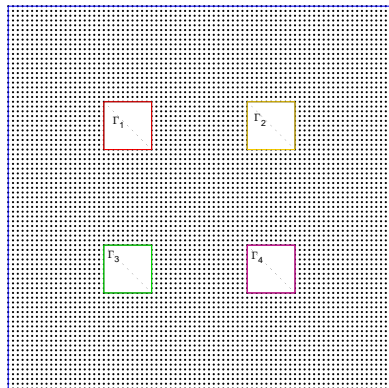
Find $u : \Omega \rightarrow \mathbb{R}$ such that

$$-\Delta u = 0 \text{ in } \Omega \subset \mathbb{R}^2,$$

$$u = 0 \text{ on } \Gamma_{10},$$

$$u = -1 \text{ on } \Gamma_2 \cup \Gamma_3,$$

$$u = 1 \text{ on } \Gamma_1 \cup \Gamma_4,$$





Problème de condensateur en 2D

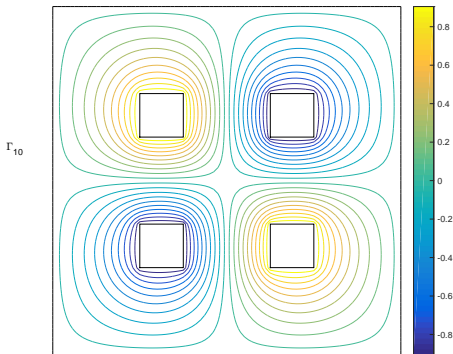
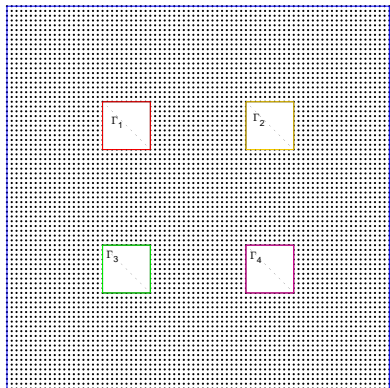
Find $u : \Omega \rightarrow \mathbb{R}$ such that

$$-\Delta u = 0 \text{ in } \Omega \subset \mathbb{R}^2,$$

$$u = 0 \text{ on } \Gamma_{10},$$

$$u = -1 \text{ on } \Gamma_2 \cup \Gamma_3,$$

$$u = 1 \text{ on } \Gamma_1 \cup \Gamma_4,$$





Champ de vitesses en 2D : $V = \nabla u$

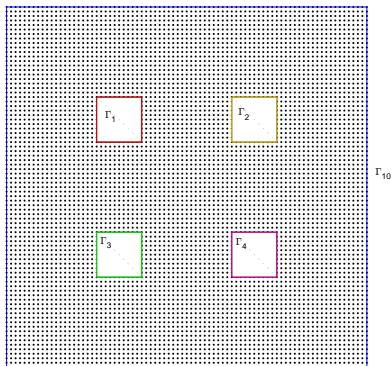
Trouver $u : \Omega \rightarrow \mathbb{R}$ tel que

$$-\Delta u = 0 \text{ dans } \Omega \subset \mathbb{R}^2,$$

$$u = -1 \text{ sur } \Gamma_2 \cup \Gamma_3,$$

$$u = 1 \text{ sur } \Gamma_1 \cup \Gamma_4,$$

$$\frac{\partial u}{\partial n} = 0 \text{ sur } \Gamma_{10}.$$





Champ de vitesses en 2D : $V = \nabla u$

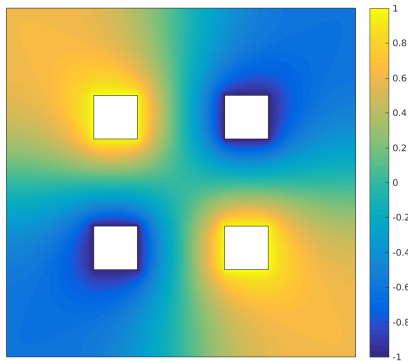
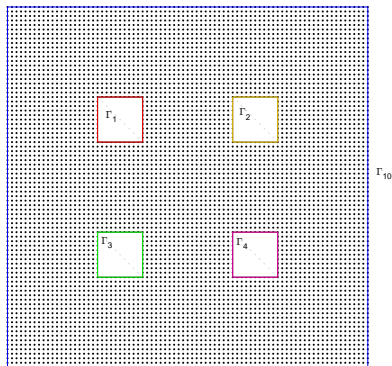
Trouver $u : \Omega \rightarrow \mathbb{R}$ tel que

$$-\Delta u = 0 \text{ dans } \Omega \subset \mathbb{R}^2,$$

$$u = -1 \text{ sur } \Gamma_2 \cup \Gamma_3,$$

$$u = 1 \text{ sur } \Gamma_1 \cup \Gamma_4,$$

$$\frac{\partial u}{\partial n} = 0 \text{ sur } \Gamma_{10}.$$





Champ de vitesses en 2D : $V = \nabla u$

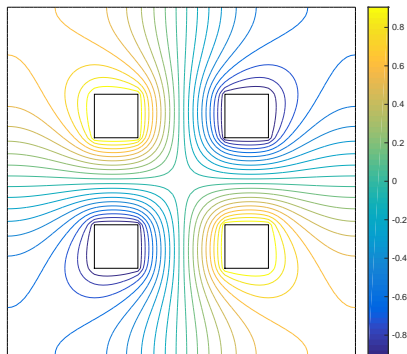
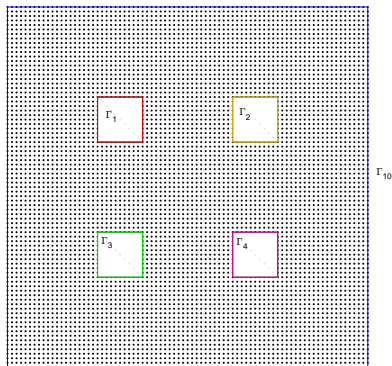
Trouver $u : \Omega \rightarrow \mathbb{R}$ tel que

$$-\Delta u = 0 \text{ dans } \Omega \subset \mathbb{R}^2,$$

$$u = -1 \text{ sur } \Gamma_2 \cup \Gamma_3,$$

$$u = 1 \text{ sur } \Gamma_1 \cup \Gamma_4,$$

$$\frac{\partial u}{\partial n} = 0 \text{ sur } \Gamma_{10}.$$





Problème en dimension n

Find $u : \Omega \rightarrow \mathbb{R}$ such that

$$\begin{aligned} -\Delta u &= f \text{ in } \Omega \subset \mathbb{R}^n, \\ \frac{\partial u}{\partial n} &= g \text{ sur } \partial\Omega. \end{aligned}$$

où Ω est un domaine borné de \mathbb{R}^n .

Ce problème est **mal posé** : non unicité de la solution.

$$u \text{ solution} \Rightarrow u + \text{constante solution}$$

Equation de la chaleur

$$\frac{\partial u}{\partial t}(t, \mathbf{x}) - D\Delta u(t, \mathbf{x}) = \frac{f(t, \mathbf{x})}{\rho c}, \quad \forall \mathbf{x} \in \Omega, \quad \forall t \in [0, T] \quad (4.5)$$

- $\Omega \subset \mathbb{R}^d$ de frontière $\partial\Omega$
- D , coefficient de diffusivité thermique (en m^2/s),
- f , production volumique de chaleur (en W/m^3),
- ρ , masse volumique du matériau (en kg/m^3),
- c , chaleur spécifique massique du matériau (en $J/kg/K$),
- Δu laplacien (en espace) : $\Delta u = \frac{\partial^2 u}{\partial x_1^2} + \dots + \frac{\partial^2 u}{\partial x_d^2}$

Problème bien posé ?

Equation de la chaleur

$$\frac{\partial u}{\partial t}(t, \mathbf{x}) - D\Delta u(t, \mathbf{x}) = \frac{f(t, \mathbf{x})}{\rho c}, \quad \forall \mathbf{x} \in \Omega, \quad \forall t \in [0, T] \quad (4.5)$$

Problème bien posé :

- **condition initiale**

$$\forall \mathbf{x} \in \Omega, \quad u(0, \mathbf{x}) = u_0(\mathbf{x}) \quad (4.6)$$

- **conditions aux limites sur $\partial\Omega$**

- ▶ **Dirichlet :**

$$\forall \mathbf{x} \in \Gamma_D \subset \partial\Omega, \quad \forall t \in [0, T] \quad u(t, \mathbf{x}) = g_D(t, \mathbf{x})$$

- ▶ **Neumann :**

$$\forall \mathbf{x} \in \Gamma_N \subset \partial\Omega, \quad \forall t \in [0, T] \quad D \frac{\partial u}{\partial n}(t, \mathbf{x}) = g_N(t, \mathbf{x})$$

- ▶ **Robin :**

$$\forall \mathbf{x} \in \Gamma_R \subset \partial\Omega, \quad \forall t \in [0, T] \quad D \frac{\partial u}{\partial n}(t, \mathbf{x}) + \alpha u(t, \mathbf{x}) = g_N(t, \mathbf{x})$$

💡 Problème de chaleur en 2D

Find $u : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ such that

$$\frac{\partial u}{\partial t} - D\Delta u = 0 \text{ in } [0, T] \times \Omega,$$

$$u(0, \mathbf{x}) = 20 \quad \forall \mathbf{x} \in \Omega,$$

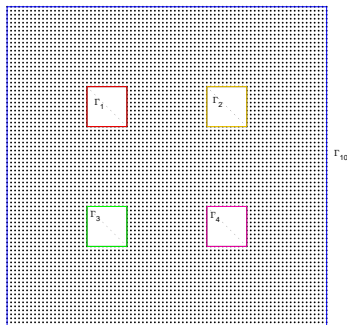
$$\frac{\partial u}{\partial n} = 0 \text{ on } \Gamma_{10},$$

$$u = g_1 \text{ on } \Gamma_2 \cup \Gamma_3,$$

$$u = g_2 \text{ on } \Gamma_1 \cup \Gamma_4,$$

où Ω (cotés de 20cm)

- $D = 98.8 \times 10^{-6}$ (aluminium)
ou $D = 23.9 \times 10^{-6}$ (plomb),
- $\forall \mathbf{x} \in \Gamma_2 \cup \Gamma_3$,
 $g_1(t, \mathbf{x}) = (20 + 40t)$ si $t \leq 1$
et $g_1(t, \mathbf{x}) = 60$ sinon,
- $\forall \mathbf{x} \in \Gamma_1 \cup \Gamma_4$,
 $g_2(t, \mathbf{x}) = (20 + 80t)$ si $t \leq 1$
et $g_2(t, \mathbf{x}) = 100$ sinon.



$$\frac{\partial^2 u}{\partial t^2}(t, \mathbf{x}) - c^2 \Delta u(t, \mathbf{x}) = 0, \quad \forall \mathbf{x} \in \Omega, \quad \forall t \in [0, T] \quad (4.7)$$

- $\Omega \subset \mathbb{R}^d$ de frontière $\partial\Omega$
- $c > 0$ vitesse de propagation de l'onde,

Problème bien posé ?

$$\frac{\partial^2 u}{\partial t^2}(t, \mathbf{x}) - c^2 \Delta u(t, \mathbf{x}) = 0, \quad \forall \mathbf{x} \in \Omega, \quad \forall t \in [0, T]$$

- **conditions initiales**

$$u(0, \mathbf{x}) = u_0(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega \quad \text{[position initiale]} \quad (4.8)$$

$$\frac{\partial u}{\partial t}(0, \mathbf{x}) = v_0(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega \quad \text{[vitesse initiale]} \quad (4.9)$$

- **conditions aux limites sur $\partial\Omega$**

- ▶ **Dirichlet :**

$$\forall \mathbf{x} \in \Gamma_D \subset \partial\Omega, \quad \forall t \in [0, T], \quad u(t, \mathbf{x}) = g_D(t, \mathbf{x})$$

- ▶ **Neumann :**

$$\forall \mathbf{x} \in \Gamma_N \subset \partial\Omega, \quad \forall t \in [0, T], \quad c^2 \frac{\partial u}{\partial n}(t, \mathbf{x}) = g_N(t, \mathbf{x})$$

- ▶ **Robin :**

$$\forall \mathbf{x} \in \Gamma_R \subset \partial\Omega, \quad \forall t \in [0, T], \quad c^2 \frac{\partial u}{\partial n}(t, \mathbf{x}) + \alpha u(t, \mathbf{x}) = g_N(t, \mathbf{x})$$

- 1 Exemples d'E.D.P.
 - Equation de Laplace/Poisson
 - Equation de la chaleur
 - Equation des ondes
- 2 Méthodes de résolution numérique d'EDP
- 3 Méthode des différences finies 1D
 - EDP stationnaire 1D + Dirichlet
 - EDP stationnaire + CL mixtes
- 4 Problème modèle évolutif
 - Schéma explicite
 - Schéma implicite

Méthodes déterministes :

- **méthode des différences finies**
- méthode des éléments finis
- méthode des volumes finis

- 1 Exemples d'E.D.P.
 - Equation de Laplace/Poisson
 - Equation de la chaleur
 - Equation des ondes
- 2 Méthodes de résolution numérique d'EDP
- 3 Méthode des différences finies 1D
 - EDP stationnaire 1D + Dirichlet
 - EDP stationnaire + CL mixtes
- 4 Problème modèle évolutif
 - Schéma explicite
 - Schéma implicite

- 1 Exemples d'E.D.P.
 - Equation de Laplace/Poisson
 - Equation de la chaleur
 - Equation des ondes
- 2 Méthodes de résolution numérique d'EDP
- 3 Méthode des différences finies 1D
 - EDP stationnaire 1D + Dirichlet
 - EDP stationnaire + CL mixtes
- 4 Problème modèle évolutif
 - Schéma explicite
 - Schéma implicite

Soient $a < b$, $c > 0$, $\alpha \in \mathbb{R}$, $\beta \in \mathbb{R}$, et $f : [a, b] \rightarrow \mathbb{R}$ donnés.

EDP modèle stationnaire 1D

Trouver $u : [a, b] \rightarrow \mathbb{R}$ telle que

$$-u'' + cu = f \text{ in }]a, b[,$$

$$u(a) = \alpha,$$

$$u(b) = \beta.$$

ou

EDP modèle stationnaire 1D : formulation aux points

Trouver $u(x) \in \mathbb{R}$, $\forall x \in [a, b]$ telle que

$$-u''(x) + cu(x) = f(x) \quad \forall x \in]a, b[,$$

$$u(a) = \alpha,$$

$$u(b) = \beta.$$

Chercher u ou $u(x)$, $\forall x \in [a, b]$ (infinité de points!)

EDP modèle stationnaire 1D : formulation aux points

Trouver $u(x) \in \mathbb{R}$, $\forall x \in [a, b]$ telle que

$$-u''(x) + cu(x) = f(x) \quad \forall x \in]a, b[,$$

$$u(a) = \alpha,$$

$$u(b) = \beta.$$

$$x_i = a + ih, \quad \forall i \in \llbracket 0, N \rrbracket, \text{ avec } h = \frac{b-a}{N}.$$

EDP modèle stationnaire 1D : formulation aux points de discrétisation

Trouver $u(x_i) \in \mathbb{R}$, $\forall i \in \llbracket 0, N \rrbracket$ tels que

$$-u''(x_i) + cu(x_i) = f(x_i) \quad \forall i \in \llbracket 0, N \rrbracket, \quad (4.10)$$

$$u(x_0) = \alpha, \quad (4.11)$$

$$u(x_N) = \beta. \quad (4.12)$$

 EDP modèle stationnaire 1D : formulation aux points de discrétisation

Trouver $u(x_i) \in \mathbb{R}$, $\forall i \in \llbracket 0, N \rrbracket$ tels que

$$-u''(x_i) + cu(x_i) = f(x_i) \quad \forall i \in \llbracket 0, N \rrbracket,$$

$$u(x_0) = \alpha,$$

$$u(x_N) = \beta.$$

$$u''(x_i) = (D_h^2 u)(x_i) + \mathcal{O}(h^2) = \frac{u(x_{i+1}) - 2u(x_i) + u(x_{i-1}))}{h^2} + \mathcal{O}(h^2).$$

💡 EDP modèle stationnaire 1D : formulation aux points de discrétisation

Trouver $u(x_i) \in \mathbb{R}$, $\forall i \in \llbracket 0, N \rrbracket$ tels que

$$-u''(x_i) + cu(x_i) = f(x_i) \quad \forall i \in \llbracket 0, N \rrbracket,$$

$$u(x_0) = \alpha,$$

$$u(x_N) = \beta.$$

$$u''(x_i) = (D_h^2 u)(x_i) + \mathcal{O}(h^2) = \frac{u(x_{i+1}) - 2u(x_i) + u(x_{i-1}))}{h^2} + \mathcal{O}(h^2).$$


💡 EDP modèle stationnaire en dimension 1 : formulation aux points de discrétisation (bis)

Trouver $u(x_i) \in \mathbb{R}$, $\forall i \in \llbracket 0, N \rrbracket$ tels que

$$-\frac{u(x_{i+1}) - 2u(x_i) + u(x_{i-1}))}{h^2} - \mathcal{O}(h^2) + cu(x_i) = f(x_i) \quad \forall i \in \llbracket 0, N \rrbracket, \quad (4.13)$$

$$u(x_0) = \alpha, \quad (4.14)$$


$$u(x_N) = \beta. \quad (4.15)$$

 EDP modèle stationnaire en dimension 1 : formulation aux points de discrétisation (bis)

Trouver $u(x_i) \in \mathbb{R}$, $\forall i \in \llbracket 0, N \rrbracket$ tels que

$$\begin{aligned} -\frac{u(x_{i+1}) - 2u(x_i) + u(x_{i-1}))}{h^2} - \mathcal{O}(h^2) + cu(x_i) &= f(x_i) \quad \forall i \in \llbracket 0, N \rrbracket, \\ u(x_0) &= \alpha, \\ u(x_N) &= \beta. \end{aligned}$$

On oublie le $\mathcal{O}(h^2)$ et on pose $u_i \approx u(x_i)$.

 EDP modèle stationnaire en dimension 1 : formulation aux points de discrétisation (bis)

Trouver $u(x_i) \in \mathbb{R}, \forall i \in \llbracket 0, N \rrbracket$ tels que

$$\begin{aligned} -\frac{u(x_{i+1}) - 2u(x_i) + u(x_{i-1}))}{h^2} - \mathcal{O}(h^2) + cu(x_i) &= f(x_i) \quad \forall i \in \llbracket 0, N \rrbracket, \\ u(x_0) &= \alpha, \\ u(x_N) &= \beta. \end{aligned}$$

On oublie le $\mathcal{O}(h^2)$ et on pose $u_i \approx u(x_i)$.

 EDP modèle stationnaire 1D : schéma aux différences finies

Trouver $u_i \in \mathbb{R}, \forall i \in \llbracket 0, N \rrbracket$ tels que

$$-\frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + cu_i = f(x_i) \quad \forall i \in \llbracket 0, N \rrbracket, \quad (4.16)$$

$$u_0 = \alpha, \quad (4.17)$$

$$u_N = \beta. \quad (4.18)$$



EDP modèle stationnaire 1D : schéma aux différences finies

Trouver $u_i \in \mathbb{R}$, $\forall i \in \llbracket 0, N \rrbracket$ tels que

$$-\frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + cu_i = f(x_i) \quad \forall i \in \llbracket 0, N \rrbracket, \quad (4.16)$$

$$u_0 = \alpha, \quad (4.17)$$

$$u_N = \beta. \quad (4.18)$$

système linéaire de $N + 1$ équations à $N + 1$ inconnues !

$$\left\{ \begin{array}{lll} u_0 & = & \alpha \quad \leftarrow \text{eq. en } x_0 \\ -u_2 + \mu u_1 - u_0 & = & h^2 f(x_1) \quad \leftarrow \text{eq. en } x_1 \\ & \vdots & \\ -u_N + \mu u_{N-1} - u_{N-2} & = & h^2 f(x_{N-1}) \quad \leftarrow \text{eq. en } x_{N-1} \\ u_N & = & \beta \quad \leftarrow \text{eq. en } x_N \end{array} \right.$$

avec $\mu = 2 + ch^2$.

$$\left\{ \begin{array}{lll} u_0 & = & \alpha \quad \leftarrow \text{eq. en } x_0 \\ -u_2 + \mu u_1 - u_0 & = & h^2 f(x_1) \quad \leftarrow \text{eq. en } x_1 \\ -u_3 + \mu u_2 - u_1 & = & h^2 f(x_2) \quad \leftarrow \text{eq. en } x_2 \\ & \vdots & \\ -u_{N-1} + \mu u_{N-2} - u_{N-3} & = & h^2 f(x_{N-2}) \quad \leftarrow \text{eq. en } x_{N-2} \\ -u_N + \mu u_{N-1} - u_{N-2} & = & h^2 f(x_{N-1}) \quad \leftarrow \text{eq. en } x_{N-1} \\ u_N & = & \beta \quad \leftarrow \text{eq. en } x_N \end{array} \right.$$

$$\mathbb{A} \mathbf{U} \stackrel{\text{def}}{=} \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & \dots & 0 & 0 \\ -1 & \mu & -1 & 0 & \dots & \dots & 0 & 0 \\ 0 & -1 & \mu & -1 & 0 & \dots & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & 0 & \vdots \\ \vdots & 0 & \dots & 0 & 1 & \mu & -1 & 0 \\ 0 & 0 & \dots & \dots & 0 & -1 & \mu & -1 \\ 0 & \dots & \dots & \dots & \dots & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ \vdots \\ u_{N-2} \\ u_{N-1} \\ u_N \end{pmatrix} = \begin{pmatrix} \alpha \\ h^2 f(x_1) \\ h^2 f(x_2) \\ \vdots \\ \vdots \\ h^2 f(x_{N-2}) \\ h^2 f(x_{N-1}) \\ \beta \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{B} \quad (4.19)$$

avec $\mu = 2 + ch^2$.



Proposition 4.1: admis

Le schéma aux différences finies (4.16)-(4.18) est consistant à l'ordre 2 avec l'EDP (4.10)-(4.10) et on a

$$\max_{i \in \llbracket 0, N \rrbracket} |u(x_i) - u_i| = \mathcal{O}(h^2). \quad (4.20)$$



Exercice 7.1: (schéma étudié en cours)

Q.1 Ecrire la fonction `ASSEMBLEMAT1D` retournant la matrice $\mathbb{M} \in \mathcal{M}_d(\mathbb{R})$ définie par

$$\mathbb{M} = \begin{pmatrix} \gamma & 0 & 0 & \dots & \dots & \dots & 0 \\ \beta & \alpha & \beta & \ddots & & & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & & \ddots & \beta & \alpha & \beta \\ 0 & \dots & \dots & \dots & 0 & 0 & \gamma \end{pmatrix} \quad (4.21)$$

où α , β et γ sont des réels donnés.

On souhaite résoudre par un schéma aux différences finies l'EDP suivante

$$\begin{aligned} -u'' + cu &= f \text{ in }]a, b[, \\ u(a) &= \alpha, \\ u(b) &= \beta. \end{aligned}$$

Q.2 En prenant le jeu de données $a = 0$, $b = 2\pi$, $c = 1$, $\alpha = 1$, $\beta = -1$ et $f : x \mapsto \cos(x^2)$, écrire une fonction permettant de résoudre l'EDP précédente. On pourra utiliser la fonction $\mathbf{X} \leftarrow \text{SOLVE}(\mathbb{A}, \mathbf{B})$ retournant la solution du système linéaire $\mathbb{A}\mathbf{X} = \mathbf{B}$.

Q.3 En choisissant judicieusement un jeu de données écrire un programme permettant de vérifier l'ordre du schéma utilisé à l'aide de la formule (4.20).

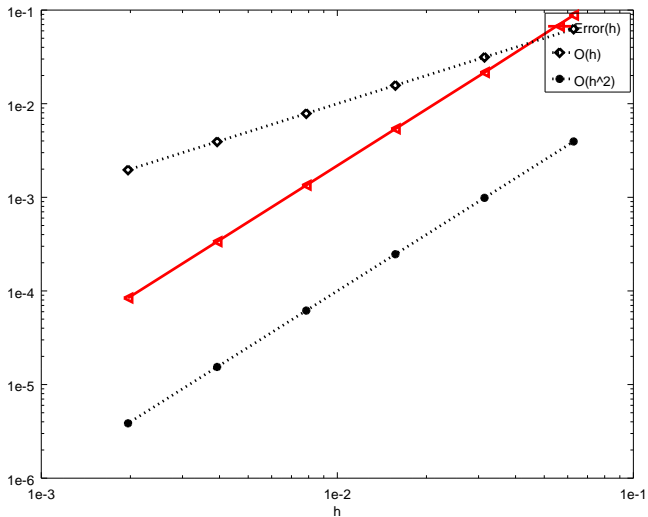


Figure: Représentation en échelle logarithmique

```

1 function M=AssembleMat1D(d,alp,bet,gam)
2 M=sparse(d,d);
3 M(1,1)=gam;M(d,d)=gam;
4 for i=2:d-1
5     M(i,i)=alp;
6     M(i,i-1)=bet;M(i,i+1)=bet;
7 end
8 end

```

Listing: fonction Matlab/Octave AssembleMat1D

```

1 function [x,U]=solveEDP1(a,b,c,alp,bet,f,N)
2 h=(b-a)/N;
3 x=a:h:b;
4 A=AssembleMat1D(N+1,2+c*h*h,-1,h*h);
5 B=zeros(N+1,1);
6 B(1)=alp;B(N+1)=bet;
7 for i=2:N
8     B(i)=f(x(i));
9 end
10 B=h*h*B;
11 U=A\B;
12 end

```

Listing: fonction Matlab/Octave solveEDP1

```

1 clear all
2 close all
3 % Initialisation des donnees
4 uex=@(x) sin(x.^2);
5 c=1;
6 f=@(x) 4*x^2*sin(x^2) - 2*cos(x^2) + c*sin(x^2);
7 a=0;b=2*pi;
8 % Calcul des erreurs
9 LN=[100,200,400,800,1600,3200];
10 k=1;
11 for N=LN
12     [x,U]=solveEDP1(a,b,c,uex(a),uex(b),f,N);
13     H(k)=(b-a)/N;
14     E(k)=max(abs(uex(x)'-U));
15     k=k+1;
16 end
17 % Representation graphique
18 loglog(H,E,'r<-','LineWidth',2)
19 hold on
20 loglog(H,H,'kd:','LineWidth',2)
21 loglog(H,H.^2,'k*:', 'LineWidth',2)
22 legend('Error(h)','0(h)','0(h^2)')
23 xlabel('h')

```

Listing: Script Matlab/Octave pour la représentation de l'ordre

- 1 Exemples d'E.D.P.
 - Equation de Laplace/Poisson
 - Equation de la chaleur
 - Equation des ondes
- 2 Méthodes de résolution numérique d'EDP
- 3 Méthode des différences finies 1D
 - EDP stationnaire 1D + Dirichlet
 - EDP stationnaire + CL mixtes
- 4 Problème modèle évolutif
 - Schéma explicite
 - Schéma implicite

EDP modèle stationnaire 1D avec condition de Dirichlet à droite et Neumann à gauche


Trouver $u : [a, b] \rightarrow \mathbb{R}$ telle que

$$-u'' + cu = f \text{ in }]a, b[, \quad (4.22)$$

$$u(a) = \alpha, \quad (4.23)$$

$$u'(b) = \beta. \quad (4.24)$$

Seule la dernière ligne du système linéaire est à modifier! Remplacer par ???

 EDP modèle stationnaire 1D avec condition de Dirichlet à droite et Neumann à gauche

Trouver $u : [a, b] \rightarrow \mathbb{R}$ telle que


$$-u'' + cu = f \text{ in }]a, b[, \quad (4.22)$$

$$u(a) = \alpha, \quad (4.23)$$

$$u'(b) = \beta. \quad (4.24)$$

Seule la dernière ligne du système linéaire est à modifier! Remplacer par ???

$$u'(x_N) = (D_h^+ u)(x_N) + \mathcal{O}(h) = \frac{u(x_N) - u(x_{N-1})}{h} + \mathcal{O}(h) = \beta.$$


EDP modèle stationnaire 1D avec condition de Dirichlet à droite et Neumann à gauche

Trouver $u : [a, b] \rightarrow \mathbb{R}$ telle que

$$-u'' + cu = f \text{ in }]a, b[, \quad (4.22)$$

$$u(a) = \alpha, \quad (4.23)$$

$$u'(b) = \beta. \quad (4.24)$$

Seule la dernière ligne du système linéaire est à modifier! Remplacer par ???

$$u'(x_N) = (D_h^+ u)(x_N) + \mathcal{O}(h) = \frac{u(x_N) - u(x_{N-1})}{h} + \mathcal{O}(h) = \beta.$$

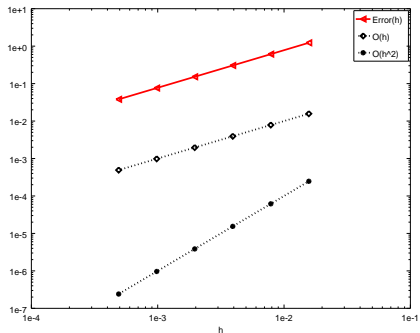
$$\frac{u_N - u_{N-1}}{h} = \beta. \quad (4.25)$$

$$\left\{ \begin{array}{lll} u_0 & = & \alpha \quad \leftarrow \text{eq. en } x_0 \\ -u_2 + \mu u_1 - u_0 & = & h^2 f(x_1) \quad \leftarrow \text{eq. en } x_1 \\ -u_3 + \mu u_2 - u_1 & = & h^2 f(x_2) \quad \leftarrow \text{eq. en } x_2 \\ & \vdots & \\ -u_{N-1} + \mu u_{N-2} - u_{N-3} & = & h^2 f(x_{N-2}) \quad \leftarrow \text{eq. en } x_{N-2} \\ -u_N + \mu u_{N-1} - u_{N-2} & = & h^2 f(x_{N-1}) \quad \leftarrow \text{eq. en } x_{N-1} \\ u_N - u_{N-1} & = & h\beta \quad \leftarrow \text{eq. en } x_N \end{array} \right.$$

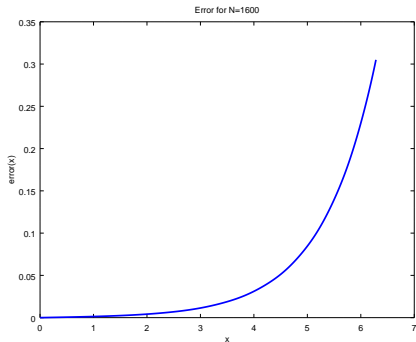
$$\mathbb{A} \mathbf{U} \stackrel{\text{def}}{=} \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & \dots & 0 & 0 \\ -1 & \mu & -1 & 0 & \dots & \dots & 0 & 0 \\ 0 & -1 & \mu & -1 & 0 & \dots & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & 0 & \vdots \\ \vdots & 0 & \dots & 0 & 1 & \mu & -1 & 0 \\ 0 & 0 & \dots & \dots & 0 & -1 & \mu & -1 \\ 0 & \dots & \dots & \dots & \dots & \dots & -1 & 1 \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ \vdots \\ u_{N-2} \\ u_{N-1} \\ u_N \end{pmatrix} = \begin{pmatrix} \alpha \\ h^2 f(x_1) \\ h^2 f(x_2) \\ \vdots \\ \vdots \\ h^2 f(x_{N-2}) \\ h^2 f(x_{N-1}) \\ h\beta \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{B} \quad (4.26)$$

Mais ...

Schéma d'ordre 1 !!!



(a) Représentation en échelle logarithmique de l'ordre du schéma



(b) Représentation de l'erreur en fonction de x pour $N = 1600$

Ecrire un schéma d'ordre 2 pour Neumann



Exercice 7.2

Soit φ une fonction suffisamment régulière et $h > 0$

Q.1 Montrer que

$$\frac{d\varphi}{dx}(x) = \frac{-3\varphi(x) + 4\varphi(x+h) - \varphi(x+2h)}{2h} + \mathcal{O}(h^2) \quad (4.27)$$

Q.2 Montrer que

$$\frac{d\varphi}{dx}(x) = \frac{3\varphi(x) - 4\varphi(x-h) + \varphi(x-2h)}{2h} + \mathcal{O}(h^2) \quad (4.28)$$

Q.3 Déterminer une formule permettant de calculer une approximation à l'ordre 2 de $\frac{d^2\varphi}{dx^2}(x)$ en utilisant uniquement des valeurs de la fonction φ aux points $x+ih$ avec $i \in \mathbb{N}$.

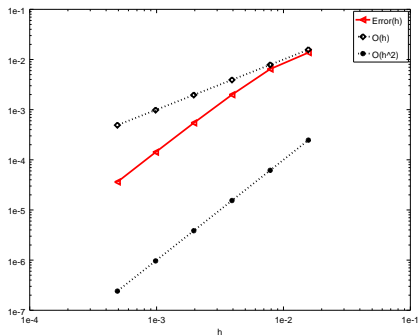
Q.4 Déterminer une formule permettant de calculer une approximation à l'ordre 2 de $\frac{d^2\varphi}{dx^2}(x)$ en utilisant uniquement des valeurs de la fonction φ aux points $x-ih$ avec $i \in \mathbb{N}$.

$$\left\{ \begin{array}{lll} u_0 & = & \alpha \quad \leftarrow \text{eq. en } x_0 \\ -u_2 + \mu u_1 - u_0 & = & f(x_1) \quad \leftarrow \text{eq. en } x_1 \\ -u_3 + \mu u_2 - u_1 & = & f(x_2) \quad \leftarrow \text{eq. en } x_2 \\ & \vdots & \\ -u_{N-1} + \mu u_{N-2} - u_{N-3} & = & f(x_{N-2}) \quad \leftarrow \text{eq. en } x_{N-2} \\ -u_N + \mu u_{N-1} - u_{N-2} & = & f(x_{N-1}) \quad \leftarrow \text{eq. en } x_{N-1} \\ 3u_N - 4u_{N-1} + u_{N-2} & = & 2h\beta \quad \leftarrow \text{eq. en } x_N \end{array} \right.$$

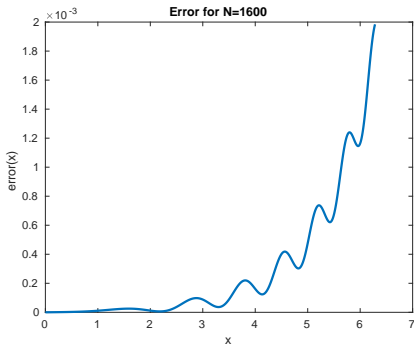
$$\mathbb{A} \mathbf{U} \stackrel{\text{def}}{=} \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & \dots & 0 & 0 \\ -1 & \mu & -1 & 0 & \dots & \dots & 0 & 0 \\ 0 & -1 & \mu & -1 & 0 & \dots & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & 0 & \vdots \\ \vdots & 0 & \dots & 0 & 1 & \mu & -1 & 0 \\ 0 & 0 & \dots & \dots & 0 & -1 & \mu & -1 \\ 0 & \dots & \dots & \dots & \dots & 1 & -4 & 3 \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ \vdots \\ u_{N-2} \\ u_{N-1} \\ u_N \end{pmatrix} = \begin{pmatrix} \alpha \\ h^2 f(x_1) \\ h^2 f(x_2) \\ \vdots \\ \vdots \\ h^2 f(x_{N-2}) \\ h^2 f(x_{N-1}) \\ 2h\beta \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{B} \quad (4.54)$$

et ...

Schéma d'ordre 2



(a) Représentation en échelle logarithmique de l'ordre du schéma



(b) Représentation de l'erreur en fonction de x pour $N = 1600$

Exercice 7.3

Soit le problème suivant

$$-u''(x) + c(x)u(x) = f(x), \quad \forall x \in]a; b[, \quad (4.55)$$

$$u'(a) = \alpha, \quad (4.56)$$

$$u(b) = \beta. \quad (4.57)$$

où c est une fonction positive.

Q.1

- Quelles sont les données du problème (4.55)-(4.57)? (préciser le type de chaque donnée : réel, entier, fonction, vecteur, ...)
- Quelles sont les inconnues du problème (4.55)-(4.57)? (préciser le type)
- Quelles sont les conditions initiales?
- Quelles sont les conditions aux limites?

Q.2 Construire une discrétisation régulière de $[a; b]$ avec N pas de discrétisation en espace. On note x_i , $i \in [0, N]$ cette discrétisation. On souhaite résoudre (4.55) à l'aide du schéma numérique

$$-\frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2} + c_i u_i = f_i. \quad (4.58)$$

Q.3

- Expliquer comment le schéma (4.58) a été obtenu à partir de (4.55) et préciser ce que représente les termes u_i , f_i , c_i et Δx ?
- Donner l'ensemble \mathcal{E} des valeurs que peut prendre i dans le schéma (4.55).
- Construire une discrétisation des conditions aux limites d'ordre 2 au moins.
- Le schéma global est de quel ordre? Justifiez.

On note \mathbf{V} le vecteur de dimension $N + 1$, de composantes $\mathbf{V}_i = u_{i-1}$, $\forall i \in [1, N + 1]$.

Q.4 Montrer que le vecteur \mathbf{V} est solution du système linéaire

$$\mathbb{A}\mathbf{V} = \mathbf{F} \quad (4.59)$$

en explicitant la matrice \mathbb{A} et le vecteur \mathbf{F} (préciser les dimensions).

Q.5 Ecrire un algorithme complet de résolution du problème (4.55) à (4.57) basé sur (4.59). (Utiliser au maximum les fonctions). On pourra utiliser la fonction $\mathbf{X} \leftarrow \text{SOLVE}(\mathbb{A}, \mathbf{B})$ retournant la solution du système linéaire $\mathbb{A}\mathbf{X} = \mathbf{B}$.

- 1 Exemples d'E.D.P.
 - Equation de Laplace/Poisson
 - Equation de la chaleur
 - Equation des ondes
- 2 Méthodes de résolution numérique d'EDP
- 3 Méthode des différences finies 1D
 - EDP stationnaire 1D + Dirichlet
 - EDP stationnaire + CL mixtes
- 4 **Problème modèle évolutif**
 - Schéma explicite
 - Schéma implicite

- 1 Exemples d'E.D.P.
 - Equation de Laplace/Poisson
 - Equation de la chaleur
 - Equation des ondes
- 2 Méthodes de résolution numérique d'EDP
- 3 Méthode des différences finies 1D
 - EDP stationnaire 1D + Dirichlet
 - EDP stationnaire + CL mixtes
- 4 **Problème modèle évolutif**
 - Schéma explicite
 - Schéma implicite



EDP modèle instationnaire en dimension 1 : équation de la chaleur

Trouver $u : [0, T] \times [a, b] \rightarrow \mathbb{R}$ telle que

$$\frac{\partial u}{\partial t}(t, x) - D \frac{\partial^2 u}{\partial x^2}(t, x) = f(t, x), \quad \forall (t, x) \in]0, T] \times]a, b[, \quad (4.76)$$

$$u(0, x) = u_0(x), \quad \forall x \in [a, b] \quad (4.77)$$

$$-D \frac{\partial u}{\partial x}(t, a) = \alpha(t), \quad \forall t \in [0, T] \quad (4.78)$$

$$u(t, b) = \beta(t), \quad \forall t \in [0, T] \quad (4.79)$$

où $a < b$, $D > 0$ (coefficient de diffusivité), $\alpha : [0, T] \rightarrow \mathbb{R}$,
 $\beta : [0, T] \rightarrow \mathbb{R}$, $u_0 : [a, b] \rightarrow \mathbb{R}$ et $f : [0, T] \times [a, b] \rightarrow \mathbb{R}$ donnés.

condition de compatibilité :

$$u_0(b) = \beta(0). \quad (4.80)$$

$$x_i = a + i\Delta_x, \quad \forall i \in \llbracket 0, N_x \rrbracket, \quad \text{avec } \Delta_x = (b - a)/N_x$$

$$t^n = n\Delta_t, \quad \forall n \in \llbracket 0, N_t \rrbracket, \quad \text{avec } \Delta_t = T/N_t.$$

Objectif: Trouver $\mathbf{u}_i^n \approx u(t^n, x_i)$, $\forall n \in \llbracket 0, N_t \rrbracket$, $\forall i \in \llbracket 0, N_x \rrbracket$

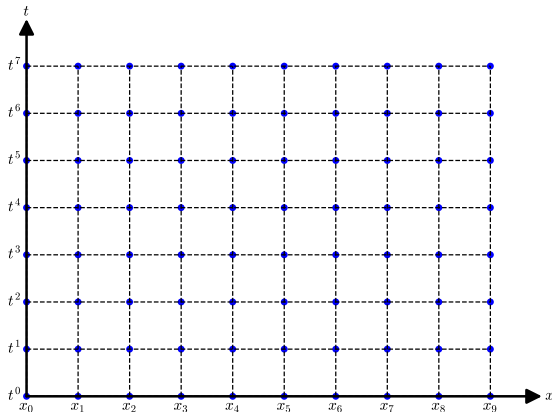


Figure: Grille espace-temps avec $N_t = 7$ et $N_x = 9$.



EDP modèle d'évolution en dimension 1 : équation de la chaleur, formulation aux points de discrétisation

Trouver $u(t^n, x_i) \in \mathbb{R}$, $\forall n \in \llbracket 0, N_t \rrbracket$, $\forall i \in \llbracket 0, N_x \rrbracket$, tels que

$$\frac{\partial u}{\partial t}(t^n, x_i) - D \frac{\partial^2 u}{\partial x^2}(t^n, x_i) = f(t^n, x_i), \quad (4.81)$$

$$u(t^0, x_i) = u_0(x_i), \quad \forall i \in \llbracket 0, N_x \rrbracket, \quad (4.82)$$

$$-D \frac{\partial u}{\partial x}(t^n, x_0) = \alpha(t^n), \quad \forall n \in \llbracket 0, N_t \rrbracket \quad (4.83)$$

$$u(t^n, x_{N_x}) = \beta(t^n), \quad \forall n \in \llbracket 0, N_t \rrbracket \quad (4.84)$$

⇒ il nous faut maintenant discrétiser les opérateurs de dérivation

$$\frac{\partial u}{\partial t}, \quad \frac{\partial^2 u}{\partial x^2} \quad \text{et} \quad \frac{\partial u}{\partial x}$$



Proposition 4.1: (admis)

Soit U un ouvert non vide de \mathbb{R}^n et f une application $f : U \subset \mathbb{R}^n \rightarrow \mathbb{R}$ avec $f \in \mathcal{C}^r(U)$. Soient $\mathbf{x} \in U$, $i \in \llbracket 1, n \rrbracket$, et $h \in \mathbb{R}^*$ vérifiant $\forall t \in [0, 1]$, $\mathbf{x} + t h \mathbf{e}^{[i]} \in U$ où $\mathbf{e}^{[i]}$ est le i -ème vecteur de la base canonique de \mathbb{R}^n .

Alors il existe $\theta \in]0, 1[$ tel quel

$$f(\mathbf{x} + h \mathbf{e}^{[i]}) = f(\mathbf{x}) + \sum_{k=1}^{r-1} \frac{h^k}{k!} \frac{\partial^k f}{\partial x_i^k}(\mathbf{x}) + \frac{h^r}{r!} \frac{\partial^r f}{\partial x_i^r}(\mathbf{x} + \theta h \mathbf{e}^{[i]}) \quad (4.85)$$

où $\mathbf{e}^{[i]}$ est le i -ème vecteur de la base canonique de \mathbb{R}^n .

$$f(\mathbf{x} + h \mathbf{e}^{[i]}) = f(\mathbf{x}) + \sum_{k=1}^{r-1} \frac{h^k}{k!} \frac{\partial^k f}{\partial x_i^k}(\mathbf{x}) + \mathcal{O}(h^r) \quad (4.86)$$

On déduit des développements de Taylor :

$$\frac{\partial^2 u}{\partial x^2}(t, x) = \frac{u(t, x + h) - 2u(t, x) + u(t, x - h)}{h^2} + \mathcal{O}(h^2)$$

Avec $h = \Delta_x$ on obtient

$$\frac{\partial^2 u}{\partial x^2}(t^n, x_i) = \frac{u(t^n, x_{i+1}) - 2u(t^n, x_i) + u(t^n, x_{i-1}))}{\Delta_x^2} + \mathcal{O}(\Delta_x^2) \quad (4.87)$$

On déduit des développements de Taylor :

$$\begin{aligned}\frac{\partial u}{\partial t}(t, x) &= \frac{u(t+h, x) - u(t, x)}{h} + \mathcal{O}(h) \\ \frac{\partial u}{\partial t}(t, x) &= \frac{u(t, x) - u(t-h, x)}{h} + \mathcal{O}(h).\end{aligned}$$

Avec $h = \Delta_t$ on obtient

$$\frac{\partial u}{\partial t}(t^n, x_i) = \frac{u(t^{n+1}, x_i) - u(t^n, x_i)}{\Delta_t} + \mathcal{O}(\Delta_t) \quad (4.88)$$

$$\frac{\partial u}{\partial t}(t^n, x_i) = \frac{u(t^n, x_i) - u(t^{n-1}, x_i)}{\Delta_t} + \mathcal{O}(\Delta_t). \quad (4.89)$$

On déduit des développements de Taylor :

$$\frac{\partial u}{\partial x}(t, x) = \frac{u(t, x+h) - u(t, x)}{h} + \mathcal{O}(h)$$

$$\frac{\partial u}{\partial x}(t, x) = \frac{u(t, x) - u(t, x-h)}{h} + \mathcal{O}(h)$$

$$\frac{\partial u}{\partial x}(t, x) = \frac{u(t, x+h) - u(t, x-h)}{2h} + \mathcal{O}(h^2)$$

$$\frac{\partial u}{\partial x}(t, x) = \frac{-3u(t, x) + 4u(t, x+h) - u(t, x+2h)}{2h} + \mathcal{O}(h^2)$$

$$\frac{\partial u}{\partial x}(t, x) = \frac{3u(t, x) - 4u(t, x-h) + u(t, x-2h)}{2h} + \mathcal{O}(h^2)$$

On veut approcher $\frac{\partial u}{\partial x}(t^n, x_0)$ à l'ordre 2! \Rightarrow 4ème approximation.

Avec $h = \Delta_x$ on obtient

$$\frac{\partial u}{\partial x}(t^n, x_0) = \frac{-3u(t^n, x_0) + 4u(t^n, x_1) - u(t^n, x_2)}{2\Delta_x} + \mathcal{O}(\Delta_x^2) \quad (4.90)$$

- 1 Exemples d'E.D.P.
 - Equation de Laplace/Poisson
 - Equation de la chaleur
 - Equation des ondes
- 2 Méthodes de résolution numérique d'EDP
- 3 Méthode des différences finies 1D
 - EDP stationnaire 1D + Dirichlet
 - EDP stationnaire + CL mixtes
- 4 **Problème modèle évolutif**
 - **Schéma explicite**
 - Schéma implicite

Schéma explicite en temps pour l'EDP (4.81) à (4.84)

On rappelle (4.81)

$$\frac{\partial u}{\partial t}(t^n, x_i) - D \frac{\partial^2 u}{\partial x^2}(t^n, x_i) = f(t^n, x_i), \quad \forall n \in \llbracket 0, N_t \rrbracket, \quad \forall i \in \llbracket 0, N_x \rrbracket,$$

qui devient avec (4.88) et (4.87)

$$\begin{aligned} & \frac{u(t^{n+1}, x_i) - u(t^n, x_i)}{\Delta_t} + \mathcal{O}(\Delta_t) \\ -D & \frac{u(t^n, x_{i+1}) - 2u(t^n, x_i) + u(t^n, x_{i-1}))}{\Delta_x^2} + \mathcal{O}(\Delta_x^2) = f(t^n, x_i) \end{aligned} \quad (4.94)$$

avec $n \in \llbracket 0, N_t \rrbracket$ et $i \in \llbracket 0, N_x \rrbracket$.

En utilisant (4.89) en lieu et place de (4.88) on obtient un schéma implicite...

Schéma explicite en temps pour l'EDP (4.81) à (4.84)

Un schéma numérique d'ordre 1 en temps et d'ordre 2 en espace pour (4.81): $\forall n \in \llbracket 0, N_t \llbracket, \forall i \in \llbracket 0, N_x \llbracket$

$$\frac{\mathbf{u}_i^{n+1} - \mathbf{u}_i^n}{\Delta_t} - D \frac{\mathbf{u}_{i+1}^n - 2\mathbf{u}_i^n + \mathbf{u}_{i-1}^n}{\Delta_x^2} = \mathbf{f}_i^n \quad (4.95)$$

avec $\mathbf{f}_i^n = f(t^n, x_i)$ et (en espérant) $\mathbf{u}_i^n \approx u(t^n, x_i)$.

(4.95) est équivalent à

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n + D \frac{\Delta_t}{\Delta_x^2} (\mathbf{u}_{i+1}^n - 2\mathbf{u}_i^n + \mathbf{u}_{i-1}^n) + \Delta_t \mathbf{f}_i^n \quad (4.96)$$

Et (4.82), (4.83), (4.84)?

Schéma explicite en temps pour l'EDP (4.81) à (4.84)

On rappelle respectivement (4.82) et (4.84):

$$\begin{aligned}u(t^0, x_i) &= u_0(x_i), & \forall i \in \llbracket 0, N_x \rrbracket \\u(t^n, x_{N_x}) &= \beta(t^n), & \forall n \in \llbracket 0, N_t \rrbracket\end{aligned}$$

qui donne immédiatement (sans approximation)

$$\begin{aligned}\mathbf{u}_i^0 &= u_0(x_i), & \forall i \in \llbracket 0, N_x \rrbracket \\ \mathbf{u}_{N_x}^n &= \beta(t^n), & \forall n \in \llbracket 0, N_t \rrbracket\end{aligned} \quad (4.97)$$

Et (4.83)?

Schéma explicite en temps pour l'EDP (4.81) à (4.84)

On rappelle (4.83) $\forall n \in \llbracket 0, N_t \rrbracket$

$$-D \frac{\partial u}{\partial x}(t^n, x_0) = \alpha(t^n)$$

qui donne avec (4.90)

$$-D \frac{-3u(t^n, x_0) + 4u(t^n, x_1) - u(t^n, x_2)}{2\Delta_x} + \mathcal{O}(\Delta_x^2) = \alpha(t^n)$$

Un schéma numérique d'ordre 2 en espace pour (4.83):

$$-D \frac{-3\mathbf{u}_0^n + 4\mathbf{u}_1^n - \mathbf{u}_2^n}{\Delta_x} = \alpha(t^n)$$

ou encore

$$\mathbf{u}_0^n = \frac{1}{3} \left(\frac{2\Delta_x}{D} \alpha(t^n) + 4\mathbf{u}_1^n - \mathbf{u}_2^n \right). \quad (4.105)$$

Schéma explicite en temps pour l'EDP (4.81) à (4.84)

En résumé, avec $E = D \frac{\Delta_t}{\Delta_x^2}$ et $C = 1 - 2E$

$$\mathbf{u}_i^{n+1} = C\mathbf{u}_i^n + E(\mathbf{u}_{i+1}^n + \mathbf{u}_{i-1}^n) + \Delta_t \mathbf{f}_i^n, \quad \begin{cases} \forall n \in \llbracket 0, N_t \llbracket, \\ \forall i \in \llbracket 0, N_x \llbracket \end{cases} \quad (4.96)$$

$$\mathbf{u}_i^0 = u_0(x_i), \quad \forall i \in \llbracket 0, N_x \llbracket \quad (4.106)$$

$$\mathbf{u}_0^n = \frac{1}{3} \left(\frac{2\Delta_x}{D} \alpha(t^n) + 4\mathbf{u}_1^n - \mathbf{u}_2^n \right), \quad \forall n \in \llbracket 0, N_t \llbracket \quad (4.105)$$

$$\mathbf{u}_{N_x}^n = \beta(t^n), \quad \forall n \in \llbracket 0, N_t \llbracket \quad (4.97)$$

Peut-on calculer $(\mathbf{u}_i^{n+1})_{i=0}^{N_x}$ connaissant $(\mathbf{u}_i^n)_{i=0}^{N_x}$?

$$\mathbf{u}_i^{n+1} = \quad (4.107)$$

$$\mathbf{u}_{N_x}^{n+1} = \quad (4.108)$$

$$\mathbf{u}_0^{n+1} = \quad (4.109)$$

Schéma explicite en temps pour l'EDP (4.81) à (4.84)

En résumé, avec $E = D \frac{\Delta_t}{\Delta_x^2}$ et $C = 1 - 2E$

$$\mathbf{u}_i^{n+1} = C\mathbf{u}_i^n + E(\mathbf{u}_{i+1}^n + \mathbf{u}_{i-1}^n) + \Delta_t \mathbf{f}_i^n, \quad \begin{cases} \forall n \in \llbracket 0, N_t \llbracket, \\ \forall i \in \llbracket 0, N_x \llbracket \end{cases} \quad (4.96)$$

$$\mathbf{u}_i^0 = u_0(x_i), \quad \forall i \in \llbracket 0, N_x \llbracket \quad (4.106)$$

$$\mathbf{u}_0^n = \frac{1}{3} \left(\frac{2\Delta_x}{D} \alpha(t^n) + 4\mathbf{u}_1^n - \mathbf{u}_2^n \right), \quad \forall n \in \llbracket 0, N_t \llbracket \quad (4.105)$$

$$\mathbf{u}_{N_x}^n = \beta(t^n), \quad \forall n \in \llbracket 0, N_t \llbracket \quad (4.97)$$

Peut-on calculer $(\mathbf{u}_i^{n+1})_{i=0}^{N_x}$ connaissant $(\mathbf{u}_i^n)_{i=0}^{N_x}$?

$$\mathbf{u}_i^{n+1} = C\mathbf{u}_i^n + E(\mathbf{u}_{i+1}^n + \mathbf{u}_{i-1}^n) + \Delta_t \mathbf{f}_i^n, \quad \forall i \in \llbracket 0, N_x \llbracket, \quad (4.107)$$

$$\mathbf{u}_{N_x}^{n+1} = \quad (4.108)$$

$$\mathbf{u}_0^{n+1} = \quad (4.109)$$

Schéma explicite en temps pour l'EDP (4.81) à (4.84)

En résumé, avec $E = D \frac{\Delta_t}{\Delta_x^2}$ et $C = 1 - 2E$

$$\mathbf{u}_i^{n+1} = C\mathbf{u}_i^n + E(\mathbf{u}_{i+1}^n + \mathbf{u}_{i-1}^n) + \Delta_t \mathbf{f}_i^n, \quad \begin{cases} \forall n \in \llbracket 0, N_t \llbracket, \\ \forall i \in \llbracket 0, N_x \llbracket \end{cases} \quad (4.96)$$

$$\mathbf{u}_i^0 = u_0(x_i), \quad \forall i \in \llbracket 0, N_x \llbracket \quad (4.106)$$

$$\mathbf{u}_0^n = \frac{1}{3} \left(\frac{2\Delta_x}{D} \alpha(t^n) + 4\mathbf{u}_1^n - \mathbf{u}_2^n \right), \quad \forall n \in \llbracket 0, N_t \llbracket \quad (4.105)$$

$$\mathbf{u}_{N_x}^n = \beta(t^n), \quad \forall n \in \llbracket 0, N_t \llbracket \quad (4.97)$$

Peut-on calculer $(\mathbf{u}_i^{n+1})_{i=0}^{N_x}$ connaissant $(\mathbf{u}_i^n)_{i=0}^{N_x}$?

$$\mathbf{u}_i^{n+1} = C\mathbf{u}_i^n + E(\mathbf{u}_{i+1}^n + \mathbf{u}_{i-1}^n) + \Delta_t \mathbf{f}_i^n, \quad \forall i \in \llbracket 0, N_x \llbracket, \quad (4.107)$$

$$\mathbf{u}_{N_x}^{n+1} = \beta(t^{n+1}), \quad (4.108)$$

$$\mathbf{u}_0^{n+1} = \quad (4.109)$$

Schéma explicite en temps pour l'EDP (4.81) à (4.84)

En résumé, avec $E = D \frac{\Delta_t}{\Delta_x^2}$ et $C = 1 - 2E$

$$\mathbf{u}_i^{n+1} = C\mathbf{u}_i^n + E(\mathbf{u}_{i+1}^n + \mathbf{u}_{i-1}^n) + \Delta_t \mathbf{f}_i^n, \quad \begin{cases} \forall n \in \llbracket 0, N_t \llbracket, \\ \forall i \in \llbracket 0, N_x \llbracket \end{cases} \quad (4.96)$$

$$\mathbf{u}_i^0 = u_0(x_i), \quad \forall i \in \llbracket 0, N_x \llbracket \quad (4.106)$$

$$\mathbf{u}_0^n = \frac{1}{3} \left(\frac{2\Delta_x}{D} \alpha(t^n) + 4\mathbf{u}_1^n - \mathbf{u}_2^n \right), \quad \forall n \in \llbracket 0, N_t \llbracket \quad (4.105)$$

$$\mathbf{u}_{N_x}^n = \beta(t^n), \quad \forall n \in \llbracket 0, N_t \llbracket \quad (4.97)$$

Peut-on calculer $(\mathbf{u}_i^{n+1})_{i=0}^{N_x}$ connaissant $(\mathbf{u}_i^n)_{i=0}^{N_x}$?

$$\mathbf{u}_i^{n+1} = C\mathbf{u}_i^n + E(\mathbf{u}_{i+1}^n + \mathbf{u}_{i-1}^n) + \Delta_t \mathbf{f}_i^n, \quad \forall i \in \llbracket 0, N_x \llbracket, \quad (4.107)$$

$$\mathbf{u}_{N_x}^{n+1} = \beta(t^{n+1}), \quad (4.108)$$

$$\mathbf{u}_0^{n+1} = \frac{1}{3} \left(\frac{2\Delta_x}{D} \alpha(t^{n+1}) + 4\mathbf{u}_1^{n+1} - \mathbf{u}_2^{n+1} \right). \quad (4.109)$$

Schéma explicite en temps pour l'EDP (4.81) à (4.84)

L'algorithme formel est donc le suivant :

- 1: $\mathbf{u}_i^0 \leftarrow u_0(x_i), \forall i \in \llbracket 0, N_x \rrbracket$
- 2: **Pour** $n \leftarrow 0$ à $N_t - 1$ **faire**
- 3: $\mathbf{u}_i^{n+1} \leftarrow C\mathbf{u}_i^n + E(\mathbf{u}_{i+1}^n + \mathbf{u}_{i-1}^n) + \Delta_t \mathbf{f}_i^n, \forall i \in \llbracket 0, N_x \rrbracket$
- 4: $\mathbf{u}_{N_x}^{n+1} \leftarrow \beta(t^{n+1})$
- 5: $\mathbf{u}_0^{n+1} \leftarrow \frac{1}{3} \left(\frac{2\Delta_x}{D} \alpha(t^{n+1}) + 4\mathbf{u}_1^{n+1} - \mathbf{u}_2^{n+1} \right)$
- 6: **Fin Pour**



Exercice 8.1

Q.1 *Ecrire une fonction algorithmique **HEAT1DEX** permettant de retourner la discrétisation en temps, la discrétisation en espace et l'ensemble des \mathbf{u}_i^n , $i \in \llbracket 0, N_x \rrbracket$, $n \in \llbracket 1, N_t \rrbracket$ calculés par le schéma explicite en temps pour l'EDP (4.81) à (4.84).*

Q.2 *Ecrire un programme utilisant cette fonction et permettant de calculer la solution numérique d'un problème dont on connaît la solution exacte.*

Schéma explicite en temps pour l'EDP (4.81) à (4.84)

- Données :**
- a, b : deux réels $a < b$,
 - T : $T > 0$,
 - D : D réel strictement positif, (coefficient de diffusivité)
 - f : $f : [0, T] \times [a, b] \longrightarrow \mathbb{R}$,
 - u_0 : $u_0 : [a, b] \longrightarrow \mathbb{R}$,
 - α : $\alpha : [0, T] \longrightarrow \mathbb{R}$,
 - β : $\beta : [0, T] \longrightarrow \mathbb{R}$, tel que $\beta(0) = u_0(b)$,
 - N_x : $N_x \in \mathbb{N}^*$, nombre de discrétisation en espace,
 - N_t : $N_t \in \mathbb{N}^*$, nombre de discrétisation en temps.

Résultat : \mathbb{U} : tableau 2D/matrice $(N_x + 1) \times (N_t + 1)$ de réels tel que
 $\mathbb{U}(i + 1, n + 1) = \mathbf{u}_i^n, \forall i \in \llbracket 0, N_x \rrbracket, \forall n \in \llbracket 0, N_t \rrbracket$.

colonnes

		1	2	3				$N_t - 1$	N_t	$N_t + 1$
\mathbb{U}	lignes	1	\mathbf{u}_0^0	\mathbf{u}_0^1	\mathbf{u}_0^2			$\mathbf{u}_0^{N_t - 2}$	$\mathbf{u}_0^{N_t - 1}$	$\mathbf{u}_0^{N_t}$
	2	\mathbf{u}_1^0	\mathbf{u}_1^1	\mathbf{u}_1^2			$\mathbf{u}_1^{N_t - 2}$	$\mathbf{u}_1^{N_t - 1}$	$\mathbf{u}_1^{N_t}$	
	3	\mathbf{u}_2^0	\mathbf{u}_2^1	\mathbf{u}_2^2			$\mathbf{u}_2^{N_t - 2}$	$\mathbf{u}_2^{N_t - 1}$	$\mathbf{u}_2^{N_t}$	
	N_x	$\mathbf{u}_{N_x - 1}^0$	$\mathbf{u}_{N_x - 1}^1$	$\mathbf{u}_{N_x - 1}^2$			$\mathbf{u}_{N_x - 1}^{N_t - 2}$	$\mathbf{u}_{N_x - 1}^{N_t - 1}$	$\mathbf{u}_{N_x - 1}^{N_t}$	
	$N_x + 1$	$\mathbf{u}_{N_x}^0$	$\mathbf{u}_{N_x}^1$	$\mathbf{u}_{N_x}^2$			$\mathbf{u}_{N_x}^{N_t - 2}$	$\mathbf{u}_{N_x}^{N_t - 1}$	$\mathbf{u}_{N_x}^{N_t}$	

Algorithm Fonction **HEAT1DEX** (version non vectorisée)

```
1: Fonction  $[\mathbf{t}, \mathbf{x}, \mathbb{U}] \leftarrow \text{HEAT1DEX} ( a, b, T, D, f, u_0, \alpha, \beta, N_x, N_t )$ 
2:    $\mathbf{t} \leftarrow \text{DISREG}(0, T, N_t)$ 
3:    $\Delta_t \leftarrow T/N_t$ 
4:    $\mathbf{x} \leftarrow \text{DISREG}(a, b, N_x)$ 
5:    $\Delta_x \leftarrow (b - a)/N_x$ 
6:    $E \leftarrow D \frac{\Delta_t}{\Delta_x^2}, C \leftarrow 1 - 2E$ 
7:   Pour  $i \leftarrow 1$  à  $N_x + 1$  faire ▷ Condition initiale
8:      $\mathbb{U}(i, 1) \leftarrow u_0(\mathbf{x}(i))$ 
9:   Fin Pour
10:  Pour  $n \leftarrow 1$  à  $N_t$  faire ▷ Boucle en temps
11:    Pour  $i \leftarrow 2$  à  $N_x$  faire ▷ Schéma
12:       $\mathbb{U}(i, n+1) \leftarrow C * \mathbb{U}(i, n) + E * (\mathbb{U}(i+1, n) + \mathbb{U}(i-1, n)) + \Delta_t * f(\mathbf{t}(n), \mathbf{x}(i))$ 
13:    Fin Pour
14:     $\mathbb{U}(N_x, n+1) \leftarrow \beta(\mathbf{t}(n+1))$ 
15:     $\mathbb{U}(1, n+1) \leftarrow (\frac{\Delta_x}{D} * \alpha(\mathbf{t}(n+1)) + 4\mathbb{U}(2, n+1) - \mathbb{U}(3, n+1))/3$ 
16:  Fin Pour
17: Fin Fonction
```

Algorithm Fonction **HEAT1DEX** (version vectorisée)

1: Fonction $[\mathbf{t}, \mathbf{x}, \mathbb{U}] \leftarrow \mathbf{HEAT1DEX} (a, b, T, D, f, u_0, \alpha, \beta, N_x, N_t)$
2: $\mathbf{t} \leftarrow \mathbf{DISREG}(0, T, N_t), \Delta_t \leftarrow T/N_t$
3: $\mathbf{x} \leftarrow \mathbf{DISREG}(a, b, N_x), \Delta_x \leftarrow (b - a)/N_x$
4: $E \leftarrow D \frac{\Delta_t}{\Delta_x^2}, C \leftarrow 1 - 2E$
5: $\mathbb{U}(:, 1) \leftarrow u_0(\mathbf{x})$ ▷ Condition initiale
6: $\mathbf{l} \leftarrow [2 : N_x]$ ▷ Indices des points intérieurs
7: **Pour** $n \leftarrow 1$ à N_t **faire** ▷ Boucle en temps
8: $\mathbb{U}(\mathbf{l}, n+1) \leftarrow C * \mathbb{U}(\mathbf{l}, n) - E * (\mathbb{U}(\mathbf{l} + 1, n) + \mathbb{U}(\mathbf{l} - 1, n)) + \Delta_t * f(\mathbf{t}(n), \mathbf{x}(\mathbf{l}))$
9: $\mathbb{U}(N_x, n+1) \leftarrow \beta(\mathbf{t}(n+1))$
10: $\mathbb{U}(1, n+1) \leftarrow (\frac{\Delta_x}{D} * \alpha(\mathbf{t}(n+1)) + 4\mathbb{U}(2, n+1) - \mathbb{U}(3, n+1))/3$
11: **Fin Pour**
12: **Fin Fonction**

```

1 function [t,x,U]=Heat1DexLight(a,b,T,D,f,u0,alpha,beta,Nx,Nt)
2     dt=T/Nt;      t=0:dt:T;
3     dx=(b-a)/Nx; x=[a:dx:b]';
4     U=zeros(Nx+1,Nt+1);
5     E=D*dt/dx^2; C=1-2*E;
6     I=2:Nx;
7     U(:,1)=u0(x);
8     for n=1:Nt
9         U(I,n+1)=C*U(I,n) + E*(U(I+1,n)+U(I-1,n)) + ...
10            dt*f(t(n),x(I));
11        U(Nx+1,n+1)=beta(t(n+1));
12        U(1,n+1)=((2*dx/D)*alpha(t(n+1)) + 4*U(2,n+1) - ...
13            U(3,n+1))/3;
14    end
15 end

```

Listing: fonction pour la résolution de l'EDP de la chaleur (4.81)-(4.84) par le schéma explicite (4.107)-(4.109)

Application avec solution exacte

On choisi

$$u(t, x) \stackrel{\text{def}}{=} \cos(t)\cos(x)$$

et en injectant dans l'EDP on obtient

$$f(t, x) \stackrel{\text{def}}{=} D \cos(t) \cos(x) - \sin(t) \cos(x),$$

$$u_0(x) \stackrel{\text{def}}{=} \cos(x),$$

$$\alpha(t) \stackrel{\text{def}}{=} D \cos(t) \sin(a),$$

$$\beta(t) \stackrel{\text{def}}{=} \cos(t) \cos(b).$$

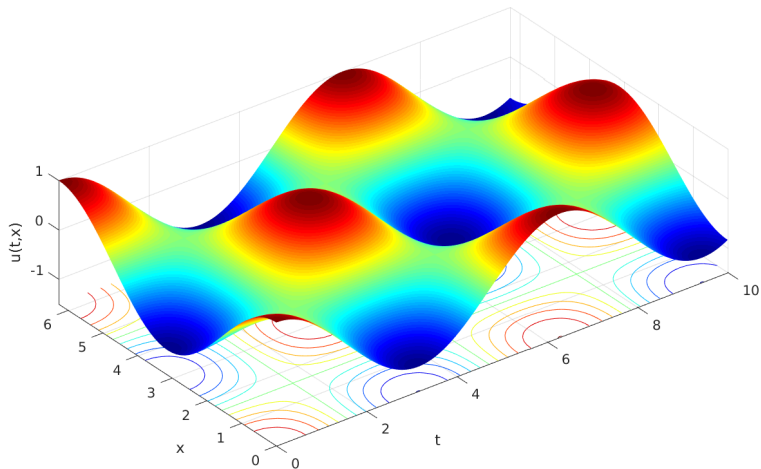
Calcul numérique avec

$$D = 1, a = 0, b = 2\pi, T = 10, N_x = 50 \text{ et } N_t = 5000$$

```
1 a=0;b=2*pi;Nx=50;
2 T=10;Nt=5000
3 D=1;
4 u=@(t,x) cos(t).*cos(x); % solution exacte
5 f=@(t,x) D*cos(t).*cos(x) - sin(t).*cos(x);
6 beta=@(t) u(t,b);
7 alpha=@(t) D*cos(t).*sin(a);
8 u0=@(x) cos(x);
9 [t,x,U]=Heat1DexLight(a,b,T,D,f,u0,alpha,beta,Nx,Nt);
10 surf(t,x,U)
11 xlabel('t'),ylabel('x'),zlabel('u(t,x)')
12 shading interp, colormap(jet)
13 axis image
```

Listing: Equation de la chaleur 1D avec solution exacte. Représentation de la solution calculée. : code Matlab

Application avec solution exacte: numerical solution $N_x = 50$

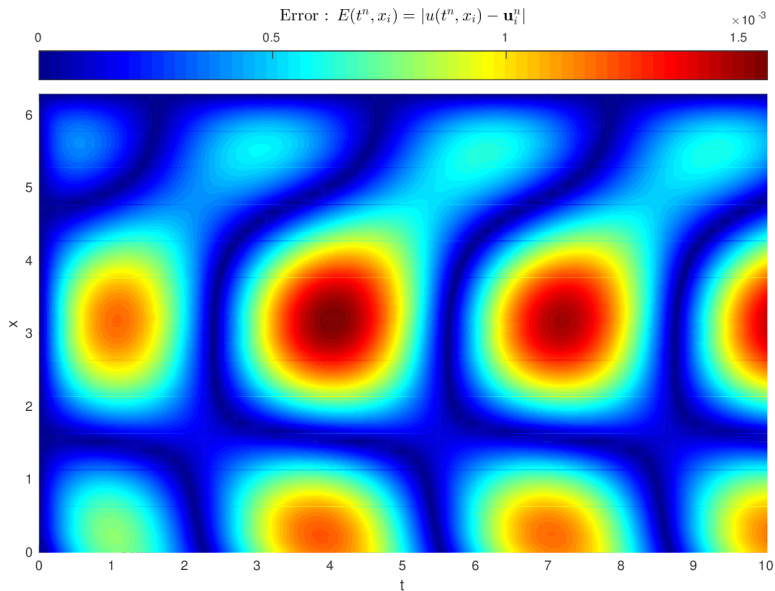


Application avec solution exacte: error $N_x = 50$

```
1 [T,X]=meshgrid(t,x);
2 Uex=u(T,X);
3 figure(2)
4 pcolor(t,x,abs(U-Uex))
5 xlabel('t'),ylabel('x')
6 shading interp, colormap(jet)
7 axis image
8 h=colorbar('Location','northoutside');
9 set(get(h,'title'),'string','Error $\square$ : $\square$ ...
    '$E(t^n,x_i)=|u(t^n,x_i)-\mathbf{u}_i^n|$', ...
    'interpreter','latex','fontsize',12)
```

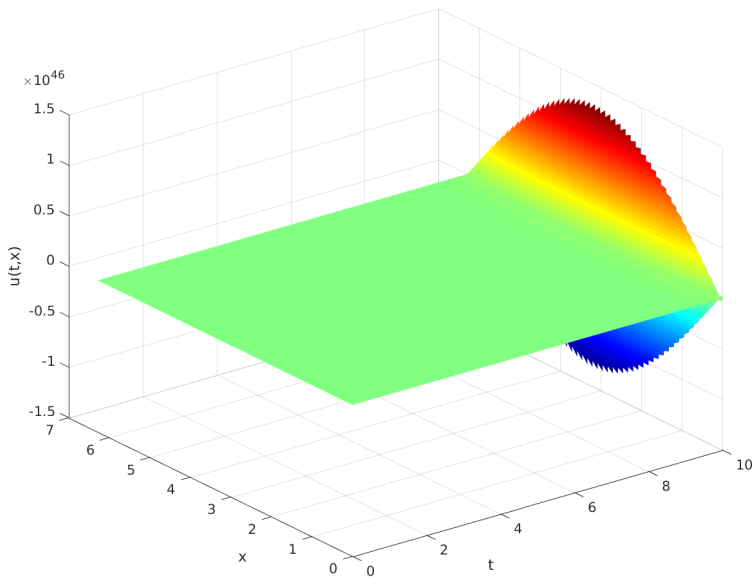
Listing: Equation de la chaleur 1D avec solution exacte. Représentation de l'erreur : code Matlab

Application avec solution exacte: error $N_x = 50$



On change uniquement $N_x : 50 \rightarrow 100$!
A quoi peut-on s'attendre?

Application avec solution exacte: numerical solution $N_x = 100$



Application avec solution exacte: Phénomène d'instabilité

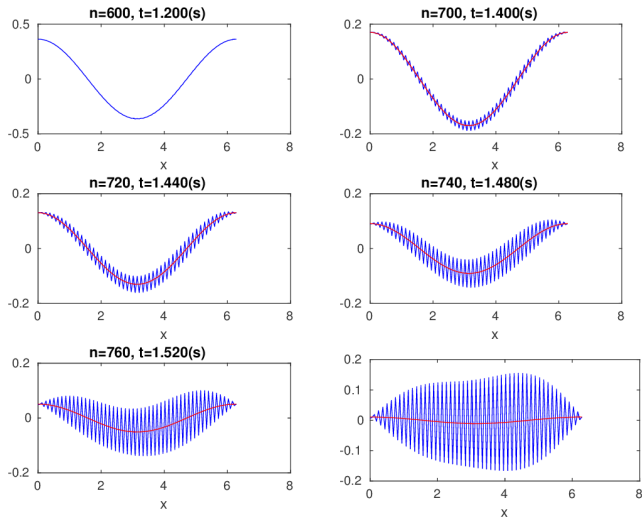


Figure: Equation de la chaleur 1D avec solution exacte (en rouge). Debut du phénomène d'instabilité.

Etude de la stabilité au sens de Von Neumann du schéma explicite donne la **condition de C.F.L.** (R. Courant, K. Friedrichs, and H. Lewy en 1928): le schéma est stable si

$$D \frac{\Delta_t}{dx^2} \leq \frac{1}{2}. \quad (4.110)$$

Illustration: représenter en échelle logarithmique l'erreur commise en fonction de Δ_x et Δ_t calculée par

$$E(\Delta_x, \Delta_t) = \max_{i \in \llbracket 0, N_x \rrbracket, n \in \llbracket 0, N_t \rrbracket} |u_i^n - u_{\text{ex}}(t^n, x_i)|$$

où u_{ex} est la solution exacte.

Application avec solution exacte: Phénomène d'instabilité

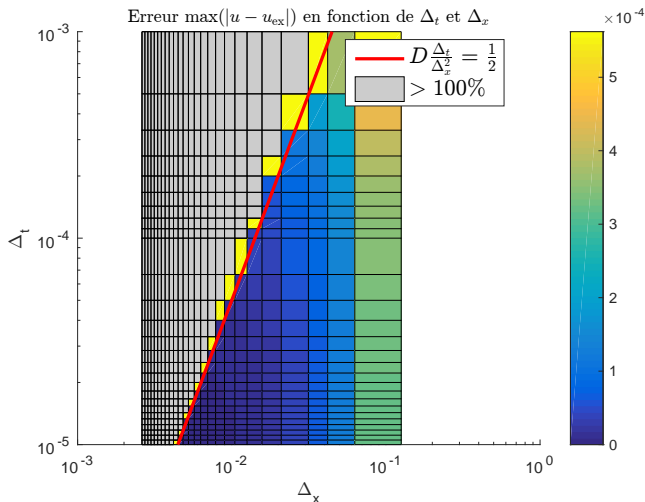


Figure: Equation de la chaleur 1D avec solution exacte. Condition de CFL.

- 1 Exemples d'E.D.P.
 - Equation de Laplace/Poisson
 - Equation de la chaleur
 - Equation des ondes
- 2 Méthodes de résolution numérique d'EDP
- 3 Méthode des différences finies 1D
 - EDP stationnaire 1D + Dirichlet
 - EDP stationnaire + CL mixtes
- 4 **Problème modèle évolutif**
 - Schéma explicite
 - **Schéma implicite**



EDP modèle d'évolution en dimension 1 : équation de la chaleur, formulation aux points de discrétisation

Trouver $u(t^n, x_i) \in \mathbb{R}$, $\forall n \in \llbracket 0, N_t \rrbracket$, $\forall i \in \llbracket 0, N_x \rrbracket$, tels que

$$\frac{\partial u}{\partial t}(t^n, x_i) - D \frac{\partial^2 u}{\partial x^2}(t^n, x_i) = f(t^n, x_i), \quad (4.81)$$

$$u(t^0, x_i) = u_0(x_i), \quad \forall i \in \llbracket 0, N_x \rrbracket, \quad (4.82)$$

$$-D \frac{\partial u}{\partial x}(t^n, x_0) = \alpha(t^n), \quad \forall n \in \llbracket 0, N_t \rrbracket \quad (4.83)$$

$$u(t^n, x_{N_x}) = \beta(t^n), \quad \forall n \in \llbracket 0, N_t \rrbracket \quad (4.84)$$

Pour approcher $\frac{\partial u}{\partial t}(t^n, x_i)$, on utilise cette fois

$$\frac{\partial u}{\partial t}(t^n, x_i) = \frac{u(t^n, x_i) - u(t^{n-1}, x_i)}{\Delta t} + \mathcal{O}(\Delta t). \quad (4.89)$$

Schéma implicite en temps pour l'EDP (4.81) à (4.84)

Un schéma numérique d'ordre 1 en temps et d'ordre 2 en espace pour (4.81): $\forall n \in \llbracket 0, N_t \rrbracket, \forall i \in \llbracket 0, N_x \rrbracket$

$$\frac{\mathbf{u}_i^n - \mathbf{u}_i^{n-1}}{\Delta_t} - D \frac{\mathbf{u}_{i+1}^n - 2\mathbf{u}_i^n + \mathbf{u}_{i-1}^n}{\Delta_x^2} = \mathbf{f}_i^n, \quad (4.113)$$

avec $\mathbf{f}_i^n = f(t^n, x_i)$ et (en espérant) $\mathbf{u}_i^n \approx u(t^n, x_i)$.

(4.116) est équivalent à

$$\mathbf{u}_i^n - D \frac{\Delta_t}{\Delta_x^2} (\mathbf{u}_{i+1}^n - 2\mathbf{u}_i^n + \mathbf{u}_{i-1}^n) = \mathbf{u}_i^{n-1} + \Delta_t \mathbf{f}_i^n. \quad (4.114)$$

Ce schéma est **implicite en temps** : il n'est pas possible de calculer explicitement \mathbf{u}_i^n en fonction des \mathbf{u}_i^{n-1} (au temps précédent).

Schéma implicite en temps pour l'EDP (4.81) à (4.84)

Le calcul des \mathbf{u}_i^n , $\forall i \in \llbracket 0, N_x \rrbracket$, est possible en résolvant les $N_x + 1$ équations **linéaires** suivantes où l'on note $E = D \frac{\Delta_t}{\Delta_x^2}$ et $C = 1 + 2E$:

$$3\mathbf{u}_0^n - 4\mathbf{u}_1^n + \mathbf{u}_2^n = 2 \frac{\Delta_x}{D} \alpha(t^n). \quad (4.115)$$

$$C\mathbf{u}_i^n - E(\mathbf{u}_{i+1}^n + \mathbf{u}_{i-1}^n) = \mathbf{u}_i^{n-1} + \Delta_t \mathbf{f}_i^n, \quad \forall i \in \llbracket 0, N_x \rrbracket \quad (4.116)$$

$$\mathbf{u}_{N_x}^n = \beta(t^n), \quad (4.117)$$

Schéma implicite en temps pour l'EDP (4.81) à (4.84)

$$\mathbb{A} \mathbf{U}^n = \mathbf{b}^n \quad (4.118)$$

avec $\mathbb{A} \in \mathcal{M}_{N_x+1}(\mathbb{R})$, $\mathbf{b}^n \in \mathbb{R}^{N_x+1}$ et

$$\mathbf{U}^n = \begin{pmatrix} \mathbf{u}_0^n \\ \mathbf{u}_1^n \\ \vdots \\ \mathbf{u}_{N_x-1}^n \\ \mathbf{u}_{N_x}^n \end{pmatrix} \in \mathbb{R}^{N_x+1}, \quad \text{i.e. } \mathbf{U}^n(i) = \mathbf{u}_{i-1}^n, \quad \forall i \in \llbracket 1, N_x + 1 \rrbracket$$

$$\mathbb{A} \mathbf{U}^n = \mathbf{b}^n \iff$$

$$\left(\begin{array}{cccc|cccc} 3 & -4 & 1 & 0 & \dots & 0 & 0 & 0 \\ -E & C & -E & 0 & \dots & 0 & 0 & 0 \\ \hline 0 & -E & C & -E & \ddots & \vdots & \vdots & \vdots \\ \vdots & 0 & \ddots & \ddots & \ddots & 0 & \vdots & \vdots \\ \vdots & \vdots & \ddots & -E & C & -E & 0 & 0 \\ 0 & 0 & \dots & 0 & -E & C & -E & 0 \\ \hline 0 & 0 & \dots & 0 & 0 & 0 & 0 & 1 \end{array} \right) \begin{pmatrix} u_0^n \\ u_1^n \\ u_2^n \\ \vdots \\ u_{N_x-2}^n \\ u_{N_x-1}^n \\ u_{N_x}^n \end{pmatrix} = \begin{pmatrix} 2\frac{\Delta_x}{D}\alpha(t^n) \\ \hline u_1^{n-1} + \Delta_t f_1^n \\ u_2^{n-1} + \Delta_t f_2^n \\ \vdots \\ u_{N_x-2}^{n-1} + \Delta_t f_{N_x-2}^n \\ u_{N_x-1}^{n-1} + \Delta_t f_{N_x-1}^n \\ \hline \beta(t^n) \end{pmatrix} \quad (4.119)$$

L'algorithme formel est donc le suivant :

- 1: $\mathbf{U}_i^0 \leftarrow u_0(x_i), \forall i \in \llbracket 0, N_x \rrbracket$
- 2: **Pour** $n \leftarrow 1$ à N_t **faire**
- 3: Calcul de \mathbf{b}^n
- 4: Résoudre le système linéaire $\mathbb{A} \mathbf{U}^n = \mathbf{b}^n$
- 5: **Fin Pour**



Exercice 8.2

Q.1 Ecrire la fonction `ASSEMBLEMATGEN1D` retournant la matrice $M \in \mathcal{M}_d(\mathbb{R})$ définie par

$$M = \begin{pmatrix} a_1 & a_2 & a_3 & 0 & \dots & \dots & 0 \\ \beta & \alpha & \beta & 0 & \dots & \dots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & \dots & 0 & \beta & \alpha & \beta \\ 0 & \dots & \dots & 0 & b_3 & b_2 & b_1 \end{pmatrix} \quad (4.120)$$

où $\alpha, \beta, a_1, a_2, a_3, b_1, b_2$ et b_3 sont des réels donnés.

Q.2 Ecrire la fonction `SNDMBRGEN1D` retournant le vecteur $B \in \mathbb{R}^d$ défini par

$$B = \begin{pmatrix} \alpha \\ c_1 \\ \vdots \\ c_{d-2} \\ \beta \end{pmatrix} \quad (4.121)$$

où $\alpha, \beta, c_1, \dots, c_{d-2}$ sont des réels donnés.



Exercice 8.3

Q.1 *Ecrire une fonction algorithmique **HEAT1DIM** permettant de retourner la discrétisation en temps, la discrétisation en espace et l'ensemble des \mathbf{u}^n , $i \in \llbracket 0, N_x \rrbracket$, $n \in \llbracket 1, N_t \rrbracket$ calculés par le schéma **implicite** en temps pour l'EDP (4.81) à (4.84).*

Q.2 *Ecrire un programme utilisant cette fonction et permettant de calculer la solution numérique d'un problème dont on connaît la solution exacte.*

Algorithm Fonction **HEAT1DIM** (version non vectorisée)

```
1: Fonction  $[\mathbf{t}, \mathbf{x}, \mathbb{U}] \leftarrow \mathbf{HEAT1DIM} ( a, b, T, D, f, u_0, \alpha, \beta, N_x, N_t )$ 
2:    $\mathbf{t} \leftarrow \mathbf{DISREG}(0, T, N_t)$ 
3:    $\Delta_t \leftarrow T/N_t$ 
4:    $\mathbf{x} \leftarrow \mathbf{DISREG}(a, b, N_x)$ 
5:    $\Delta_x \leftarrow (b - a)/N_x$ 
6:    $E \leftarrow D \frac{\Delta_t}{\Delta_x^2}, C \leftarrow 1 + 2E$ 
7:   Pour  $i \leftarrow 1$  à  $N_x + 1$  faire ▷ Condition initiale
8:      $\mathbb{U}(i, 1) \leftarrow u_0(\mathbf{x}(i))$ 
9:   Fin Pour
10:   $\mathbb{A} \leftarrow \mathbf{ASSEMBLEMATGEN1D}(d, (3, -4, 1), (1, 0, 0), C, -E)$ 
11:  Pour  $n \leftarrow 1$  à  $N_t$  faire ▷ Boucle en temps
12:    Pour  $i \leftarrow 1$  à  $N_x - 2$  faire ▷ Schéma
13:       $\mathbf{v}(i) \leftarrow \mathbb{U}(i + 1, n) + \Delta_t f(\mathbf{t}(n), \mathbf{x}i)$ 
14:    Fin Pour
15:     $L \leftarrow 2(\Delta_x/D)\alpha(\mathbf{t}(n))$ 
16:     $\mathbf{B} \leftarrow \mathbf{SNDMBRGEN1D}(d, \mathbf{v}, L, \beta(\mathbf{t}(n)))$ 
17:     $\mathbb{U}(:, n + 1) \leftarrow \mathbf{SOLVE}(\mathbb{A}, \mathbf{B})$ 
18:  Fin Pour
19: Fin Fonction
```

Algorithm Fonction **HEAT1DIM** (version vectorisée)

- 1: **Fonction** $[t, x, U] \leftarrow \mathbf{HEAT1DIM} (a, b, T, D, f, u_0, \alpha, \beta, N_x, N_t)$
- 2: $t \leftarrow \mathbf{DISREG}(0, T, N_t)$
- 3: $\Delta_t \leftarrow T/N_t$
- 4: $x \leftarrow \mathbf{DISREG}(a, b, N_x)$
- 5: $\Delta_x \leftarrow (b - a)/N_x$
- 6: $E \leftarrow D \frac{\Delta_t}{\Delta_x^2}, C \leftarrow 1 + 2E$
- 7: $U(:, 1) \leftarrow u_0(x)$ ▷ Condition initiale
- 8: $A \leftarrow \mathbf{ASSEMBLEMATGEN1D}(d, (3, -4, 1), (1, 0, 0), C, E)$
- 9: **Pour** $n \leftarrow 1$ à N_t **faire** ▷ Boucle en temps
- 10: $v \leftarrow U([2 : N_x], n) + \Delta_t f(t(n), x([2 : N_x]))$
- 11: $L \leftarrow 2(\Delta_x/D)\alpha(t(n))$
- 12: $B \leftarrow \mathbf{SNDMBRGEN1D}(d, v, L, \beta(t(n)))$
- 13: $U(:, n + 1) \leftarrow \mathbf{SOLVE}(A, B)$
- 14: **Fin Pour**
- 15: **Fin Fonction**

Une étude de stabilité au sens de Von Neumann de ce schéma **implicite** permet de montrer qu'il est **inconditionnellement stable**!