

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHE SCIENTIFIQUE

Université M^{ed} Khider, Biskra

Faculté des sciences exactes, des sciences de la nature et de la vie

Département de mathématiques



APPROXIMATION NUMÉRIQUE DES EQUATIONS
DIFFÉRENTIELLES ORDINAIRES
ET PARTIELLES

D^r Nacer KHELIL

Année 2011/2012

Contents

Table des matières	iii
Liste des tableaux	iv
Table des figures	v
1 Approximation numérique des Equations Différentielles Ordinaires	1
1.1 Définitions	1
1.2 Schémas de résolution numérique	2
1.2.1 Méthodes de Taylor	2
1.2.2 Exercices	4
1.2.3 Utilisation de Matlab pour résoudre un (PVI)	7
1.3 Devoir	7
2 Equations aux dérivées partielles	9
2.1 Rappel et complément	9
2.1.1 Définitions	9
2.1.2 Théorème de Taylor	10
2.1.3 Théorème	11
2.1.4 Approximation de dérivées par des différences finies	11
2.1.5 Fonctions de plusieurs variables	12
2.2 L'équation de transport	13
2.3 Équation de la Chaleur	14
2.3.1 Schéma de Crank-Nicolson	15
2.4 Équation de Laplace	17
2.5 Exercices	17
3 Théorème d'Equivalence de Lax	22
3.1 Consistance	22
3.1.1 Exemple	23
3.2 Stabilité	25
3.2.1 Stabilité Matricielle	25

4	Equations aux dérivées partielles dans MATLAB	32
5	LA METHODE DES ELEMENTS FINIS	37

List of Tables

List of Figures

4.1	Mesh plot for solution to Equation 5.1	35
4.2	Solution at $t = 2$	36
5.1	Mesh plot for solution to Equation 5.1	40
5.2	Solution at $t = 2$	41

Chapter 1

Approximation numérique des Equations Différentielles Ordinaires

1.1 Définitions

Une équation différentielle ordinaire (EDO) est une relation entre **une variable** indépendante notée x et une variable dépendante, y , et ses dérivées successives. L'**ordre** de cette équation est déterminé par l'ordre du degré le plus élevé de la dérivation. D'une manière générale, il existe trois types de problèmes importants.

1. Equation différentielle donnée sous forme explicite $y' = f(x, y)$.
2. Les systèmes d'équations différentielles qui s'écrivent sous la forme: $M(x, y)y' = f(x, y)$ où $M(x, y)$ est une matrice, et y' est une variable (vecteur).
3. Equation différentielle ordinaire implicite de la forme $f(x, y, y') = 0$.

Une équation différentielle possède une infinité de solutions. Et pour calculer une solution particulière, nous devons connaître une condition initiale. L'ensemble formé par l'équation différentielle et la condition initiale est appelé **problème aux valeurs initiales** ou **problème de Cauchy**:

$$\begin{cases} y' = f(x, y) \\ y(x_0) = y_0 \end{cases} \quad (PVI) \quad (1.1)$$

Remarque1: Ce problème possède une solution unique lorsque :

1. f est une application définie et continue sur $[x_0, x] \times R$.
2. f est lipschitzienne: il existe une constante $k > 0$ réelle, telle que $|f(x, \theta_1) - f(x, \theta_2)| \leq k |\theta_1 - \theta_2|$, pour tout $x \in [x_0, T]$ et tout couple $(\theta_1, \theta_2) \in R \times R$, (pour la démonstration de ces résultats et l'établissement d'importants théorèmes généraux sur les équations différentielles, voir cours d'analyse).

Remarque2: En réalité les systèmes d'équations différentielles contiennent les équations d'ordre supérieur à 1. Puisque une équation différentielle d'ordre n :

$$y^{(n)} = f(x, y, y', y'', \dots, y^{(n-1)})$$

peut se transformer en un système de n équations en posant :

$$y_1 = y, y_2 = y', \dots, y_n = y^{(n-1)}$$

de manière que nous obtenions le système suivant:

$$y_1' = y_2, y_2' = y_3, \dots, y_n' = f(x, y_1, y_2, \dots, y_n).$$

1.2 Schémas de résolution numérique

1.2.1 Méthodes de Taylor

Une manière simple de construire des méthodes pour la résolution d'un (PVI) est le remplacement de dérivées dans un développement de Taylor, c'est-à-dire en introduisant des expressions telles que:

$$\begin{aligned} y' &= f \\ y'' &= f_x + f_y f \\ y''' &= f_{xx} + 2f_{xy}f + 2f_{yx} + f_{yy}f^2 + f_y^2 f. \end{aligned}$$

dans

$$y_{i+1} = y_i + h \frac{dy}{dx} \Big|_i + \frac{h^2}{2} \frac{d^2y}{dx^2} \Big|_i + \dots + \frac{h^n}{n!} \frac{d^ny}{dx^n} \Big|_i$$

Le choix du nombre de termes après y_i permet la définition d'une grande famille de méthodes avec un **ordre** de précision égale au nombre de termes employés.

1. **Ordre d'une méthode:** La série de Taylor permet aussi la définition de l'ordre d'une méthode à un pas. Plus précisément, on considère

$$y_{i+1} - y_i = hf(x, y) + \frac{h^2}{2} f' + \dots + \frac{h^p}{p!} f^{p-1} + O(h^{p+1})$$

et relatif

$$\Delta(x, y, h) = f(x, y) + \frac{h^2}{2} f' + \dots + \frac{h^p}{p!} f^{p-1} + O(h^p)$$

et on approxime ce dernier par

$$\Phi(x, y, h) = f(x, y) + \frac{h^2}{2} f' + \dots + \frac{h^p}{p!} f^{p-1} \quad (1.2)$$

On dit que la méthode définie par l'Eq (1.2) est une méthode de Taylor d'ordre p .

2. **Erreur de troncature** En général, on ne peut pas calculer l'erreur globale $E_g = y(x_i) - y_i$, car la solution exacte $y(x_i)$ n'est pas connue. Pour cela on calcule l'erreur locale, ce qui permettra le contrôle de l'erreur globale. Le calcul de l'erreur locale, demande la prise en compte de tous les termes que l'on a négligé en prenant une série tronquée:

$$y_{i+1} = y_i + hy'_i + \frac{h^2}{2!} y''_i + \frac{h^{N+1}}{N+1!} y^{N+1}(\xi)$$

Le terme $hT = \frac{h^{N+1}}{N+1!} y^{N+1}(\xi)$, qui représente l'erreur locale de troncature, est exprimée en fonction de ξ avec $x_i < \xi < x_{i+1}$, mais on ignore la position de ξ . Cependant, un estimé de l'erreur peut être obtenu en comparant la solution obtenue par deux méthodes de précision différentes.

Soit, localement, $U(x_{i+1})$ la solution exacte de l'équation $y'(t) = f(x, y)$ à la position x_{i+1} . Cette solution peut être exprimée par

$$U(x_{i+1}) = y_i + hF(x_i, y_i) + hT$$

Où y_i indique la solution à la position x_i , h le pas, $F(x_i, y_i)$ une fonction dépendante de la méthode choisie, et hT l'erreur locale de troncature de la méthode. Puisque y_{i+1} , la valeur numérique qu'approche $U(x_{i+1})$, est donnée par

$$y_{i+1} = y_i + hF(x_i, y_i)$$

on déduit

$$hT = U(x_{i+1}) - y_{i+1} \quad (1.3)$$

De même

$$h\bar{T} = U(x_{i+1}) - \bar{y}_{i+1} \quad (1.4)$$

pour la deuxième méthode \bar{F} . En soustrayant l'équation (1.4) de (1.3) et on négligeant l'erreur de la plus précise on obtient :

$$hT = \bar{y}_{i+1} - y_{i+1}$$

ou bien

$$T = \frac{\bar{y}_{i+1} - y_{i+1}}{h} \quad (1.5)$$

1.2.2 Exercices

Exercice 1: Etant donné un pas de temps h calculer approximativement à l'aide du développement de Taylor à l'ordre 1 les valeurs $y(x_0 + h), y(x_0 + 2h), \dots, y(x_0 + nh)$

1. Ecrire l'algorithme (correspondant à cette méthode) d'Euler
2. pour $\begin{cases} y' = -y(x) + x + 1 \\ y(0) = 1 \end{cases}$ (PVI) faire trois itérations par l'algorithme d'Euler.

Exercice 2: pour le (PVI) $\begin{cases} y' = y(x) \\ y(0) = 1 \end{cases}$ $x \in [0, 1]$ l'erreur de la méthode d'Euler peut être analysé directement.

1. Prouver que la solution approchée s'écrit sous la forme $y_h(x_n) = c(h)^{x_n}$, où $c(h) = (1 + h)^{\frac{1}{h}}$.

2. En utilisant la formule de l'Hospital vérifie que $\lim_{h \rightarrow 0} c(h) = e$, en déduire que pour $x=x_n$ fixé, $\lim_{h \rightarrow 0} y_h(x) = e^x$.
3. Prouver que $\max_{0 \leq x_n \leq 1} |y(x_n) - y_h(x_n)| = e - c(h)$. Que peut-on conclure.

Exercice 3: Rappelons qu'un (PVI) est une relation de la forme $\begin{cases} y' = f(x, y) \\ y(x_0) = y_0 \end{cases}$

1. Intégrer l'éq. précédente de x_n à x_{n+1} en utilisant la méthode des trapèzes.
2. Remplacer dans l'éq.obtenue $y(x_{n+1})$ par $y(x_n) + hf(x_n, y(x_n))$, alors on obtient la formule $y(x_{n+1}) = y(x_n) + \frac{h}{2} [f(x_n, y(x_n)) + f(x_{n+1}, y_n + h(x_n, y(x_n)))]$
3. Ecrire l'algorithme de cette méthode.
4. pour $\begin{cases} y' = -y(x) + x + 1 \\ y(0) = 1 \end{cases}$ (PVI) faire trois itérations par cet algorithme.

Exercice 4: Méthode de Runge-Kutta d'ordre 2 (RK₂)

1. Etant donné un pas de temps h , calculer approximativement à l'aide du développement de Taylor à l'ordre 2 la valeur $y(x_{n+1})$.
2. On propose de remplacer l'expression obtenue par l'expression équivalente $y(x_{n+1}) = y(x_n) + a_1 hf(x_n, y(x_n)) + a_2 hf(x_n + a_3 h, y(x_n) + a_4 h)$. Déterminer les constantes a_1, a_2, a_3 et a_4 de telle sorte que ces deux expressions aient une erreur en $o(h^3)$. (développer la fonction de deux variables $f(x_n + a_3 h, y(x_n) + a_4 h)$ à l'ordre 1 autour du point $(x_n, y(x_n))$).
3. Par analogie des deux expressions, obtenir un système de 3 équations à 4 inconnues (c-à-d ce système ne possède pas une solution unique).
4. Pour un choix convenable des a_i , écrire l'algorithme correspondant.
5. pour $\begin{cases} y' = -y(x) + x + 1 \\ y(0) = 1 \end{cases}$ (PVI) faire trois itérations par l'algorithme de RK₂.

Exercice 5: Méthode de Runge-Kutta d'ordre 4 (RK₄)

En reprenant le développement de Taylor de $y(x)$ mais cette fois j'usqu'à l'ordre 5, un

raisonnement similaire à celui qui a mené aux méthodes RK_2 aboutit à un système de 8 équations non linéaires comprenant 10 inconnues. Le résultat final est la méthode de RK_4 , qui représente un outil de grand utilité.

Algorithme RK_4 :

1. étant donné un pas de temps h , une condition initiale (t_0, y_0) et un nombre maximal d'itérations N .
2. Pour $0 \leq n \leq N$.

$$k_1 = hf(x_n, y_n)$$

$$k_2 = hf((x_n + h/2, y_n + k_1/2)$$

$$k_3 = hf((x_n + h/2, y_n + k_2/2)$$

$$k_4 = hf(x_n + h, y_n + k_3)$$

$$y_{n+1} = y_n + 1/6(k_1 + 2k_2 + 2k_3 + k_4)$$

$$x_{n+1} = x_n + h$$
 Ecrire (x_{n+1}, y_{n+1})
3. Arrêt
4. pour $\begin{cases} y' = -y(x) + x + 1 \\ y(0) = 1 \end{cases}$ (PVI) faire trois itérations par l'algorithme de RK_4 .

Remarque 3: Les méthodes de Runge-Kutta peuvent être associées à la quadrature numérique. Par exemple, l'intégrale: $y_{i+1} - y_i = \int_{x_i}^{x_{i+1}} f(x, y) dy$ peut être approximée en tenant compte de la fonction à la position f_i , ce qui mène à la méthode d'Euler explicite. Elle peut aussi être approximée à la position f_{i+1} , ce qui conduit à la méthode d'Euler implicite. Finalement, on peut considérer une moyenne de ces deux quantités, ce qui vient à considérer une quadrature selon la règle des trapèzes (1.2.2). De la même manière, la méthode de Runge-Kutta d'ordre 4 peut être regardée à partir de la formule de quadrature de Simpson

Exercice 6 (Stabilité): Considérons l'équation modèle

$$y'(x) = -\alpha y$$

et un schéma à un pas,

$$y_{n+1} = G(-\alpha h)y_n = \dots = [G(-\alpha h)]^{n+1}y_0$$

où G est la fonction d'*amplification* qui représente le comportement de la solution exacte

$$y(x_{n+1}) = \exp(-\alpha h)y(x_n)$$

Notant $z = -\alpha h$, **vérifier** que $G(z) = (1 + z)$ pour la méthode d'Euler, $G(z) = 1 + z + z^2/2$ pour la méthode RK_2 et $G(z) = 1 + z + z^2/2 + z^3/6 + z^4/24$ pour la méthode RK_4 .

Définition Le lieu géométrique de points $z \in \mathbb{C}$ pour lesquels $|G(z)| < 1$ s'appelle la **zone de stabilité** (absolue) du schéma.

1.2.3 Utilisation de Matlab pour résoudre un (PVI)

Remarque 4: Nous pouvons vérifier la solution analytique des exercices avec la fonction `dsolve` de MATLAB, en utilisant le code suivant:

```
syms x y z
z = dsolve('Dy = -y + x + 1','y0 = 1','x')
alors on obtient: z = x + exp(-x)
```

Remarque 5: MATLAB possède plusieurs (solvers) fonctions pour calculer les solutions numériques des équations différentielles Ordinaires. La syntaxe la plus simple est : `[t,y]=solver ('f',tspan,y0)`, où **solver** = **ode23** pour les méthodes RK_2 et RK_3 par exemple, Le premier argument, **f**, entre guillemets, est le nom de la fonction MATLAB défini par l'utilisateur, **tspan** définit l'intervalle d'intégration et **y0**, représente la condition initiale qui assure l'unicité de la solution. Cette fonction nous fournit deux ensembles de valeurs le premier représente les t et l'autre leurs évaluations.

1.3 Devoir

Considérons le problème de Cauchy suivant

$$u' + \alpha u(t) = f(t), \quad \forall t > 0 \quad u(0) = u_0 \quad (1.6)$$

où $\alpha \in \mathbb{R}$ est une constante donnée.

1. Poser $u(t) = e^{-\alpha t}v(t)$ et écrire le pvi vérifié par v . Vérifier que la solution de (1.6) est donnée par

$$u(t) = e^{-\alpha t}(u_0 + \int_0^t e^{\alpha s} f(s) ds)$$

2. Quelle est la solution exacte pour $\alpha = \alpha(t)$?
3. Supposant que α et f soient constantes, calculer la $\lim_{t \rightarrow \infty} u(t)$.
4. Ecrire un code Matlab qui résout (1.6) par le schéma explicite d'Euler. Utiliser $\alpha = 4, u(0) = 1, 0 \leq t \leq 3$ et $h = 1/8$. Tracer les résultats, solution exacte et numérique superposées. Comparer avec $h = 1/2$
5. Utiliser une méthode Runge-Kutta d'ordre quatre et commenter le résultat obtenu pour $h = 1/2$. Tracer les zones de stabilité

```

alpha=4 ; t0=0 ; T initial
t1=3 ; T final
u0=1 ; condition initiale
pas de discretisation 1/8
dt = 1/8 ; np1 = t1/dt ;
tp1 = linspace(t0 ,t1,np1+1) ;
u = zeros(1,np1+1) ; initialiser u
u(1) = u0 ; debuter avec la condition initiale
t = t0 ;
for i=1 :np1
    u(i+1) = u(i) + dt*(-alpha *u(i)) ;
    t = t + dt ;
end
solution exacte
sole = exp(-alpha *tp1)*u0 ;
tracer
plot(tp1,u , 'r-.',tp1,sole,'b- ', 'LineWidth',2) ;
legend('Euler', 'Solution exacte') ; xlabel('temps') ; ylabel('u') ; title('Euler Explicite')

```

Chapter 2

Equations aux dérivées partielles

2.1 Rappel et complément

2.1.1 Définitions

Une équation aux dérivées partielles (EDP), est une classe d'équations différentielles possédant au moins deux variables indépendantes. Dans ce chapitre, nous considérons l'équation du second ordre de deux variables indépendantes x et y , qui s'écrit sous la forme:

$$A(x, y) \frac{\partial^2 u}{\partial x^2} + B(x, y) \frac{\partial^2 u}{\partial x \partial y} + C(x, y) \frac{\partial^2 u}{\partial y^2} = f(x, y, u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}) \quad \text{pour} \\ x_0 \leq x \leq x_f, y_0 \leq y \leq y_f \quad (2.1)$$

Avec les conditions aux limites données par:

$$u(x, y_0) = b_{y0}(x), u(x, y_f) = b_{yf}(x) \quad \text{et} \\ u(x_0, y) = b_{x0}(y), u(x_f, y) = b_{xf}(y) \quad (2.2)$$

Ces équations EDP sont classées en trois groupes, Elles s'obtiennent en appliquant les résultats de L'Algèbre concernant les formes quadratiques

1. EDP elliptique : si $B^2 - 4AC < 0$, l'une des équations fondamentales de la physique est l'équation de Laplace $\frac{\partial^2 u}{\partial^2 t} + \frac{\partial^2 u}{\partial^2 x} = 0$
2. EDP parabolique : si $B^2 - 4AC = 0$, l'exemple typique de cette famille est l'équation de la chaleur $\frac{\partial u}{\partial t} - \alpha \frac{\partial^2 u}{\partial^2 x} = 0$

3. EDP hyperbolique : si $B^2 - 4AC > 0$ Un exemple des équations hyperboliques linéaires est l'équation des cordes vibrantes $\frac{1}{c^2} \frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} = 0$

Ces trois types d'EDP sont associés aux états d'équilibres, états de diffusions et systèmes oscillants, respectivement. Nous étudierons des méthodes numériques pour résoudre ces EDP, puisque leurs solutions analytiques sont habituellement difficiles à trouver.

Remarque: Pour passer d'un problème exact continu régité par une EDP au problème approché discret, il existe trois grandes familles de méthodes.

1. **Les différences finies** La méthode consiste à remplacer les dérivées partielles par des différences divisées ou combinaisons de valeurs ponctuelles de la fonction en un nombre fini de points discrets ou noeuds du maillage.
2. **Les volumes finis** La méthode intègre, sur des volumes élémentaires de forme simple, les équations écrites sous forme de loi de conservation. Elle fournit ainsi de manière naturelle des approximations discrètes conservatives et est particulièrement bien adaptée aux équations de la mécanique des fluides. Sa mise en oeuvre est simple avec des volumes élémentaires rectangles.
3. **Les éléments finis** La méthode consiste à approcher, dans un sous-espace de dimension finie, un problème écrit sous forme variationnelle (comme minimisation de l'énergie en général) dans un espace de dimension infinie. La solution approchée est dans ce cas une fonction déterminée par un nombre fini de paramètres comme, par exemple, ses valeurs en certains points ou noeuds du maillage.

2.1.2 Théorème de Taylor

Soit $f \in C^n$ dans un intervalle I autour d'un point x_0 : Parmi tous les polynômes de degré n il existe un seul dont les dérivées d'ordre inférieure ou égale à n sont toutes égales aux dérivées de f : Nous définissons le polynôme de Taylor comme

$$P^k(x_0) = f^k(x_0), \quad k = 0, 1, \dots, n \quad (2.3)$$

Lorsque $n = 1$, P est la ligne tangente à f au point x_0

2.1.3 Théorème

Si $f \in C^{n+1}$ dans un voisinage I de x_0 alors pour tout $x \in I$

$$\begin{aligned} f(x) &= \sum_{k=0}^{\infty} f^{(k)}(x_0) \frac{(x-x_0)^k}{k!} \\ &= f(x_0) + f'(x_0)(x-x_0) + f''(x_0) \frac{(x-x_0)^2}{2!} \\ &\quad + \dots + f^{(n)}(x_0) \frac{(x-x_0)^n}{n!} + R_n(x) \end{aligned} \quad (2.4)$$

où $R_n(x) = f^{(n+1)}(\xi) \frac{(x-x_0)^{n+1}}{(n+1)!} = O((x-x_0)^{n+1})$ et $x < \xi < x_0$.

Démonstration: Ce théorème est une généralisation du théorème de la valeur moyenne...

2.1.4 Approximation de dérivées par des différences finies

Lorsqu'une fonction u et ses dérivées sont des fonctions bornées et régulières de x , alors selon le théorème de Taylor nous avons les développements, en amont («forward»).

$$u(x+h) = u(x) + hu'(x) + \frac{h^2}{2}u''(x) + \frac{h^3}{6}u'''(x) + \dots \quad (2.5)$$

et en aval («backward»)

$$u(x-h) = u(x) - hu'(x) + \frac{h^2}{2}u''(x) - \frac{h^3}{6}u'''(x) + \dots \quad (2.6)$$

Leur somme est

$$u(x+h) + u(x-h) = 2u(x) + h^2u''(x) + O(h^4)$$

Nous pouvons donc approcher

$$u''(x) \approx \frac{1}{h^2}[u(x+h) - 2u(x) + u(x-h)]$$

avec une erreur d'ordre h^2 . Prenant la différence, nous obtenons.

$$u'(x) \approx \frac{1}{2h}[u(x+h) - u(x-h)]$$

avec une erreur d'ordre h^2 aussi. Ce sont des approximations par des différences centrées. Nous pouvons également approcher par des différences avant

$$u'(x) \approx \frac{1}{h}[u(x+h) - u(x)]$$

ou des différences arrière

$$u'(x) \approx \frac{1}{h}[u(x) - u(x-h)]$$

avec une erreur d'ordre h .

2.1.5 Fonctions de plusieurs variables

Supposons que $u = u(x, t)$. Nous divisons le plan $x - t$ en rectangles par des lignes de grille aux points

$$x_i = ih \quad i = 0, 1, \dots, I$$

et

$$t_n = nk \quad n = 0, 1, \dots, J$$

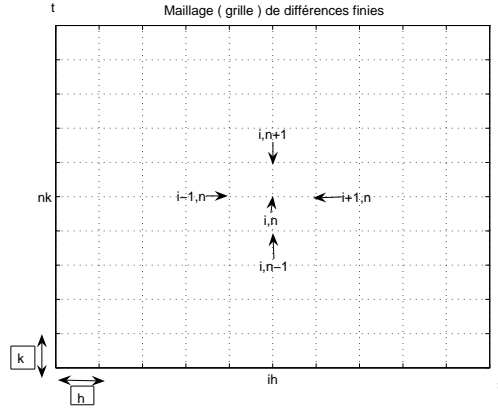
Nous notons alors

$$u(x_i, t_n) = u(ih, nk) = u_{i,n} = u_i^n$$

Les dérivées partielles deviennent, par exemple,

$$\frac{\partial u}{\partial t} \approx \frac{u_i^{n+1} - u_i^n}{k}$$

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{h^2}$$



2.2 L'équation de transport

$$\frac{\partial u}{\partial t} + c \frac{\partial u}{\partial x} = 0 \quad (2.7)$$

Remarque :

Pour démontrer que cette équation est une équation hyperbolique, il suffit de démontrer que l'équation des ondes $\frac{\partial^2 u}{\partial t^2} - c^2 \frac{\partial^2 u}{\partial x^2} = 0$ peut se ramener à une équation de transport. Hint: faire le changement de variable $v = \frac{\partial u}{\partial t}$ et $w = \frac{\partial u}{\partial x}$.

Plusieurs approches peuvent être utiliser pour discrétiser cette équation :

1. Nous utilisons un schéma avant d'ordre 1 pour évaluer la dérivée temporelle et un schéma avant d'ordre 1 pour la dérivée première en espace; Euler décentré aval (downwind)

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + c \frac{u_{j+1}^n - u_j^n}{\Delta x} = 0 \quad (2.8)$$

2. Nous utilisons un schéma avant d'ordre 1 pour évaluer la dérivée temporelle et un schéma arrière d'ordre 1 pour la dérivée première en espace; Euler décentré amont (upwind)

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + c \frac{u_j^n - u_{j-1}^n}{\Delta x} = 0 \quad (2.9)$$

3. Nous utilisons un schéma avant d'ordre 1 pour évaluer la dérivée temporelle et un schéma centré d'ordre 1 pour la dérivée première en espace; FTCS (forward in time, centered in space)

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + c \frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} = 0 \quad (2.10)$$

4. Nous utilisons un schéma centré d'ordre 1 pour évaluer la dérivée temporelle et un schéma centré d'ordre 1 pour la dérivée première en espace; Saut-mouton (leap-frog)

$$\frac{u_j^{n+1} - u_j^{n-1}}{2\Delta t} + c \frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} = 0 \quad (2.11)$$

2.3 Équation de la Chaleur

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} \quad (2.12)$$

Cette équation peut s'interpréter comme celle qui décrit l'évolution de la température d'une barre dans le temps, causée par l'absorption d'un flux de chaleur. Avec ce modèle, on cherche à prédire la température en tout point à tout instant à partir d'une distribution initiale u_0 .

Que nous approchons par

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

ce qui donne la formule de récurrence explicite (schéma FTCS)

$$u_i^{n+1} = ru_{i-1}^n + (1 - 2r)u_i^n + ru_{i+1}^n \quad (2.13)$$

avec $r = k/h^2$, qui peut s'écrire sous la forme matricielle;

$$U^{n+1} = AU^n$$

Avec

$$\mathbf{A} = \begin{pmatrix} 1 - 2r & r & & \\ r & \ddots & \ddots & \\ & \ddots & \ddots & r \\ & & r & 1 - 2r \end{pmatrix}$$

Une analyse de stabilité (voir plus bas) révèle une valeur limite pour r : Afin de remédier à cette restriction, nous proposons le schéma- θ suivant :

$$\frac{\partial u}{\partial t} = \frac{u_j^{n+1} - u_j^n}{\Delta t} + o(\Delta t)$$

$$\frac{\partial u^2}{\partial x^2} = (1 - \theta) \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{\Delta x^2} + \theta \frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}}{\Delta x^2} + o(\Delta x^2)$$

avec $0 \leq \theta \leq 1$, ce qui donne

$$\frac{u_i^{n+1} - u_i^n}{k} = \theta \left[\frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}}{h^2} \right] + (1 - \theta) \left[\frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{h^2} \right]$$

ou

$$-r\theta u_{i-1}^{n+1} + (1 + 2r\theta)u_i^{n+1} - r\theta u_{i+1}^{n+1} = r(1 - \theta)u_{i-1}^n + (1 - 2r(1 - \theta))u_i^n + r(1 - \theta)u_{i+1}^n$$

Nous remarquons les faits suivants :

1. Le schéma est maintenant explicite-implicite et nous devons résoudre un système linéaire à chaque pas de temps.
2. Lorsque $\theta = 0$; nous retrouvons le schéma explicite (les u_i^{n+1} se déduisent directement des u_i^n).
3. Lorsque $\theta = 1$; le schéma est dit totalement implicite.
4. Lorsque $\theta = 1/2$; le schéma est dit de Crank-Nicolson.

Les conditions de stabilité pour ce schéma sont

$$r \leq \frac{0.5}{1-2\theta} \quad si \quad 0 \leq \theta \leq \frac{1}{2}$$

$$0 < r < \infty \quad si \quad \frac{1}{2} \leq \theta \leq 1$$

2.3.1 Schéma de Crank-Nicolson

Ce schéma implicite-explicite peut être écrit comme,

$$-ru_{i-1}^{n+1} + (2 + 2r)u_i^{n+1} - ru_{i+1}^{n+1} = ru_{i-1}^n + (2 - 2r)u_i^n + ru_{i+1}^n$$

où à chaque pas de temps, il faut résoudre un système linéaire tridiagonal :

$$CU^{n+1} = DU^n$$

où

$$C = \begin{bmatrix} 2+2r & -r & . & . & . \\ -r & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & -r \\ . & . & . & -r & 2+2r \end{bmatrix}$$

$$D = \begin{bmatrix} 2-2r & r & . & . & . \\ r & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & r \\ . & . & . & r & 2-2r \end{bmatrix}$$

Le schéma est inconditionnellement stable et il est très largement utilisé.

2.4 Équation de Laplace

L'équation de Laplace (ou Poisson) décrit des états stationnaires. La solution est déterminée par les conditions aux bords. Elle prend la forme:

$$\frac{\partial u^2}{\partial x^2} + \frac{\partial u^2}{\partial y^2} = 0 \quad (2.14)$$

Que nous approchons par

$$\frac{u_i^{n+1} - 2u_i^n + u_i^{n-1}}{k^2} + \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{h^2} = 0$$

Remarque: ce qui donne pour $h = k$

$$\frac{u_i^{n+1} + u_{i+1}^n - 4u_i^n + u_i^{n-1} + u_{i-1}^n}{h^2} = 0$$

A chaque étape, nous remarquons que pour calculer la valeur de u_i^n nous avons besoin de connaître les points u_i^{n+1} , u_{i+1}^n , u_i^{n-1} et u_{i-1}^n comme l'indique le schéma suivant (formule à 5 points):

$$\Delta u = 0 \Leftrightarrow \frac{1}{h^2} \begin{pmatrix} & 1 & & \\ 1 & -4 & 1 & \\ & 1 & & \end{pmatrix} u_i^n = 0$$

2.5 Exercices

Exercice 7

Quel est le type de l'équation $\frac{\partial^2 u}{\partial y^2} + y \frac{\partial^2 u}{\partial x^2} = 0$

Exercice 8 Démontrer les formules suivantes. (Ces formules sont à la base des méthodes de différences finies utilisées pour résoudre les équations aux dérivées partielles.)

$$\begin{aligned} f'(x) &= \frac{f(x+h)-f(x)}{h} + O(h) \\ f'(x) &= \frac{f(x)-f(x-h)}{h} + O(h) \\ f'(x) &= \frac{f(x+h)-f(x-h)}{2h} + O(h^2) \\ f''(x) &= \frac{f(x+h)-2f(x)+f(x-h)}{h^2} + O(h^2) \end{aligned}$$

Exercice 9

On considère le problème :

$$\begin{cases} -u''(x) + \sin(u(x)) = f(x), & 0 < x < 1 \\ u(0) = a, & u(1) = b \end{cases}$$

Ecrire le schéma de différences finies avec pas constant pour ce problème.

Exercice 10

1. Démontrer que la solution de l'équation $\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0$ est toute fonction de la forme $u(t, x) = \phi(x - t)$ où ϕ est une fonction arbitraire.
2. Démontrer que $u(t, x) = u(0, x - t)$.
3. vérifier que si on connaît $u(0, x)$ pour tout $x \in R$, alors on peut calculer $u(t, x)$ pour tout $t \in R$ et pour tout $x \in R$

Exercice 11

Pour un maillage régulier en temps $t_n = n\Delta t, n \in N$ et en espace $x_j = j\Delta x, j \in Z$, on introduit le schéma d'approximation par différences finies où $u_j^n \approx u(x_j, t_n)$:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + \frac{a}{4\Delta x}(u_{j+1}^{n+1} - u_{j-1}^{n+1} + u_{j+1}^n - u_{j-1}^n) = 0, j \in Z \quad n \geq 0$$

où a est une constante réelle strictement positive.

1. Pour quelle équation ce schéma est-il défini (justifiez précisément votre réponse)?
2. Dessinez le stencil du schéma. Le stencil permet-il de déduire une condition nécessaire de stabilité ?

Exercice 12

Le système $AX = Y$ est appelé système tridiagonal lorsque sa matrice a la forme:

$$A = \begin{bmatrix} b_1 & c_1 & . & . & 0 \\ a_2 & b_2 & . & . & . \\ 0 & . & . & . & . \\ . & . & . & . & c_{n-1} \\ 0. & . & 0. & a_n & b_n \end{bmatrix}$$

Ecrire un algorithme de résolution d'un système tridiagonal $AX = Y$, en factorisant la matrice du système en un produit de deux matrices triangulaires L et U .

$$L = \begin{bmatrix} 1 & 0. & . & . & 0 \\ \alpha_2 & 1 & . & . & . \\ 0. & . & . & . & . \\ . & . & . & . & 0 \\ 0. & . & 0 & \alpha_n & 1 \end{bmatrix} \cdot U = \begin{bmatrix} \beta_1 & c_1 & . & . & 0 \\ 0 & \beta_2 & . & . & . \\ . & . & . & . & . \\ . & . & . & . & c_{n-1} \\ 0. & . & . & 0 & \beta_n \end{bmatrix}$$

Exercice 13

Soit à résoudre l'équation de Laplace

$$\begin{cases} \Delta u = 0 \text{ dans le domaine } (x, y) \in [0, 20] \times [0, 10] \\ u(x, 0) = u(x, 10) = u(0, y) = 0 \text{ et } u(20, y) = 100 \\ h \in \{5, 2.5, 1.25, 0.625, 0.3125\} \end{cases}$$

Tout en variant h , résoudre cette EDP

1. En utilisant la méthode directe

2. En utilisant la méthode de relaxation de Liebmann.
3. Conclure.

```

clc;clear
Ceci est un commentaire; h=2.5 ou h=0.625
h=0.3125
a=0;
b=20;
c=0; d=10;
nx=(b-a)/h;
ny=(d-c)/h;
n=(nx-1)*(ny-1);
Ceci est un commentaire; (remplissage des éléments de la matrice A)
A=zeros(n);
for i=1:(n-1)
    A(i,i)=-4;
    A(i+1,i)=1;
    A(i,i+1)=1;
    if (mod(i,(nx-1))==0)
        A(i+1,i)=0;
        A(i,i+1)=0;
    end
end
for i=1:n-nx+1
    A(nx-1+i,i)=1;
    A(i,nx-1+i)=1;
end
A(n,n)=-4;
Ceci est un commentaire; (remplissage des éléments de la matrice B)
for i=1:n
    B(i)=0;
    if (mod(i,nx-1)==0)
        B(i)=-100;
    end
end
Ceci est un commentaire; (résolution du système  $Av=B$  et transformation du vecteur
v en la matrice U).
V = A';
k=1;

```

```
for j=1:ny-1
for i=1:nx-1
u(j,i)=V(k);
k=k+1;
end
end
Ceci est un commentaire; (décallage des éléments pour insérer les conditions aux
limites)
for j=ny:-1:2
for i=nx:-1:2
u(j,i)=u((j-1),i-1);
end
end
for i=1:nx
for j=1:ny
u(1,i)=0;
u(j,1)=0;
u(ny+1,i)=0;
u(j,nx+1)=100;
end
end
u(1,nx+1)=0;
x=0:h:b; y=0:h:d;
Ceci est un commentaire; (affichage la courbe en tenant compte des vecteurs x ,y et de
la matrice U)
mesh(x,y,u)
```

Chapter 3

Théorème d'Equivalence de Lax

Ce théorème est considéré comme le «théorème fondamentale de l'analyse numérique.» Il nous donne des conditions nécessaires et suffisantes pour la convergence de la solution discrète vers la solution de l'équation aux dérivées partielles.

3.1 Consistance

Soit L_h un schéma d'approximation pour un opérateur différentiel L : On définit l'erreur de troncature (ou de consistance) par

$$e_h(u) = L_h(u) - L(u)$$

L'approximation de L par L_h est consistante si l'erreur de troncature tend vers zéro lorsque les pas de discrétisation tendent vers zéro. La consistance ne concerne que le schéma de résolution, pas la solution elle-même. Un schéma consistant peut donner une mauvaise solution.

Définitions 1 Les opérateurs L_h sont des approximations consistantes de L si

$$\|L_h(u) - L(u)\| \rightarrow 0 \text{ quand } h \rightarrow 0$$

3.1.1 Exemple

D'abord on considère l'équation de la chaleur

$$\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} = 0$$

et d'après le developpement en série de Taylor:

$$\frac{\partial u}{\partial t} = \frac{u_i^{n+1} - u_i^n}{\Delta t} - \frac{\Delta t}{2} \left(\frac{\partial^2 u}{\partial t^2} \right) - \theta(\Delta t^2)$$

et

$$\frac{\partial^2 u}{\partial x^2} = \frac{u_{i-1}^{n+1} - 2u_i^n + u_{i+1}^n}{\Delta x^2} - \frac{\Delta x^2}{12} \left(\frac{\partial^4 u}{\partial x^4} \right) - \theta(\Delta x^4)$$

on peut l'écrire comme :

$$\begin{aligned} L(u) &= \frac{u_i^{n+1} - u_i^n}{\Delta t} \\ &\quad - \frac{u_{i-1}^{n+1} - 2u_i^n + u_{i+1}^n}{\Delta x^2} \\ &\quad - \frac{\Delta t}{2} \left(\frac{\partial^2 u}{\partial t^2} \right) + \frac{\Delta x^2}{12} \left(\frac{\partial^4 u}{\partial x^4} \right) \\ &\quad - \theta(\Delta t^2) - \theta(\Delta x^4) \\ &= 0 \end{aligned} \tag{3.1}$$

et par conséquent l'erreur de troncature

$$e_h(u) = L_h(u) - L(u) = \frac{\Delta t}{2} \left(\frac{\partial^2 u}{\partial t^2} \right) - \frac{\Delta x^2}{12} \left(\frac{\partial^4 u}{\partial x^4} \right) + \theta(\Delta t^2) - \theta(\Delta x^4)$$

Lorsque Δt et Δx tendent vers zéro, l'erreur de troncature $e_h(u)$ tend vers zéro; le schéma est donc consistant.

Exercice 9

Démontrer que le schéma de Dufort-Frankel défini par:

$$\frac{u_i^{n+1} - u_i^{n-1}}{2\Delta t} = \frac{u_{i+1}^n - u_i^{n+1} - u_i^{n-1} + u_{i-1}^n}{\Delta x^2}$$

n'est pas consistant.

Solution

Le calcul de l'erreur de troncature se fait en utilisant les développements de Taylor

$$u_i^{n+1} = u_i^n + \Delta t \frac{\delta u}{\delta t} + \frac{\Delta t^2}{2} \frac{\delta^2 u}{\delta t^2} + \frac{\Delta t^3}{6} \frac{\delta^3 u}{\delta t^3} + \frac{\Delta t^4}{24} \frac{\delta^4 u}{\delta t^4} + \dots$$

$$u_i^{n-1} = u_i^n - \Delta t \frac{\delta u}{\delta t} + \frac{\Delta t^2}{2} \frac{\delta^2 u}{\delta t^2} - \frac{\Delta t^3}{6} \frac{\delta^3 u}{\delta t^3} + \frac{\Delta t^4}{24} \frac{\delta^4 u}{\delta t^4} + \dots$$

d'où

$$u_i^{n-1} + u_i^{n+1} = 2u_i^n + \Delta t^2 \frac{\delta^2 u}{\delta t^2} + \frac{\Delta t^4}{12} \frac{\delta^4 u}{\delta t^4} + \dots$$

et

$$u_i^{n+1} - u_i^{n-1} = 2\Delta t \frac{\delta u}{\delta t} + \frac{\Delta t^3}{3} \frac{\delta^3 u}{\delta t^3} + \dots$$

également on peut trouver

$$u_i^{n+1} + u_{i+1}^n = 2u_i^n + \Delta x^2 \frac{\delta^2 u}{\delta x^2} + \frac{\Delta x^4}{12} \frac{\delta^4 u}{\delta x^4} + \dots$$

En remplaçant ces dernières équations dans le schéma on a:

$$\frac{\delta u}{\delta t} + \frac{\delta t^2}{6} \frac{\delta^3 u}{\delta t^3} + \dots = \frac{2}{\delta x^2} u_i^n + \frac{\delta^2 u}{\delta x^2} + \frac{\delta x^2}{12} \frac{\delta^4 u}{\delta x^4} - \frac{2u_i^n}{\delta x^2} - \frac{\delta t^2}{\delta x^2} \frac{\delta^2 u}{\delta t^2} - \frac{\delta t^4}{12\delta x^2} \frac{\delta^4 u}{\delta t^4} \dots$$

$$\frac{\delta u}{\delta t} = \frac{\delta^2 u}{\delta x^2} - \frac{\delta t^2}{6} \frac{\delta^3 u}{\delta t^3} + \frac{\delta x^2}{12} \frac{\delta^4 u}{\delta x^4} - \left(\frac{\delta t}{\delta x}\right)^2 \frac{\delta^2 u}{\delta t^2} - \frac{1}{12} \left(\frac{\delta t}{\delta x}\right)^2 \delta t^2 \frac{\delta^4 u}{\delta t^4} + \dots$$

On voit que si δt et δx ne tendent pas vers zéro de la même façon, l'erreur de troncature ne tend pas forcément vers zéro.

3.2 Stabilité

La stabilité d'un schéma traduit sa capacité à ne pas amplifier les erreurs lors du calcul numérique de la solution. Si un schéma s'écrit:

$$u^{n+1} = Au^n$$

avec n définissant l'indice temporelle ou l'itération, alors le schéma est stable s'il existe une constante K telle

$$\|A^n\| \leq K$$

Le schéma peut être globalement stable si K est bornée quelque soit h et k ; ou conditionnellement stable si K est bornée pour certaines valeurs de h et k : Dans ce dernier cas, on parlera d'une condition de stabilité qui relie h et k :

Il existe différentes manières d'effectuer une analyse de stabilité; analyse basée sur **la formulation matricielle** des schémas de calcul, la méthode spectrale de Von Neumann et la méthode de l'équation modifiée de Hirt-Shokin.

3.2.1 Stabilité Matricielle

Exercice 10

Etudier la stabilité de l'équation du transport, pour les schémas FTCS et d'Euler décentré aval .

L'équation du transport (2.7) pour $c = 1$, $\frac{\partial u}{\partial t} \approx \frac{u_i^{n+1} - u_i^n}{\Delta t}$ et $\frac{\partial u}{\partial x} \approx \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x}$ s'écrit :

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} = 0$$

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{2\Delta x} u_{i-1}^n - \frac{\Delta t}{2\Delta x} u_{i+1}^n$$

essayons de majorer u_i^{n+1} , soit $\alpha_n = \sup |u_i^n|$, on a:

$$|u_i^{n+1}| \leq |u_i^n| + \frac{\Delta t}{2\Delta x} |u_{i-1}^n| + \frac{\Delta t}{2\Delta x} |u_{i+1}^n|$$

$$\alpha_{n+1} \leq (1 + \frac{\Delta t}{\Delta x}) \alpha_n$$

cela donne

$$\alpha_n \leq (1 + \frac{\Delta t}{\Delta x})^n \alpha_0$$

pour $\Delta t = \Delta x$

$$\alpha_n \leq 2^n \alpha_0$$

pour $t=1$, $\alpha_{\frac{1}{k}} \leq 2^{\frac{1}{k}} \alpha_0$, qui est très grand quand k est très petit et c'est l'origine des instabilités.

Essayons un autre schéma

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + \frac{u_i^n - u_{i-1}^n}{\Delta x} = 0$$

qui s'écrit

$$u_i^{n+1} = (1 - \frac{\Delta t}{\Delta x}) u_i^n + (\frac{\Delta t}{\Delta x}) u_{i-1}^n$$

on alors

$$\left| u_i^{n+1} \right| \leq (1 - \frac{\Delta t}{\Delta x}) \left| u_i^n \right| + (\frac{\Delta t}{\Delta x}) \left| u_{i-1}^n \right|$$

ce qui donne

$$\alpha_{n+1} \leq \alpha_n$$

si $1 - \frac{\Delta t}{\Delta x} > 0$, c-à-d si $\Delta t < \Delta x$ Pour le cas périodique de période égale à 1, $h=0.01$, $k=0.005$, le programme suivant résout et analyse la stabilité de l'équation du transport avec donnée initiale $u(0, x) = \sin(2\pi x)$. (la solution exacte est $u(t, x) = \sin(2\pi(x - t))$)

```

**Schéma décentré
**Parametre
h=0.001;
k=0.5;
N=1/h;
***initialisation
**U="";
U(2:N+1) = sin(2*pi*h*(2:N+1));
U(1)=U(N+1);
U(N+2)=U(2);
**Boucle
for t=0:k:1,
U(2:N+1)=U(2:N+1)*(1-(k/(h)))+(k/(2*h))*U(1:N);
U(1)=U(N+1);
U(N+2)=U(2);
t
plot([1:N+2]*h ,U)
pause
end

```

Etude de la stabilité de l'équation de la chaleur

Résultats préliminaires:

1. Les s ($s = 1, \dots, N$) valeurs propres d'une matrice tridiagonale d'ordre N , dont les coefficients sont constants sur chaque diagonale (matrice de Toeplitz)

$$T_D = \begin{bmatrix} b & c & . & . & . \\ a & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & c \\ . & . & . & a & b \end{bmatrix}$$

sont données par

$$\lambda_s = b + 2\sqrt{ac}\cos\left(\frac{s\pi}{N+1}\right)$$

2. Une valeur propre de $[P(T_D)]^{-1}$ est donnée par $\frac{1}{P(\lambda)}$ avec $P(T_D) = \alpha_0 I + \alpha_1 T_D + \alpha_2 T_D^2 + \dots + \alpha_n T_D^n$

Exercice 11

Démontrer que le schéma explicite $u^{n+1} = (I + rT_D)u^n$ de l'équation de la chaleur est stable est stable pour $0 \leq r \leq 1/2$, avec

$$T_D = \begin{bmatrix} -2 & 1 & . & . & . \\ 1 & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & 1 \\ . & . & . & 1 & -2 \end{bmatrix}$$

La question qui se pose est le calcul des valeurs propres de la matrice A.

Commençons tout d'abord par celles de T_D . Par analogie(résultats préliminaires), les valeurs propres de T_D sont $\mu_s = -2 + 2\cos(\frac{s\pi}{N+1}) = 2[\cos(\theta) - 1]$ où $\theta = \frac{s\pi}{N+1}$. Puisque $\sin^2(\frac{\theta}{2}) = \frac{1}{2}(1 - \cos\theta)$, l'expression pour les valeurs propres de la matrice tridiagonale T_D devient:

$$\mu_s = -4\sin^2(\frac{\theta}{2})$$

Etant donné que la matrice d'itération $A = I + rT_D$, correspond à la forme polynomiale $P(A) = \alpha_0 I + \alpha_1 T_D$ avec $\alpha_0 = 1$ et $\alpha_1 = r$, les valeurs propres de A sont:

$$\lambda_s = 1 - 4r\sin^2(\frac{s\pi}{2M}) \quad s = 1, \dots, M-1$$

Par conséquent, la méthode est stable si:

$$\rho(A) = \max_s \left| 1 - 4r\sin^2(\frac{s\pi}{2M}) \right| \leq 1$$

donc (lorsque $M \rightarrow \infty \quad \sin^2\theta \rightarrow 1$)

$$0 \leq r \leq \frac{1}{2}$$

Ce schéma est conditionnellement stable.

Exercice 12

Démontrer que le schéma implicite $u^{n+1} = (I - rT_D)^{-1}u^n$ de l'équation de la chaleur est stable pour $-1 \leq \frac{1}{1+4r} \leq 1$

Exercice 13

Démontrer que le schéma de Crank-Nicolson $(2I - rT_D)u^{n+1} = (2I + rT_D)u^n$ de l'équation de la chaleur est stable pour toutes les valeurs de r

Exercice 14 de révision

On considère le système suivant

$$(3.2) \quad \begin{cases} \partial_t u + \partial_x(u^2) = 0, x \in \mathbb{R}, t \succ 0 \\ u(x, 0) = u_0(x) \quad x \in \mathbb{R} \end{cases}$$

avec u_0 une fonction bornée de classe C^1 sur \mathbb{R} . On admet que (3.2) admet une unique solution notée $u \in C^1(\mathbb{R} \times \mathbb{R}_+)$.

On discrétise le problème (3.2) avec un schéma en différences finies de type:

$$(3.3) \quad \begin{cases} \frac{u_i^{n+1} - u_i^n}{\Delta t} + \frac{1}{\Delta x} \left(g(u_i^n, u_{i+1}^n) - g(u_{i-1}^n, u_i^n) \right) = 0 & i \in \mathbb{Z}, n \in \mathbb{N} \\ u_i^0 = u(i\Delta x) & i \in \mathbb{Z} \end{cases}$$

où $\Delta t \succ 0$ et $\Delta x \succ 0$ sont des constantes, et $g : \mathbb{R}^2 \rightarrow \mathbb{R}$ une fonction vérifiant:

- i) g croissante par rapport à sa première variable.
- ii) g décroissante par rapport à sa deuxième variable.
- iii) g est k -Lipschitzienne sur $[m, M]$, avec $m = \inf_{x \in \mathbb{R}} u_0(x)$ et $M = \sup_{x \in \mathbb{R}} u_0(x)$.

On se propose de montrer par récurrence que le schéma (3.3) est stable en norme L^∞ et que $u_i^n \in [m, M]$ pour tout $i \in \mathbb{Z}$ et $n \in \mathbb{N}$

1. Montrer que le schéma (3.3) peut s'écrire sous la forme

$$u_i^{n+1} = u_i^n + \alpha_i^n (u_{i-1}^n - u_i^n) + \beta_i^n (u_{i+1}^n - u_i^n).$$

avec $\alpha_i^n \geq 0$ et $\beta_i^n \geq 0 \forall i \in \mathbb{Z}$ et $n \in \mathbb{N}$.

2. soit $n \in \mathbb{N}$ fixé, et supposons que $u_i^n \in [m, M]$ pour tout $i \in \mathbb{Z}$. Montrer que sous la condition $k \frac{\Delta t}{\Delta x} \leq \frac{1}{2}$, on a $\alpha_i^n \leq \frac{1}{2}$ et $\beta_i^n \leq \frac{1}{2} \forall i \in \mathbb{Z}$.
3. En déduire que $u_i^{n+1} \in [m, M]$ pour tout $i \in \mathbb{Z}$, puis conclure.
4. Application : Que devient le schéma (3.3) pour $g(u, v) = 2u^2 - v^2$? Montrer que le schéma ainsi obtenu est consistant d'ordre au moins un en temps et en espace.

Corrigé Exercice 14

1. On ajoutant et retranchant $g(u_i^n, u_i^n)$, le système (3.3) s'écrit:

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{\Delta x} \left(-g(u_i^n, u_{i+1}^n) + g(u_i^n, u_i^n) - g(u_i^n, u_i^n) + g(u_{i-1}^n, u_i^n) \right)$$

ou de façon équivalente

$$u_i^{n+1} = u_i^n + \alpha_i^n (u_{i-1}^n - u_i^n) + \beta_i^n (u_{i+1}^n - u_i^n)$$

avec

$$\alpha_i^n = \begin{cases} \lambda \frac{g(u_{i-1}^n, u_i^n) - g(u_i^n, u_i^n)}{u_{i-1}^n - u_i^n} & \text{si } u_{i-1}^n \neq u_i^n \\ 0 & \text{si } u_{i-1}^n = u_i^n \end{cases}$$

$$\beta_i^n = \begin{cases} \lambda \frac{g(u_i^n, u_{i+1}^n) - g(u_i^n, u_i^n)}{u_{i+1}^n - u_i^n} & \text{si } u_{i+1}^n \neq u_i^n \\ 0 & \text{si } u_{i+1}^n = u_i^n \end{cases}$$

avec $\frac{\Delta t}{\Delta x} \succ 0$. Comme g est croissante par rapport à sa première variable, alors $\alpha_i^n \geq 0$ pour tout $i \in \mathbb{Z}$ et $n \in \mathbb{N}$. En effet, si $u_{i-1}^n \succ u_i^n$ alors $g(u_{i-1}^n, u_i^n) \geq g(u_i^n, u_i^n)$ et $\alpha_i^n = \lambda \frac{g(u_{i-1}^n, u_i^n) - g(u_i^n, u_i^n)}{u_{i-1}^n - u_i^n} \geq 0$. De même, si $u_{i-1}^n \prec u_i^n$ alors $g(u_{i-1}^n, u_i^n) \leq g(u_i^n, u_i^n)$ et $\alpha_i^n = \lambda \frac{g(u_{i-1}^n, u_i^n) - g(u_i^n, u_i^n)}{u_{i-1}^n - u_i^n} \geq 0$, et finalement si $u_{i-1}^n = u_i^n$ alors $\alpha_i^n = 0$. On vérifie de la même façon que $\beta_i^n \geq 0$ pour tout $i \in \mathbb{Z}$ et $n \in \mathbb{N}$.

2. Comme g est k -Lipschitzienne alors $\forall v, v', w$ et $w' \in [m, M]$ on a :
 $|g(v, w) - g(v', w)| \leq k |v - v'|$ et $|g(v, w) - g(v, w')| \leq k |w - w'|$.
 Il vient que si $u_i^n \in [m, M]$ pour tout $i \in \mathbb{Z}$ alors :

$$\alpha_i^n = \lambda \frac{g(u_{i-1}^n, u_i^n) - g(u_i^n, u_i^n)}{u_{i-1}^n - u_i^n} \leq k\lambda$$

et

$$\beta_i^n = \lambda \frac{g(u_i^n, u_i^n) - g(u_i^n, u_{i+1}^n)}{u_{i+1}^n - u_i^n} \leq k\lambda$$

Ainsi, si $k\lambda \leq 1/2$ alors $\alpha_i^n \leq 1/2$ et $\beta_i^n \leq 1/2$.

3. On a $u_{i+1}^n = (1 - \alpha_i^n - \beta_i^n)u_i^n + \alpha_i^n u_{i-1}^n + \beta_i^n u_{i+1}^n$. Comme $0 \leq \alpha_i^n \leq 1/2$ et $0 \leq \beta_i^n \leq 1/2$ alors $0 \leq \alpha_i^n - \beta_i^n \leq 1$. Ainsi u_{i+1}^n est une combinaison convexe de $u_i^n, u_{i+1}^n, u_{i-1}^n$ donc $u_{i+1}^n \in [m, M] \forall i \in \mathbb{Z}$ et $\forall i \in \mathbb{N}$.

4. pour $g(u, v) = 2u^2 - v^2$ on a:

$$g(u_i^n, u_{i+1}^n) = 2(u_i^n)^2 - (u_{i+1}^n)^2 \text{ et } g(u_{i-1}^n, u_i^n) = 2(u_{i-1}^n)^2 - (u_i^n)^2$$

donc le schéma (3.3) s'écrit:

$$\begin{cases} \frac{u_{i+1}^n - u_i^n}{\Delta t} + \frac{1}{\Delta x} \left(- (u_{i+1}^n)^2 + 3(u_i^n)^2 - 2(u_{i-1}^n)^2 \right) = 0 & i \in \mathbb{Z}, n \in \mathbb{N} \\ u_i^0 = u_0(i\Delta x) & i \in \mathbb{Z} \end{cases}$$

Notons $\bar{u}_i^n = u(i\Delta x, n\Delta t)$. L'erreur de consistance du schéma (3.3) s'écrit:

$$\epsilon_i^n = \frac{\bar{u}_i^{n+1} - \bar{u}_i^n}{\Delta t} + \frac{1}{\Delta x} \left(- (\bar{u}_{i+1}^n)^2 + 3(\bar{u}_i^n)^2 - 2(\bar{u}_{i-1}^n)^2 \right)$$

Un developpement de Taylor au voisinage de $(i\Delta x, n\Delta t)$ donne :

$$\frac{\bar{u}_i^{n+1} - \bar{u}_i^n}{\Delta t} = \partial_t \bar{u}_i^n + O(\Delta t)$$

De même, en notant $\bar{v}_i^n = (\bar{u}_i^n)^2$, on a:

$$\bar{v}_i^n = \bar{v}_i^n + \Delta x \partial_x \bar{v}_i^n + O(\Delta x^2)$$

$$\bar{v}_{i-1}^n = \bar{v}_i^n - \Delta x \partial_x \bar{v}_i^n + O(\Delta x^2)$$

donc

$$\frac{\bar{v}_{i-1}^n + 3\bar{v}_i^n - 2\bar{v}_{i+1}^n}{\Delta x} = \partial_x \bar{v}_i^n + O(\Delta x)$$

c'est-à-dire

$$\epsilon_i^n = \underbrace{\partial_t + \partial_x (\bar{u}_i^n)^2}_{=0 \text{ d'après (3.2)}} + O(\Delta t + \Delta x)$$

Ainsi, le schéma (3.3) est consistant d'ordre au moins un en temps et en espace.

Chapter 4

Equations aux dérivées partielles dans MATLAB

Les équations aux dérivées partielles (EDP) à une variable spatiale avec condition initiale possède un 'Solveur' dans Matlab pdepe.

Par exemple, suppose que nous aimerions résoudre l'équation.

$$\begin{aligned} \pi^2 \frac{\partial u}{\partial t} &= \frac{\partial u^2}{\partial x^2} \\ \text{pour } 0 \leq x \leq 1, \quad t \geq 1 \\ \text{et } u(x, 0) &= \sin \pi x \\ \text{sachant que } u(0, t) &= 0 \\ \text{et } \pi e^{-t} + \frac{\partial u}{\partial x}(1, t) &= 0 \end{aligned} \tag{4.1}$$

MATLAB spécifie tel EDP parabolique dans la forme:

$$\begin{aligned} c(x, t, u, \frac{\partial u}{\partial x}) \frac{\partial u}{\partial t} &= x^{-m} \frac{\partial}{\partial x} (x^m f(x, t, u, \frac{\partial u}{\partial x})) + s(x, t, u, \frac{\partial u}{\partial x}) \\ \text{pour } a \leq x \leq b, \quad t_0 \leq t \leq t_f \end{aligned} \tag{4.2}$$

avec les conditions aux limites

$$p(x, t, u) + q(x, t).f(x, t, u, \frac{\partial u}{\partial x}) = 0$$

Pour tout t et $x = a$ ou $x = b$.

La solution satisfait la condition initiale

$$u(x, t_0) = u_0(x)$$

(Remarquons que la fonction f apparait dans l'équation et dans les conditions aux limites). Typiquement, pour des raisons de clarté, chaque fonction devra être définie séparément dans un fichier M-file. Donc, les fonctions c , f et s associées à la fonction devront être écrit dans le même fichier (M-file), les fonctions p , q associées aux conditions aux limites dans un deuxième fichier(M-file) (en plus, se rappeler que f est la même et seulement a besoin d'être spécifiée une autre fois), finalement la fonction initiale $f(x)$ dans un troisième fichier. La commande *pdepe* va combiner ces M-fichiers et nous donne la solution du problème.

Dans notre exemple , nous avons:

$$\begin{aligned} c(x, t, u, u_x) &= \pi^2 \\ f(x, t, u, u_x) &= \frac{\partial u}{\partial x} \\ s(x, t, u, u_x) &= 0 \end{aligned}$$

Qui seront définies par la fonction M-file *eqn1.m*. (La spécification $m = 0$ sera précisée plus tard.)

```
function [c,f,s] = eqn1 ( x,t ,u,DuDx)
**   EQN1 : MATLAB function M-file that specifies
**   a PDE in time and one space dimension.
c = pi*pi ;
f = DuDx;
s = 0 ;
```

Pour les conditions aux limites, nous avons

$$\begin{aligned} p(0, t, u) &= u; \quad q(0, t) = 0 \\ p(1, t, u) &= \pi e^{-t}; \quad q(1, t) = 1 \end{aligned}$$

Qui seront définies par la fonction M-file bc1.m

```
function[pl,ql,pr ,qr ] = bc1( xl,ul,xr ,ur ,t )
**   BC1 : MATLAB function M- file that specifies boundary conditions
**   for a PDE in time and one space dimension.
pl = ul;
ql = 0 ;
pr = pi*exp(-t) ;
qr = 1 ;
```

Pour la condition initiale, nous avons;

$$u_0(x) = u(x, 0) = \sin \pi x$$

qui sera définie par la fonction M-file initial1.m

```
function value = initial1 ( x)
**   INITIAL 1 : MATLAB function M-file that specifies the initial condition
**   for a PDE in time and one space dimension.
value = sin (pi* x) ;
```

Finalement, nous sommes en mesure de résoudre le PDE avec *pdepe*. Le script suivant M-file résout le PDE et dessine la surface de la solution (voir Figure 5.2). Si nous choisissons une grille des x et des valeurs pour t .

```
**   PDE1 : MATLAB script M-file that solves and plots solutions to the PDE stored
in eqn1.m
m = 0 ;
**   NOTE: m= 0 specifies no symmetry in the problem. Taking
**   m= 1 specifies cylindrical symmetry, while m= 2 specifies
**   spherical symmetry.
**   Define the solution mesh
x = linspace( 0 ,1 ,20 ) ;
```

```

t = linspace( 0 ,2 ,10 ) ;
**  Solve the PDE
u = pdepe( m,@eqn1 ,@initial1 ,@bc1 ,x,t ) ;
**  Plot solution
surf( x,t ,u ) ;
title( Surface plot of solution. ) ;
xlabel( Distance x ) ;
ylabel( Time t ) ;

```

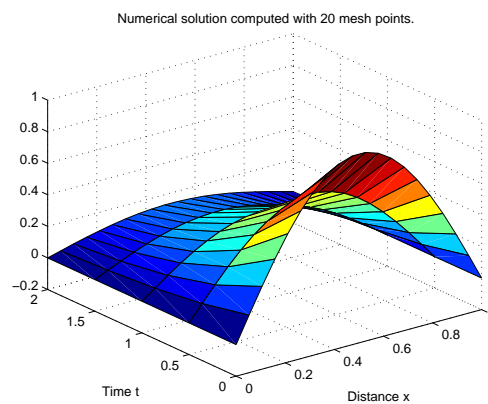


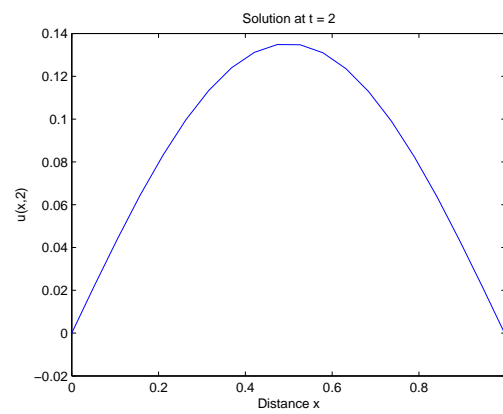
Figure 4.1: Mesh plot for solution to Equation 5.1

Souvent, il sera utile de voir l'allure de la solution pour un t fixé (u est tracé avec x). La solution $u(t, x)$ est enregistré comme étant une matrice indexée par les indices du vecteur de t et x . Par exemple, $u(1, 5)$ représente la valeur de u au point $(t(1), x(5))$. Nous pouvons tracer u initialement (à $t = 2$) avec la commande `plot(x, u(end, :))` (voir figure ci-dessous).

```

figure
plot(x,u(end,:))
title('Solution at t = 2')
xlabel('Distance x')
ylabel('u(x,2)')

```


Figure 4.2: Solution at $t = 2$

Chapter 5

LA METHODE DES ELEMENTS FINIS

Comparaison Différences finies Eléments finis Le schéma ci-dessous compare les 2 méthodes en termes de démarche. La méthode des différences finies cherche à résoudre exactement une équation approchée, alors que la méthode des éléments finis cherche une solution approchée à une équation exacte.

Différences finies : Equation physique continue aux Dérivées Partielles

1. reformulation discrète approchée de l'équation d'origine.
2. résolution exacte l'équation approchée (Méthodes d'approximation (Différences finies)).

Eléments finis : Equation physique continue aux Dérivées Partielles

1. reformulation intégrale exacte de l'équation d'origine.
2. résolution approchée de l'équation exacte (Méthodes d'approximation de solution (Eléments finis))

Par exemple, suppose que nous aimerions résoudre l'équation.

$$\begin{aligned}
& \pi^2 \frac{\partial u}{\partial t} = \frac{\partial u^2}{\partial x^2} \\
& \text{pour } 0 \leq x \leq 1, \quad t \geq 1 \\
& \text{et } u(x, 0) = \sin \pi x \\
& \text{sachant que } u(0, t) = 0 \\
& \text{et } \pi e^{-t} + \frac{\partial u}{\partial x}(1, t) = 0
\end{aligned} \tag{5.1}$$

MATLAB spécifie tel EDP parabolique dans la forme:

$$\begin{aligned}
c(x, t, u, \frac{\partial u}{\partial x}) \frac{\partial u}{\partial t} &= x^{-m} \frac{\partial}{\partial x} (x^m f(x, t, u, \frac{\partial u}{\partial x})) + s(x, t, u, \frac{\partial u}{\partial x}) \\
&\text{pour } a \leq x \leq b, \quad t_0 \leq t \leq t_f
\end{aligned} \tag{5.2}$$

avec les conditions aux limites

$$p(x, t, u) + q(x, t) \cdot f(x, t, u, \frac{\partial u}{\partial x}) = 0$$

Pour tout t et $x = a$ ou $x = b$.

La solution satisfait la condition initiale

$$u(x, t_0) = u_0(x)$$

(Remarquons que la fonction f apparait dans l'équation et dans les conditions aux limites). Typiquement, pour des raisons de clareté, chaque fonction devra être définie séparément dans un fichier M-file. Donc, les fonctions c , f et s associées à la fonction devront être écrit dans le même fichier (M-file), les fonctions p , q associées aux conditions aux limites dans un deuxième fichier(M-file) (en plus, se rappeler que f est la même et seulement a besoin d'être spécifiée une autre fois), finalement la fonction initiale $f(x)$ dans un troisième fichier. La commande *pdepe* va combiner ces M-fichiers et nous donne la solution du problème.

Dans notre exemple , nous avons:

$$\begin{aligned}c(x, t, u, u_x) &= \pi^2 \\f(x, t, u, u_x) &= \frac{\partial u}{\partial x} \\s(x, t, u, u_x) &= 0\end{aligned}$$

Qui seront définies par la fonction M-file *eqn1.m*. (La spécification $m = 0$ sera précisée plus tard.)

```
function [c,f,s] = eqn1 ( x,t ,u,DuDx)
**   EQN1 : MATLAB function M-file that specifies
**   a PDE in time and one space dimension.
c = pi*pi ;
f = DuDx;
s = 0 ;
```

Pour les conditions aux limites, nous avons

$$\begin{aligned}p(0, t, u) &= u; \quad q(0, t) = 0 \\p(1, t, u) &= \pi e^{-t}; \quad q(1, t) = 1\end{aligned}$$

Qui seront définies par la fonction M-file *bc1.m*

```
function[pl,ql,pr ,qr ] = bc1( xl,ul,xr ,ur ,t )
**   BC1 : MATLAB function M- file that specifies boundary conditions
**   for a PDE in time and one space dimension.
pl = ul;
ql = 0 ;
pr = pi*exp(-t) ;
qr = 1 ;
```

Pour la condition initiale, nous avons;

$$u_0(x) = u(x, 0) = \sin \pi x$$

qui sera définie par la fonction M-file *initial1.m*