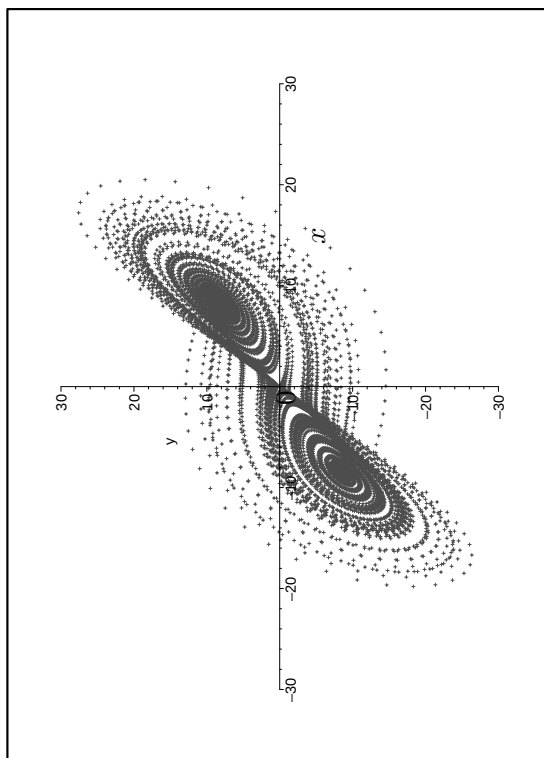


E.I.S.T.I. Département Mathématiques

**EQUATIONS AUX DÉRIVÉES PARTIELLES.
RÉSOLUTION NUMÉRIQUE DES EDP.
DIFFÉRENCES FINIES.**

Irina Kortchemski



Contents

1	Modélisation mathématique et les équations différentielles aux dérivées partielles	5
1.1	Equations non stationnaires	5
1.1.1	Equation de la chaleur	6
1.1.2	Equation de convection-diffusion	6
1.1.3	Equation d'advection	6
1.1.4	Equation d'ondes	6
1.1.5	Equations de Maxwell	7
1.1.6	Equation de Schrodinger	7
1.1.7	Equation de Klein-Gordon	7
1.1.8	Equation de Korteweg de Vries	8
1.1.9	Equation de Sine-Gordon	8
1.1.10	Equation de Black et Scholes	8
1.1.11	Equations de Keller-Segel	8
1.1.12	Equations de Navier-Stokes	9
1.1.13	Equation de Burgers	9
1.1.14	Equations de Lagrange	9
1.2	Equations stationnaires	10
1.2.1	Equations d'Einstein	10
1.2.2	Equation de Poisson	10
1.2.3	Système de Lamé	10
1.2.4	Equation des plaques	10
1.2.5	Equation de Helmholtz	10
2	Équations différentielles ordinaires	11
2.1	Problème de Cauchy	11
2.2	Principes de la méthode aux Différences Finies	11
2.3	Formules de Taylor	12
2.4	Discretisation des dérivées premières	12
2.5	Discretisation des dérivées secondes	13
2.6	Consistance	13
2.6.1	Erreur de troncature locale	13
2.6.2	Consistance du schéma	13
2.6.3	Consistance du schéma d'Euler progressif ou explicite	13
2.6.4	Stabilité du schéma d'Euler progressif ou explicite	14
2.7	Schéma de Runge-Kutta d'ordre 2	15
2.7.1	Consistance du schéma de Runge-Kutta	15
2.8	Schéma de Runge-Kutta d'ordre 4	16
2.9	Schéma implicite d'Euler et schéma implicite de Crank-Nicolson	16
2.10	Consistance du schéma implicite de Crank-Nicolson	16
2.11	Convergence du schéma implicite de Crank-Nicolson	17
2.12	Applications	18
2.12.1	Compte Epargne : Calculette d'intérêts	18
2.12.2	Oscillations harmoniques	18
2.12.3	Oscillateur nonlinéaire dissipatif	19
2.12.4	Modèle de Lotka - Volterra	20
2.12.5	Mouvement des astres célestes	21

3	Principes de la méthode aux Différences Finies pour les EDP.	23
3.1	Formule de Taylor	23
3.2	Discrétisation des dérivées premières	23
3.2.1	Discrétisation des dérivées secondes	25
4	Stratégie de base dans les approches de discrétisation	26
4.1	Maillage	26
4.2	Méthodes explicites	27
4.2.1	Méthode d'Euler progressive	27
4.2.2	Implementation du schéma explicite d'Euler progressive	27
4.2.3	Forme matricielle de l'équation discrete	29
4.2.4	Visualisation des solutions	29
4.3	Méthodes implicites	30
4.3.1	Méthode de Crank-Nicolson	30
4.3.2	Forme matricielle de l'équation discrète et Algorithme de Thomas	32
5	Equations paraboliques. Résolution numérique de l'équation de la chaleur par Crank-Nicolson	34
5.1	Algorithme de Crank-Nicolson	34
5.2	Implémentation en MATLAB ou SCILAB.	36
5.3	Visualisation des solutions	38
5.4	Comparaisons des conditions aux limites	39
5.5	Méthode d'Euler rétrograde ou implicite.	41
6	Traitement des conditions aux limites de Neumann	42
6.1	Conditions aux limites de Neumann d'ordre 1	42
6.2	Implementation des conditions aux limites de Neumann d'ordre 1 en utilisant la méthode d'Euler explicite	42
6.3	Implementation des conditions aux limites de Neumann d'ordre 2 en utilisant la méthode d'Euler explicite	44
6.4	Implementation des conditions aux limites de Neumann en utilisant la méthode de Crank-Nicolson	45
6.5	Conditions aux limites périodiques	48
7	Schémas aux plusieurs niveaux temporels	50
7.1	Schéma de Leap-Frog	50
7.2	Schéma de Lax-Wendroff	51
8	Schémas de discrétisation. Résumé	52
9	Consistance	52
9.1	Consistance de la méthode d'Euler progressif	53
9.2	Consistance du schéma de DuFort-Frankel	54
10	Stabilité	54
10.1	Stabilité d'un point du vue pratique	56
10.2	Stabilité de méthode de Crank-Nicolson	57
11	Convergence	58
11.1	Théorème de Lax	58

12 Propagation des erreurs.	60
13 Analyse de Fourier des EDP. Relation de dispersion	62
14 Equations hyperboliques	64
14.1 Equation d'advection	64
14.1.1 Schéma upwind	64
14.1.2 Implémentation du schéma explicite upwind	64
14.1.3 Schéma explicite centré	67
14.1.4 Schéma de Lax-Friedrichs	67
14.1.5 Schéma de Lax-Wendroff	68
14.1.6 Schéma de Leap-Frog	69
14.1.7 Schéma de Crank-Nicolson	71
14.2 Schémas de l'équation d'onde d'ordre 2	74
14.3 Conditions aux limites numériques	78
15 Equations elliptiques	79
15.1 Equation de Poisson en coordonnées cartésiennes	79
15.2 Equation de Poisson en coordonnées polaires	83
16 Résolution numérique de l'équation de chaleur à 2 dimension	85
16.1 Méthode explicite	85
16.2 Méthode implicite	87
17 EDP nonlinéaires	88
17.1 Résolution numérique de l'équation de Burgers à une dimension.	88
17.1.1 Méthode de Mac-Cormack	89
17.1.2 Méthode de Crank-Nicolson.	89
17.2 Résolution numérique de l'équation de Burgers non visqueuse à une dimension.	90
17.2.1 Méthode de Lax.	90
17.2.2 Méthode de Lax-Wendroff.	91
17.2.3 Méthode de Mac-Cormack.	91
17.3 Résolution numérique de l'équation de Korteweg de Vries.	91
17.3.1 Discrétisation	91
17.3.2 Résolution	92

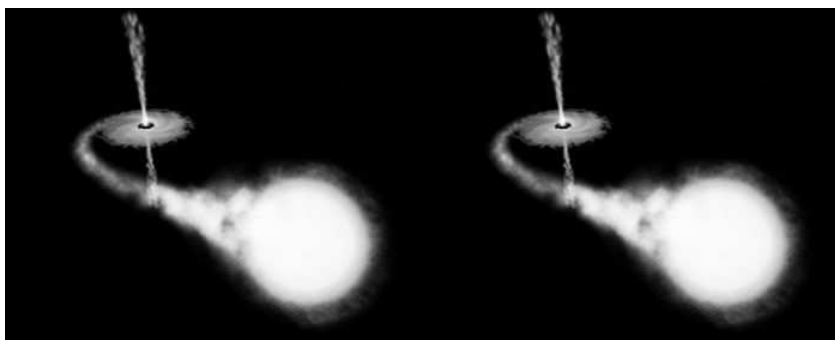
1 Modélisation mathématique et les équations différentielles aux dérivées partielles

En mathématiques une équation aux dérivées partielles ou (EDP) est une équation dont les solutions sont les fonctions inconnues vérifiant certaines conditions concernant leurs dérivées partielles.

Une EDP a souvent de très nombreuses solutions. Alors que les ensembles de solutions d'une équation différentielle ordinaire sont paramétrées par un ou plusieurs paramètres correspondant aux conditions aux limites, dans le cas des EDP les conditions aux limites se présentent plutôt sous la forme de fonction. Intuitivement cela signifie que l'ensemble des solutions est beaucoup plus grand, ce qui est vrai dans la quasi-totalité des problèmes.

Les EDP sont très répandues dans les sciences, puisqu'elles apparaissent aussi bien en dynamique des structures, mécanique des fluides que dans les théories de la gravitation, de l'électromagnétisme (équations de Maxwell) ou en finance. Elles sont primordiales dans des domaines tels que la simulation aéronautique, la synthèse d'images, ou la prévision météorologique. Enfin, les équations les plus importantes de la relativité générale et de la mécanique quantique sont également des EDP.

Comme un exemple considérons un système de deux trous noirs. Une binaire compacte spiralante est ce que deviendra le système de deux trous noirs à la fin de sa vie. A la fin de la phase spiralante, le système binaire atteint ce qu'on appelle la dernière orbite circulaire.



Sur cette orbite les calculs approximatifs doivent être remplacé par un calcul de résolution numériques des équations d'Einstein. Un tel calcul est indispensable pour décrire le mécanisme de fusion des deux trous noirs, et obtenir la forme d'onde gravitationnelle produite lors de cette phase. Ce problème difficile a été l'objet de ce qu'on a appelé le "binary black hole grande challenge", qui a mobilisé de nombreux instituts américains. En 2012 la relativité numérique a réussi le résoudre numériquement et simuler la fusion des deux trous noirs. En partie grâce à ces simulations les chercheurs de la collaboration LIGO-Virgo ont en effet détecté des ondes gravitationnelles.

L'un des dix problèmes à un million de dollars proposés par la fondation Clay consiste à montrer l'existence et la continuité par rapport aux données initiales d'un système d'EDP appelé équations de Navier Stokes. Ces équations servent énormément dans la mécanique des fluides.

1.1 Equations non stationnaires

Des équations non stationnaires correspondent à une évolution temporelle. La solution est donnée à partir d'une fonction (ou condition initiale) que l'on doit propager le long de l'axe du temps; en complément on doit tenir compte des conditions aux limites du problème.

1.1.1 Equation de la chaleur

$$\begin{cases} \frac{\partial u}{\partial t} - \beta \Delta u = f(x, y, z, t) & (x, y, z) \in \Omega \\ u(t = 0, x, y, z) = g(x, y, z) \\ u(t, x, y, z) = 0 & (x, y, z) \in \partial\Omega \end{cases} \quad (1)$$

Ici Δ est le Laplacien:

$$\Delta = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}$$

Ce problème est connue aussi sous le nom d'équation de diffusion et modélise la diffusion ou migration d'une concentration ou densité à travers des frontières $\partial\Omega$ du domaine Ω . On peut imaginer un polluant diffusant dans l'atmosphère, ou bien une espèce chimique migrant dans un substrat.

1.1.2 Equation de convection-diffusion

$$\begin{cases} \frac{\partial u}{\partial t} - V \cdot \nabla u - \beta \Delta u = f \\ u(t = 0, x) = g(x) \\ u(t, x) = 0 & x \in \partial\Omega \end{cases} \quad (2)$$

Il existe de nombreuses variantes de l'équation de la chaleur. Supposons que la chaleur se propage dans un milieu en mouvement comme, par exemple, un fluide animé d'une vitesse $V(x, t)$. On doit ajouter dans l'équation de chaleur un terme qui s'appelle le terme de convection. La solution de cette équation représente aussi la fonction de distribution dont les particules individuelles participent au mouvement Brownien (théorème Feynman-Kac).

1.1.3 Equation d'advection

$$\begin{cases} \frac{\partial u}{\partial t} - V \cdot \nabla u = 0 \\ u(t = 0, x) = g(x) \\ u(t, x) = 0 & x \in \partial\Omega \end{cases} \quad (3)$$

Cette équation décrit la propagation d'une fluctuation sans déformation. La fonction $g(x - Vt)$ est une solution de l'équation.

1.1.4 Equation d'ondes

L'équation des ondes modélise des phénomènes de propagation d'ondes ou de vibration.

$$\begin{cases} \frac{\partial^2 u}{\partial t^2} - \Delta u = f \\ u(t = 0, x) = g \\ u = 0 & x \in \partial\Omega \\ \frac{\partial u}{\partial t}(t = 0) = g_1(x) & x \in \partial\Omega \end{cases} \quad (4)$$

La nature des ondes ou des paquets d'onde ou des fluctuations qui se propagent peut être très différentes.

- A une fluctuation thermique qui se propage on associe PHONON.
- A une fluctuation de densité volumique de charge dans un plasma qui se propage on associe PLASMON.
- A une fluctuation d'aimantation qui se propage on associe MAGNON.
- A une déformation élastique qui se propage on associe POLARON.

- A un paquet d'une onde électromagnétique on associe PHOTON.
- A une fluctuation de polarisation d'une onde qui se propage on associe EXCITON.

En 2 dimensions l'équation modélise les vibrations d'une membrane élastique tendue . Sous l'action d'une force elle se déforme et son déplacement est noté u . On suppose qu'elle est fixée sur son bord, ce qui donne une condition aux limites de Dirichlet.

1.1.5 Equations de Maxwell

La propagation des ondes électromagnétiques est régie par les équations des Maxwell:

$$\begin{cases} \text{rot} \vec{E} = -\frac{\partial \vec{B}}{\partial t} \\ \text{rot} \vec{B} = \frac{1}{c^2} \frac{\partial \vec{E}}{\partial t} \end{cases} \quad (5)$$

Supposons que l'onde $\vec{E}(0, E_y, 0)$, $\vec{B}(B_x, 0, 0)$ se propage dans la direction OZ. Notons $B \equiv B_x, E \equiv E_y$. Soit $c = 1$.

Le système se transforme au système de deux équations couplées:

$$\begin{cases} \frac{\partial E}{\partial z} = \frac{\partial B}{\partial t} \\ \frac{\partial B}{\partial z} = -\frac{\partial E}{\partial t} \end{cases} \quad (6)$$

Pour résoudre le système de Maxwell on utilise un schéma aux différence finis dit Leap Frog (Saute Mouton).

1.1.6 Equation de Schrodinger

L'équation de Schrodinger décrit l'évolution de la fonction d'onde Ψ d'une particule soumise à un potentiel V . Le module au carré $|\Psi|^2$ s'interprète comme la densité de probabilité pour détecter que la particule se trouve au point (x, t) .

$$\begin{cases} i \frac{\partial \Psi}{\partial t} + \Delta \Psi - V \Psi = 0 \\ \Psi(t=0) = \Psi_0(x) \\ \Psi(0, t) = 0 \quad x \in \partial\Omega \end{cases} \quad (7)$$

Pour résoudre l'équation de Schrodinger on utilise le schéma aux différences finis dit Crank- Nicolson qui conserve la norme de fonction-d'onde.

1.1.7 Equation de Klein-Gordon

L'équation de Klein-Gordon est une version relativiste de l'équation de Schrodinger décrivant des particules massives de spin nul, sans ou avec charge électrique.

$$\begin{cases} \frac{\partial^2 \Psi}{\partial t^2} - \Delta \Psi - b\Psi = 0 \\ \Psi(t=0, x) = \Psi_0(x) \\ \Psi = 0 \quad x \in \partial\Omega \end{cases} \quad (8)$$

1.1.8 Equation de Korteweg de Vries

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + b \frac{\partial^3 u}{\partial x^3} = 0 \quad (9)$$

Une des solutions de cette équation nonlinéaire s'appelle soliton. C'est une onde solitaire qui se propage sans se déformer dans un milieu non-linéaire et dispersif.

Ce mode de propagation d'une vague sur de longues distances explique aussi la propagation des tsunامي. Ceux-ci se déplacent pratiquement sans effet notable en eaux profondes.

L'utilisation de solitons a été proposée pour améliorer la performance des transmissions dans les réseaux optiques de télécommunications.

En théorie (quantique) des champs, les solitons topologiques sont des solutions classiques non triviales topologiquement. Ils portent différents noms suivant qu'ils minimisent l'action (instanton) ou l'énergie, et en fonction des topologies respectives de l'espace et du groupe de jauge (monopôle, vortex, skyrmion, toron,...).

1.1.9 Equation de Sine-Gordon

$$\frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} - \sin u = 0 \quad (10)$$

Les solutions de cette équation nonlinéaire sont aussi les solitons: kink et antikink.

1.1.10 Equation de Black et Scholes

$$\begin{cases} \frac{\partial V}{\partial t} - rV + rS \frac{\partial V}{\partial S} + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} = 0 \\ V(t = T, S) = \max(S - K, 0) \\ S \in [0, L] \end{cases} \quad (11)$$

Cette équation permet de trouver le prix $V(S, t)$ de l'option d'achat (ou call) d'une action qui vaut initialement S et qu'on pourra acheter au prix K dans un temps ultérieur T . $V(0, S)$ est le prix au temps $t = 0$ de l'option d'achat de prix d'exercice K à l'échéance $T > 0$, et d'actif S en $t = 0$. On note σ la volatilité de l'action et r le taux d'intérêt. est un problème avec condition finale et non pas initiale, mais que le signe de dérivée seconde en espace est opposé à celui dans (10). Par conséquent, après inversion du temps (10) est bien une équation parabolique.

1.1.11 Equations de Keller-Segel

De nombreux exemples proviennent de la biologie qui constitue un vaste champ d'applications et une source de nombreuses questions des mathématiques nouvelles. L'exemple le plus classique recouvre des phénomènes appelés "chimiotactie". Cette terminologie recouvre un colonies bactériennes qui adopte des mouvements collectifs surprenants à partir de communications individuelles élémentaires. L'analyse mathématique permet de faire le lien entre ces comportement individuels supposés et les phénomènes collectifs observés (concentrations ponctuelles anormalement fortes, fronts périodiques où alternent fortes densité et zones désertées, zones d'invasion dont les frontières suivent des lois que l'on voudrait expliquer). Les équations de Keller-Segel décrivent ces phénomènes:

$$\begin{cases} \frac{\partial n}{\partial t} - \beta \Delta n - \operatorname{div}(n \chi \nabla \phi) = 0 \\ -\Delta \phi = n \end{cases} \quad (12)$$

1.1.12 Equations de Navier-Stokes

Les équations de Navier-Stokes restent un des grands problèmes mathématiques depuis qu'elles ont été écrites au début de 19^{ème} siècle pour modéliser la dynamique des fluides.

- **La forme la plus générale des équations de Navier-Stokes:**

$$\rho \left(\frac{\partial \vec{v}}{\partial t} + \frac{1}{2} \text{grad}(v^2) - [\vec{v} \times \text{rot} \vec{v}] \right) = \rho \vec{g} - \text{grad} p + \mu \Delta \vec{v} + \frac{1}{3} \mu \text{grad}(\text{div} \vec{v}) \quad (13)$$

- **Les équations d'Euler.**

$$\begin{cases} \frac{\partial \rho}{\partial t} = -(\rho \frac{\partial v_x}{\partial x} + v_x \frac{\partial \rho}{\partial x} + \rho \frac{\partial v_y}{\partial y} + v_y \frac{\partial \rho}{\partial y}) \\ \frac{\partial v_x}{\partial t} = -(v_x \frac{\partial v_x}{\partial x} + v_y \frac{\partial v_x}{\partial y} + \frac{1}{\rho} \frac{\partial p}{\partial x}) \\ \frac{\partial v_y}{\partial t} = -(v_x \frac{\partial v_y}{\partial x} + v_y \frac{\partial v_y}{\partial y} + \frac{1}{\rho} \frac{\partial p}{\partial y}) \end{cases} \quad (14)$$

Ici v_x et v_y sont les composantes des vitesses, ρ est la densité d'un fluide.

Les équations d'Euler sont le cas limite des équations de Navier-Stokes pour les grands nombres de Reynolds. On néglige par la viscosité devant les termes d'inerties nonlinéaires.

- **Ecoulement bidimensionnel d'un fluide réel visqueux incompressible.**

$$\begin{cases} \frac{\partial v_x}{\partial t} = -v_x \frac{\partial v_x}{\partial x} - v_y \frac{\partial v_x}{\partial y} - g_x - \frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{\mu}{\rho} \left(\frac{\partial^2 v_x}{\partial x^2} + \frac{\partial^2 v_x}{\partial y^2} \right) \\ \frac{\partial v_y}{\partial t} = -v_y \frac{\partial v_y}{\partial y} - v_x \frac{\partial v_y}{\partial x} - g_y - \frac{1}{\rho} \frac{\partial p}{\partial y} + \frac{\mu}{\rho} \left(\frac{\partial^2 v_y}{\partial x^2} + \frac{\partial^2 v_y}{\partial y^2} \right) \\ \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} = 0 \end{cases} \quad (15)$$

- **Ecoulement monodimensionnel**

Si il n'y a pas de gradient de pression le long du parcours ($\frac{\partial p}{\partial x} = 0$) l'équation se transforme à l'équation de Burgers:

$$\frac{\partial v_x}{\partial t} + v_x \frac{\partial v_x}{\partial x} = \frac{\mu}{\rho} \frac{\partial^2 v_x}{\partial x^2}.$$

1.1.13 Equation de Burgers

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = c \frac{\partial^2 u}{\partial x^2}.$$

L'équation de Burgers monodimensionnelle modélise le mouvement d'un fluide compressible, en particulier la propagation des ondes de choc. Celles-ci sont par exemple des ondes de choc produites par des avions se déplaçant à une vitesse supersonic ou par des explosions. Les chocs apparaissent aussi dans des phénomènes faisant intervenir des vitesses plus petites. On peut s'intéresser au flux du pétrole dans des réservoirs de pétrole : l'huile et l'eau ne se mélangent pas et l'interface ainsi constituée forme ce que s'appelle choc. En 1859, le mathématicien Riemann a considéré le problème suivant: Soient deux cylindres de gaz de pressions différentes, séparés par une fine membrane. Que se passe-t-il si on enlève la membrane? Ce problème, appelé plus tard le problème de Riemann, se révèle être très compliqué. **La solution exacte a été trouvée par Peter Lax en 1957 qui a aussi apporté contribution importante dans la méthode aux différences finies appliquée aux équations non linéaires à savoir les méthodes de Lax et de Lax-Wendroff.**

1.1.14 Equations de Lagrange

$$\frac{\partial L}{\partial q} - \frac{\partial}{\partial t} \frac{\partial L}{\partial p} = 0$$

A chaque système physique on associe une fonction que s'appelle Lagrangien. L'équation de Lagrange décrit l'évolution du système dans le temps. En mécanique classique l'équation de Lagrange est équivalente à l'équation de Newton. En physique relativiste et en théorie quantique de champ l'équation de Lagrange décrit l'interaction entre des particules élémentaires, même leurs naissance et disparition.

1.2 Equations stationnaires

1.2.1 Equations d'Einstein

$$R_{\mu\nu} - \frac{1}{2}g_{\mu\nu}R_{\rho\rho} = 8\pi GT_{\mu\nu} + \Lambda g_{\mu\nu}$$

où $R_{\mu\nu}$ est le tenseur de Ricci, R le scalaire de Ricci, $g_{\mu\nu}$ le tenseur métrique Λ la constante cosmologique, G la constante gravitationnelle et $T_{\mu\nu}$ le tenseur énergie-impulsion.

Les solutions de l'équation d'Einstein sont les tenseurs métriques de l'espace-temps. Elles décrivent la structure de l'espace-temps en incluant le mouvement inertiel des objets. Du fait que les équations de champs ne sont pas linéaires, elles ne peuvent pas être complètement résolues (c'est-à-dire sans faire des approximations).

1.2.2 Equation de Poisson

En électrostatique le potentiel coulombien $V(x, y)$ satisfait l'équation suivante:

$$\begin{cases} \Delta V = -\rho(x, y) & x \in \Omega \\ V(x, y) = h(x, y) & x \in \partial\Omega \end{cases} \quad (16)$$

où $\rho(x, y)$ est la densité locale de charge.

1.2.3 Système de Lamé

Le système de Lamé est un cas particulier des équations stationnaires de l'élasticité linéarisée qui modélisent les déformations d'un solide sous l'hypothèse de petites déformations et de petites déplacements.

$$\begin{cases} -\mu\Delta u - (\mu + \lambda)\nabla(\operatorname{div} u) = f \\ u = 0 & x \in \partial\Omega \end{cases} \quad (17)$$

1.2.4 Equation des plaques

La solution des équations des plaques donne la déformation élastique d'une plaque plane d'épaisseur petite sous l'action de la force normale.

$$\begin{cases} \Delta(\Delta u) = f \\ u = 0 & x \in \partial\Omega \\ \frac{\partial u}{\partial n} = 0 & x \in \partial\Omega \end{cases} \quad (18)$$

1.2.5 Equation de Helmholtz

$$\begin{cases} \Delta u + k^2 u = 0 \\ u(t = 0, x) = g(x) \\ u = 0 & x \in \partial\Omega \end{cases} \quad (19)$$

L'équation de Helmholtz est une équation aux dérivées partielles elliptique qui apparaît lorsque l'on cherche des solutions stationnaires de l'équation de propagation des ondes d'Alembert, appelées " modes propres ". Pour que le problème mathématique soit bien posé, il faut spécifier une condition aux limites sur le bord du domaine, typiquement :

soit la condition de Dirichlet (le champ scalaire est nul sur le bord), soit la condition de Neumann.
soit la condition de Neumann (la dérivée normale du champ scalaire est nulle sur le bord).

2 Équations différentielles ordinaires

2.1 Problème de Cauchy

On considère une fonction continue $f : R_+ \times R \rightarrow R$.

Pour $y_0 \in R$ donné, on cherche $y : t \in R_+ \rightarrow y(t) \in R$ qui satisfait le problème suivant, appelé problème de Cauchy :

$$\begin{cases} \frac{dy(t)}{dt} = f(t, y(t)) \\ y(0) = y_0 \end{cases} \quad (20)$$

Un problème de Cauchy peut être linéaire, comme par exemple :

$$\begin{cases} \frac{dy(t)}{dt} = 3y(t) - 3t \\ y(0) = 1 \end{cases}$$

pour lequel $f(t, y) = 3y - 3t$ et dont la solution est $y(t) = (1 - 1/3)e^{3t} + t + 1/3$.

On peut avoir aussi des problèmes non-linéaires, comme :

$$\begin{cases} \frac{dy(t)}{dt} = y^{1/3}(t) \\ y(0) = 0 \end{cases}$$

avec $f(t, y) = y^{1/3}$. Ce problème admet les trois solutions suivantes :

$$y(t) = 0, y(t) = \sqrt{t^3/27}, y(t) = -\sqrt{t^3/27}.$$

Le problème suivant :

$$\begin{cases} \frac{dy(t)}{dt} = 1 + y^2(t) \\ y(0) = 0 \end{cases}$$

admet comme solution la fonction $y(t) = \tan(t)$ avec $0 < t < \pi/2$, c'est-à-dire une solution locale.

Théorème (de Cauchy-Lipschitz): Si f est continue sur $R_+ \times R$ et s'il existe une constante $L > 0$ telle que $|f(t, v) - f(t, w)| = L|v - w| \quad \forall v, w \in R, \forall t > 0$, alors le problème de Cauchy admet une solution globale (c-à-d. pour tout $t > 0$) et elle est unique.

2.2 Principes de la méthode aux Différences Finies

Pour discrétiser le temps, on introduit un pas temporel Δt . On définit un maillage où des coordonnées discrètes du temps sont

$$t_n = n\Delta t.$$

On note y_n la valeur d'une solution approchée calculée aux points $t = t_n$.

On note $y(t_n)$ la valeur de la solution exacte.

Le principe de la méthode des différences finies est de remplacer les dérivées par des différences finies en utilisant des formules de Taylor dans lesquelles on néglige les restes.

2.3 Formules de Taylor

$$y(t + \Delta t) = y(t) + \frac{dy}{dt}\Delta t + \frac{1}{2}\frac{d^2y}{dt^2}(\Delta t)^2 + \dots + \frac{1}{n!}\frac{d^ny}{dt^n}(\Delta t)^n + O((\Delta t)^{n+1}) \quad (1)$$

$$y(t - \Delta t) = y(t) - \frac{dy}{dt}\Delta t + \frac{1}{2}\frac{d^2y}{dt^2}(\Delta t)^2 + \dots + \frac{(-1)^n}{n!}\frac{d^ny}{dt^n}(\Delta t)^n + O((\Delta t)^{n+1}) \quad (2)$$

2.4 Discrétisation des dérivées premières

• Dérivée progressive temporelle:

Pour obtenir l'approximation de la dérivée on exprime la dérivée $\frac{dy}{dt}$ de la formule (1):

$$\frac{dy}{dt}(t_n) = \frac{y(t_n + \Delta t) - y(t_n)}{\Delta t} + T_1(y) = \frac{y(t_{n+1}) - y(t_n)}{\Delta t} + T_1(y)$$

Ici l'erreur de troncature est

$$T_1(y) = -\frac{1}{2!}\frac{d^2y}{dt^2}(\Delta t) - \frac{1}{3!}\frac{d^3y}{dt^3}(\Delta t)^2 + O((\Delta t)^3)$$

On observe que $T_1(y) = O(\Delta t)$ et on obtient

$$\frac{\partial y}{\partial t}(t_n) \approx \frac{y_{n+1} - y_n}{\Delta t} + O(\Delta t) \approx \frac{y_{n+1} - y_n}{\Delta t}$$

• Dérivée rétrograde temporelle:

$$\frac{\partial y}{\partial t}(t_n) = \frac{y(t_n) - y(t_n - \Delta t)}{\Delta t} + T_2(y) = \frac{y(t_n) - y(t_{n-1})}{\Delta t} + T_2(y)$$

Ici l'erreur de troncature est

$$T_2(y) = \frac{1}{2!}\frac{\partial^2 y}{\partial t^2}(\Delta t) - \frac{1}{3!}\frac{\partial^3 y}{\partial t^3}(\Delta t)^2$$

On observe que $T_2(y) = O(\Delta t)$ et on obtient:

$$\frac{dy}{dt}(t_n) \approx \frac{y_n - y_{n-1}}{\Delta t} + O(\Delta t) \approx \frac{y_n - y_{n-1}}{\Delta t}$$

• Dérivée centrée temporelle:

Pour obtenir l'approximation de la dérivée on soustraie (1) de (2) et on exprime la dérivée $\frac{dy}{dt}$ du résultat:

$$\frac{dy}{dt}(t_n) = \frac{y(t_n + \Delta t) - y(t_n - \Delta t)}{2\Delta t} + T_3(y) = \frac{y(t_{n+1}) - y(t_{n-1})}{2\Delta t} + T_3(y)$$

Ici l'erreur de troncature est

$$T_3(y) = -\frac{1}{6}\frac{d^3y}{dt^3}(\Delta t)^2 + O(\Delta t^3)$$

On observe que $T_3(y) = O(\Delta t^2)$ et on obtient:

$$\frac{dy}{dt}(t_n) \approx \frac{y_{n+1} - y_{n-1}}{2\Delta t} + O(\Delta t)^2 \approx \frac{y_{n+1} - y_{n-1}}{2\Delta t}$$

2.5 Discrétisation des dérivées secondes

On ajoute les deux séries de Taylor et après arrangement, on obtient l'approximation suivante du second ordre:

- **Dérivée temporelle:**

$$\frac{d^2 y}{dt^2}(t_n) = \frac{y(t + \Delta t) - 2y(t) + y(t - \Delta t)}{(\Delta t)^2} + T_4(y),$$

où l'erreur de troncature

$$T_4(y) = -\frac{1}{12} \frac{d^4 y}{dt^4} (\Delta t)^2 + O(\Delta t^4)$$

et observant que $T_4(y) = O(\Delta t^2)$ on obtient:

$$\frac{d^2 y}{dt^2}(t_n) = \frac{u(t_{n+1}) - 2u(t_n) + u(t_{n-1}))}{(\Delta t)^2} + T_4(u) \approx \frac{u_{n+1} - 2u_n + u_{n-1}}{(\Delta t)^2} + O(\Delta t)^2$$

2.6 Consistance

2.6.1 Erreur de troncature locale

On appelle erreur de troncature locale l'erreur commise lorsqu'on effectue **UN SEUL PAS de temps** du schéma. Autrement dit, on compare la solution de l'EDO $y(t_{n+1})$

$$\begin{cases} \left(\frac{dy(t)}{dt} = f(t, y(t)) \right) \\ y(t_n) = y_n \end{cases}$$

au temps t_{n+1} et la valeur y_{n+1} obtenue avec un schéma numérique. On la note parfois l'erreur

$$\varepsilon_n = y_{n+1} - y(t_{n+1})$$

2.6.2 Consistance du schéma

Le schéma sera dit consistant si l'erreur de troncature locale tend vers zero:

$$\lim_{\Delta t \rightarrow 0} \varepsilon_n = 0$$

Et il sera d'ordre p s'il existe K tel que:

$$|\varepsilon_n| \leq K(\Delta t)^p$$

2.6.3 Consistance du schéma d'Euler progressif ou explicite

Schéma d'Euler progressif :

$$\frac{y_{n+1} - y_n}{\Delta t} = f(t_n, y_n)$$

soit

$$y_{n+1} = y_n + \Delta t f(t_n, y_n)$$

Afin de démontrer la consistance du schéma, on effectue un développement limité de l'erreur de troncature locale:

$$\begin{aligned}
y(t_{n+1}) - y_{n+1} &= y(t_{n+1}) - (y_n + \Delta t f(t_n, y_n)) = \frac{dy}{dt} \Delta t + y(t_n) - T_1(y) - (y_n + \Delta t \frac{dy}{dt}) = \\
&= -T_1(y) = \frac{1}{2!} \frac{d^2 y}{dt^2} (\Delta t) + \frac{1}{3!} \frac{d^3 y}{dt^3} (\Delta t)^2 + \dots = \frac{1}{2!} \frac{d^2 y}{dt^2} (t_n + \eta \Delta t) (\Delta t), \quad \eta \in [t_n, t_{n+1}]
\end{aligned}$$

On en déduit qu'il existe K tel que

$$|y(t_{n+1}) - y_{n+1}| \leq K(\Delta t)^1$$

On en déduit que la méthode d'Euler explicite est consistante, et d'ordre 1.

2.6.4 Stabilité du schéma d'Euler progressif ou explicite

Montrons que la méthode d'Euler explicite est stable: on va comparer d'une part la solution approchée obtenue au temps n avec le schéma exact

$$\begin{cases} y_{n+1} = y_n + \Delta t f(t_n, y_n) \\ y(0) = y_0 \end{cases}$$

et d'autre part la solution obtenue avec le schéma perturbé:

$$\begin{cases} z_{n+1} = z_n + \Delta t f(t_n, z_n) + \Delta t \tau_n \\ z(0) = y_0 + \varepsilon_0 \end{cases}$$

On commence par recherche une égalité sur un seul pas de temps, que l'on va ensuite itérer:

$$z_{n+1} - y_{n+1} = (z_n - y_n + \Delta t (f(t_n, y_n) - f(t_n, z_n)) - \Delta t \tau_n$$

f est lipschitzienne, donc on en déduit l'inégalité

$$|z_{n+1} - y_{n+1}| = (1 + \Delta t L) |z_n - y_n| + \Delta t |\tau_n|$$

Il est facile de montrer par récurrence que

$$|z_n - y_n| = (1 + \Delta t L)^n |\varepsilon_0| + \Delta t \sum_{i=0}^{n-1} (1 + \Delta t L)^{n-1-i} |\tau_i|$$

Il vient alors

$$\Delta t \sum_{i=0}^{n-1} (1 + \Delta t L)^{n-1-i} |\tau_i| \leq \max |\tau_i| \Delta t \sum_{i=0}^{n-1} (1 + \Delta t L)^{n-1-i} \leq \max |\tau_i| \Delta t \sum_{i=0}^{n-1} (1 + \Delta t L)^i \leq \max |\tau_i| \Delta t \frac{(1 + \Delta t L)^n - 1}{L}$$

Enfin, on sait que $(1 + \Delta t L) \leq e^{\Delta t L}$, et on en déduit que $(1 + \Delta t L)^n \leq e^{\Delta t L n}$. On a de plus $n \Delta t \leq T$. Donc

$$|z_n - y_n| \leq e^{LT} |\varepsilon_0| + \frac{(e^{LT} - 1)}{L} \max |\tau_i|$$

soit

$$\exists K \in \mathbb{R} \quad \forall n \in \mathbb{N} \quad |z_n - y_n| \leq K \max(|\varepsilon_0| + \max |\tau_i|)$$

La méthode d'Euler explicite est donc stable.

Théorème sur la stabilité des méthodes à un pas temporel:

Soit une méthode à un pas de la forme:

$$y_{n+1} = y_n + \Phi(t_n, y_n, \Delta t)$$

alors la méthode est stable si

$$(t, y, \Delta t) \rightarrow \Phi(t, y, \Delta t)$$

est lipschitzienne par rapport à y .

2.7 Schéma de Runge-Kutta d'ordre 2

On étudie le schéma de Runge-Kutta suivant

$$\begin{cases} y_{n+1} = y_n + \Delta t f(t', y') \\ y' = y_n + \frac{\Delta t}{2} f(y_n, t_n) \\ t' = t_n + \frac{\Delta t}{2} \end{cases}$$

L'idée de schéma est de ne pas dériver la fonction f mais d'utiliser les arguments de f déplacés.

On peut réécrire le schéma de la forme:

$$y_{n+1} = y_n + \Delta t f\left(t_n + \frac{\Delta t}{2}, y_n + \frac{\Delta t}{2} f(y_n, t_n)\right)$$

2.7.1 Consistance du schéma de Runge-Kutta

On applique la formule de Taylor à la fonction de deux variables:

$$f\left(t_n + \frac{\Delta t}{2}, y_n + \frac{\Delta t}{2} f\right) = f(t_n, y_n) + \frac{\partial f}{\partial t} \frac{\Delta t}{2} + \frac{\partial f}{\partial y} \frac{\Delta t}{2} f + O((\Delta t)^2)$$

On utilise aussi les formules:

$$\begin{cases} y'(t) = f(t, y(t)) \\ y''(t) = \frac{\partial f}{\partial t} + f \frac{\partial f}{\partial y} \end{cases}$$

On en déduit

$$\begin{aligned} y(t_{n+1}) &= y(t_n) + y'(t_n) \Delta t + \frac{\Delta t^2}{2} y''(t_n) + O((\Delta t)^3) = \\ &= y(t_n) + f(t_n, y(t_n)) \Delta t + \frac{\Delta t^2}{2} \left(\frac{\partial f}{\partial t} + f \frac{\partial f}{\partial y} \right) + O((\Delta t)^3) \end{aligned}$$

Afin de démontrer la consistance du schéma, on effectue un développement limité de l'erreur de troncature locale:

$$\begin{aligned} y(t_{n+1}) - y_{n+1} &= y(t_{n+1}) - \left(y_n + \Delta t f\left(t_n + \frac{\Delta t}{2}, y_n + \frac{\Delta t}{2} f(y_n, t_n)\right) \right) = \\ &= y(t_n) + f(t_n, y(t_n)) \Delta t + \frac{(\Delta t)^2}{2} \left(\frac{\partial f}{\partial t} + f \frac{\partial f}{\partial y} \right) + O((\Delta t)^3) - y_n - \\ &\quad - f(t_n, y_n) \Delta t + \frac{\partial f}{\partial t} \frac{(\Delta t)^2}{2} + \frac{\partial f}{\partial y} \frac{(\Delta t)^2}{2} f + O((\Delta t)^3) = O((\Delta t)^3) \end{aligned}$$

Donc le schéma est consistant et d'ordre 2.

Montrons que ce schéma est stable. Pour cela, il suffit de montrer que

$$y \mapsto \Phi(t_n, y, \Delta t)$$

est lipschitzienne par rapport à y .

$$|\Phi(t_n, u, \Delta t) - \Phi(t_n, v, \Delta t)| = |f(t_n + \frac{\Delta t}{2}, u + \frac{\Delta t}{2}f(u, t_n)) - f(t_n + \frac{\Delta t}{2}, v + \frac{\Delta t}{2}f(v, t_n))| \leq$$

$$L|u + \frac{\Delta t}{2}f(u, t_n) - v - \frac{\Delta t}{2}f(v, t_n)| \leq L|u - v| + \frac{L\Delta t}{2}|f(u, t_n) - f(v, t_n)| \leq (L + \frac{L\Delta t}{2})|u - v|$$

2.8 Schéma de Runge-Kutta d'ordre 4

Il existe d'autres méthodes plus compliquées, comme par exemple la méthode de Runge-Kutta d'ordre 4 suivante:

$$\begin{cases} y_{n+1} = y_n + \Delta t(K_1 + 2K_2 + 2K_3 + K_4) \\ K_1 = f(t_n, y_n) \\ K_2 = f(t_n + \frac{\Delta t}{2}, y_n + \frac{\Delta t}{2}K_1) \\ K_3 = f(t_n + \frac{\Delta t}{2}, y_n + \frac{\Delta t}{2}K_2) \\ K_4 = f(t_{n+1}, y_n + \Delta tK_3) \end{cases}$$

2.9 Schéma implicite d'Euler et schéma implicite de Crank-Nicolson

Schéma d'Euler progressif :

$$y_{n+1} = y_n + \Delta t f(t_n, y_n)$$

Schéma d'Euler implicite :

$$y_{n+1} = y_n + \Delta t f(t_{n+1}, y_{n+1})$$

Schéma implicite de Crank-Nicolson:

$$y_{n+1} = y_n + \frac{\Delta t}{2}(f(t_n, y_n) + f(t_{n+1}, y_{n+1}))$$

2.10 Consistance du schéma implicite de Crank-Nicolson

Afin de démontrer la consistance du schéma, on effectue un développement limité de l'erreur de troncature locale. On compare la solution de l'EDO

$$\begin{cases} (\frac{dy(t)}{dt} = f(y(t), t)) \\ y(t_n) = y_n \end{cases}$$

au temps t_{n+1} et la valeur y_{n+1} obtenue avec le schéma numérique de Crank-Nicolson:

$$\begin{cases} y_{n+1} = y_n + \frac{\Delta t}{2}(f(t_n, y_n) + f(t_{n+1}, y_{n+1})) \\ y(t_n) = y_n \end{cases}$$

On soustrait et ajoute le même terme:

$$y(t_{n+1}) - y_{n+1} = y(t_{n+1}) - y_n - \frac{\Delta t}{2}(f(t_n, y_n) + f(t_{n+1}, y_{n+1})) =$$

$$= y(t_{n+1}) - y_n - \frac{\Delta t}{2}(f(t_n, y_n) + \boxed{f(t_{n+1}, y(t_{n+1}))}) - \frac{\Delta t}{2}(f(t_{n+1}, y_{n+1}) - \boxed{f(t_{n+1}, y(t_{n+1}))})$$

On en déduit en supposant que la fonction f est lipschitzienne:

$$|y(t_{n+1}) - y_{n+1}| \leq |y(t_{n+1}) - y_n - \frac{\Delta t}{2}(f(t_n, y_n) + f(t_{n+1}, y(t_{n+1})))| + \frac{\Delta t L}{2}|y(t_{n+1}) - y_{n+1}|$$

soit

$$\begin{aligned}
(1 - \frac{\Delta t L}{2})|y(t_{n+1}) - y_{n+1}| &\leq |y(t_{n+1}) - y_n - \frac{\Delta t}{2}(f(t_n, y_n) + f(t_{n+1}, y(t_{n+1})))| = \\
&= |y(t_{n+1}) - y_n - \frac{\Delta t}{2}(y'(t_n) + y'(t_{n+1}))| = \\
&= y(t_n) + \Delta t y'(t_n) + \frac{(\Delta t)^2}{2} y''(t_n) + \frac{(\Delta t)^3}{6} y^{(3)}(t_n) - y_n - \frac{\Delta t}{2}(y'(t_n) + y'(t_{n+1})) + \Delta t y''(t_n) + \frac{(\Delta t)^2}{2} y^{(3)}(t_n) = \\
&= \frac{(\Delta t)^3}{12} y^{(3)}(t_n)
\end{aligned}$$

On a utilisé $y(t_n) = y_n$ car on ne s'intéresse que de l'ordre du schéma. (On appelle erreur de troncature locale l'erreur commise lorsqu'on effectue **UN SEUL PAS de temps** du schéma.)

Il existe K tel que

$$|y(t_{n+1}) - y_n - \frac{\Delta t}{2}(f(t_n, y_n) + f(t_{n+1}, y(t_{n+1})))| \leq K(\Delta t)^3$$

Donc

$$|y(t_{n+1}) - y_{n+1}| \leq \frac{K(\Delta t)^3}{(1 - \frac{\Delta t L}{2})}$$

Le schéma implicite de Crank-Nicolson est consistant et d'ordre 2.

2.11 Convergence du schéma implicite de Crank-Nicolson

On étudie la convergence du schéma de Crank-Nicolson

$$\begin{cases} (\frac{dy(t)}{dt} = f(y(t), t) \\ y(t_0) = y_0 \end{cases}$$

On estime l'erreur après avoir effectué n pas au total:

$$\begin{aligned}
y(t_{n+1}) - y_{n+1} &= y(t_{n+1}) - y_n - \frac{\Delta t}{2}(f(t_n, y_n) + f(t_{n+1}, y_{n+1})) = \\
&= y(t_{n+1}) - y_n - \frac{\Delta t}{2}(f(t_n, y_n) + \boxed{f(t_{n+1}, y(t_{n+1}))}) + y(t_n) - y(t_n) - \frac{\Delta t}{2}(f(t_{n+1}, y_{n+1}) - \boxed{f(t_{n+1}, y(t_{n+1}))}) \\
&= |y(t_{n+1}) - y_{n+1}| = \\
&= |y(t_{n+1}) - y(t_n)| - \frac{\Delta t}{2}(f(t_n, y_n) + f(t_{n+1}, y(t_{n+1}))) + |y(t_n) - y_n| - \frac{\Delta t}{2}(f(t_{n+1}, y_{n+1}) - f(t_{n+1}, y(t_{n+1})))
\end{aligned}$$

De la même manière on peut montrer que:

$$\begin{aligned}
|y(t_{n+1}) - y_{n+1}| &\leq (1 + \frac{L\Delta t}{1 - \frac{\Delta t L}{2}})|y(t_n) - y_n| + \frac{K(\Delta t)^3}{(1 - \frac{\Delta t L}{2})} \\
n = 0 \quad |y(t_1) - y_1| &\leq (1 + \frac{L\Delta t}{1 - \frac{\Delta t L}{2}})|y(t_0) - y_0| + \frac{K(\Delta t)^3}{(1 - \frac{\Delta t L}{2})} = \frac{K(\Delta t)^3}{(1 - \frac{\Delta t L}{2})} \\
n = 1 \quad |y(t_2) - y_2| &\leq (1 + \frac{L\Delta t}{1 - \frac{\Delta t L}{2}})|y(t_1) - y_1| + \frac{K(\Delta t)^3}{(1 - \frac{\Delta t L}{2})} = 2 \frac{K(\Delta t)^3}{(1 - \frac{\Delta t L}{2})} \\
&\dots \\
\text{Pour } n \quad |y(t_{n+1}) - y_{n+1}| &\leq \frac{Kn(\Delta t)^3}{(1 - \frac{\Delta t L}{2})} \leq T \frac{(\Delta t)^2}{(1 - \frac{\Delta t L}{2})}
\end{aligned}$$

Donc si Δt tend vers zero la solution numerique (approchée) tend vers solution exacte, ce que signifie la convergence.

2.12 Applications

2.12.1 Compte Epargne : Calculette d'intérêts

Les intérêts générés par un livret épargne vérifie l'équation différentielle:

$$\frac{dy(t)}{dt} = ry(t),$$

où r est le taux d'intérêts annuel.

Soit le capital placé $y_0 = 100\epsilon$, $r = 0.06$ Calculons la somme sur le livret épargne après 1 ans en utilisant:

a) Méthode continue (Continuous compounding)

$$y(t) = y_0 e^{rt}, \quad y(1) = 106.1835$$

b) Méthode d'Euler explicite avec $\Delta t = 1/12$ (Cette méthode utilise l'industrie bancaire).

$$y_{n+1} = y_n + ry_n \Delta t$$

$$y_n = y_0(1 + r\Delta t)^n, \quad y_{12} = 106.1678$$

c) Méthode de Runge-Kutta d'ordre 2.

$$\begin{cases} y_{n+1} = y_n + f(y', t') \Delta t \\ y' = y_n + \frac{\Delta t}{2} f(y_n, t_n) \\ t' = t_n + \frac{\Delta t}{2} \end{cases}$$

Soit

$$\begin{cases} y_{n+1} = y_n + ry' \Delta t \\ y' = y_n + \frac{\Delta t}{2} ry_n \end{cases}$$

Soit

$$y_{n+1} = y_n + ry_n \Delta t + \frac{1}{2} r^2 y_n^2 (\Delta t)^2, \quad y_{12} = 106.183628$$

e) Méthode de Crank-Nicolson

$$y_{n+1} = y_n + ry_n + ry_{n+1} \frac{\Delta t}{2}$$

Donc

$$y_n = \left(\frac{1 + \frac{r\Delta t}{2}}{1 - \frac{r\Delta t}{2}} \right)^n, \quad y_{12} = 106.1836679$$

Il est rarement possible de trouver y_{n+1} explicitement. De façon générale on utilise la méthode de Newton pour trouver y_{n+1} .

2.12.2 Oscillations harmoniques

Appliquons les méthodes numériques suivantes pour résoudre l'équation différentielle des oscillations harmoniques:

$$\frac{d^2\Theta(t)}{dt^2} = -\omega^2\Theta(t)$$

Introduisons la fonction $v(t) = \frac{d\Theta(t)}{dt}$. Ecrivons l'équation de la forme:

$$\begin{cases} \frac{d\Theta(t)}{dt} = v(t) \\ \frac{dv(t)}{dt} = -\omega^2 \Theta(t) \end{cases}$$

a) Méthode d'Euler explicite

$$\begin{cases} \Theta_{n+1} = \Theta_n + \Delta t v_n \\ v_{n+1} = v_n - \omega^2 \Theta_n \Delta t \end{cases}$$

Ce méthode ne conserve pas l'énergie et fonctionne très mal. On l'abandonne.

b) Méthode d'Euler-Cromer

$$\begin{cases} \Theta_{n+1} = \Theta_n + \Delta t v_{n+1} \\ v_{n+1} = v_n - \omega^2 \Theta_n \Delta t \end{cases}$$

c) Méthode de Runge-Kutta d'ordre 2.

$$\begin{cases} \Theta_{n+1} = \Theta_n + v' \Delta t \\ v_{n+1} = v_n - \omega^2 \Theta' \Delta t \\ \Theta' = \Theta_n + \frac{\Delta t}{2} v_n \\ v' = v_n - \frac{\Delta t}{2} \omega^2 \Theta_n \end{cases}$$

2.12.3 Oscillateur nonlinéaire dissipatif

Appliquons les méthodes numériques suivantes pour résoudre l'équation différentielle des oscillations nonlinéaires:

$$\frac{d^2\Theta(t)}{dt^2} = -\omega^2 \sin(\Theta(t)) - k \frac{d\Theta(t)}{dt}$$

a) Méthode d'Euler-Cromer

$$\begin{cases} \Theta_{n+1} = \Theta_n + \Delta t v_{n+1} \\ v_{n+1} = v_n - (\omega^2 \sin \Theta_n + k \Theta_n) \Delta t \end{cases}$$

b) Méthode de Runge-Kutta d'ordre 2.

$$\begin{cases} \Theta_{n+1} = \Theta_n + v' \Delta t \\ v_{n+1} = v_n - (\omega^2 \sin \Theta' + k v') \Delta t \\ \Theta' = \Theta_n + \frac{\Delta t}{2} v_n \\ v' = v_n - \frac{\Delta t}{2} (\omega^2 \sin \Theta_n + k \Theta_n) \end{cases}$$

c) Méthode de Crank-Nicolson

$$\begin{cases} \Theta_{n+1} = \Theta_n + v_{n+1} \Delta t \\ v_{n+1} = v_n + \frac{\Delta t}{2} (-\omega^2 \sin \Theta_{n+1} - k v_{n+1} - \omega^2 \sin \Theta_n - k v_n) \end{cases}$$

On applique après la méthode de Newton.

2.12.4 Modèle de Lotka - Volterra

On considère maintenant deux populations, x et y , où x sont les proies et y sont les prédateurs. L'évolution des deux populations est alors décrite par le système d'équations différentielles

$$\begin{cases} \frac{dx(t)}{dt} = a x(t) - b x(t) y(t) \\ \frac{dy(t)}{dt} = -c y(t) + d x(t) y(t), \end{cases}$$

où a et c sont les facteurs de croissance des deux populations, b et d tiennent compte de l'interaction entre les deux populations. Ce système d'équations différentielles est connu comme modèle de Lotka-Volterra. Comment peut-on résoudre ce problème?

Ecrivons le système de la forme:

$$\frac{dS(t)}{dt} = f(S(t), t) = AS(t), \quad S(t) = (x(t), y(t)), \quad A = \begin{pmatrix} a & -bx(t) \\ dy(t) & -c \end{pmatrix}$$

Appliquons les méthodes numériques pour résoudre cette équation vectorielle:

a) Méthode de Runge-Kutta d'ordre 2.

$$\begin{cases} S_{n+1} = S_n + f(S', t') \Delta t \\ S' = S_n + \frac{\Delta t}{2} f(S_n, t_n) \\ t' = t_n + \frac{\Delta t}{2} \end{cases}$$

Soit

$$\begin{cases} x_{n+1} = x_n + (ax' - bx'y') \Delta t \\ y_{n+1} = y_n + (dx'y' - cy') \Delta t \\ x' = x_n + \frac{\Delta t}{2} (ax_n - bx_n y_n) \\ y' = x_n + \frac{\Delta t}{2} (dx_n y_n - cy_n) \end{cases}$$

b) Méthode de Crank-Nicolson

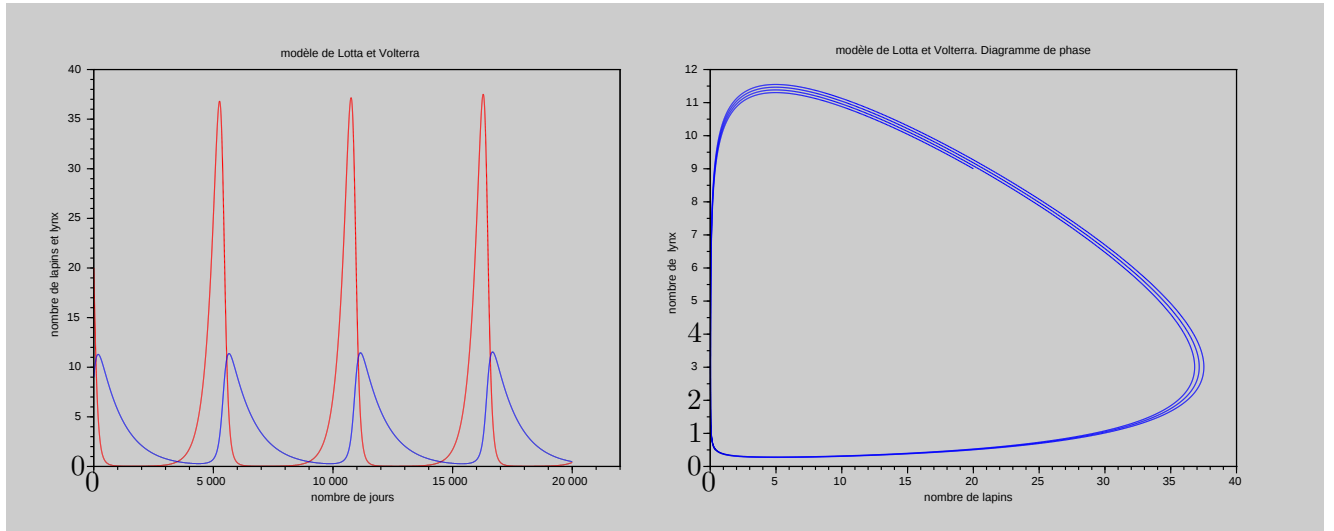
$$S_{n+1} = S_n + \frac{\Delta t}{2} (f(S_n, t_n) + f(S_{n+1}, t_{n+1})) \Delta t$$

$$\begin{cases} x_{n+1} = x_n + \frac{\Delta t}{2} (ax_n - bx_n y_n + ax_{n+1} - bx_{n+1} y_{n+1}) & (1) \\ y_{n+1} = y_n + \frac{\Delta t}{2} (dx_n y_n - cy_n + dx_{n+1} y_{n+1} - cy_{n+1}) & (2) \end{cases}$$

On exprime x_{n+1} de (1), on l'injecte dans (2), on calcule $y(n+1)$ à partir de x_n et y_n à l'aide de la méthode de Newton.

On présente ici les solutions obtenues par la méthode d'Euler explicite avec les constantes suivantes:

$$\begin{cases} x(1) = 20, \quad y(1) = 9 \\ N = 20000 \\ \Delta t = 0.001 \\ a = 2, \quad b = 1, \quad c = 1, \quad d = 0.2 \end{cases}$$



2.12.5 Mouvement des astres célestes

- Problème de deux corps: le mouvement de deux masses M et m

On écrit d'abord 2^{ème} loi de Newton qui régit le mouvement de la Terre de masse m autour du Soleil de masse M :

$$m \frac{d^2 \vec{r}(t)}{dt^2} = - \frac{m M G}{r(t)^2} \frac{\vec{r}(t)}{r(t)}$$

Ici $\vec{r}(t)$ est le vecteur position de la Terre, G est la constante fondamentale gravitationnelle. Le Soleil se trouve dans le centre du système de coordonnées. Projetons l'équation sur les axes OX et OY. On obtient deux équations scalaires:

$$\begin{cases} \frac{d^2 X(t)}{dt^2} = - \frac{G M X(t)}{(\sqrt{X(t)^2 + Y(t)^2})^3} \\ \frac{d^2 Y(t)}{dt^2} = - \frac{G M Y(t)}{(\sqrt{X(t)^2 + Y(t)^2})^3} \end{cases}$$

Introduisons les vitesses pour diminuer l'ordre des équations: $V_x(t) = \frac{dX(t)}{dt}$ et $V_y(t) = \frac{dY(t)}{dt}$ et on réécrit le système de la forme:

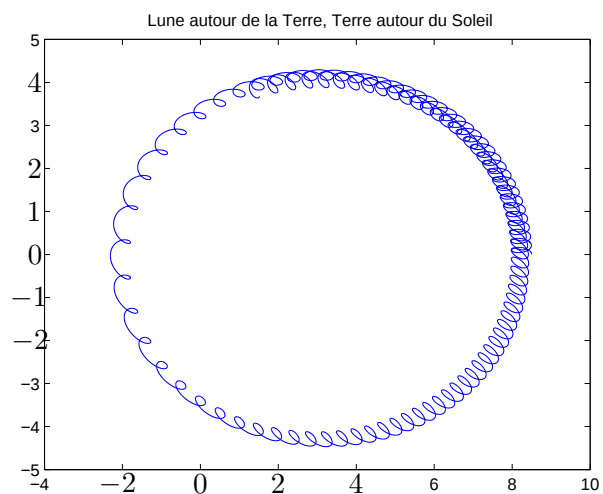
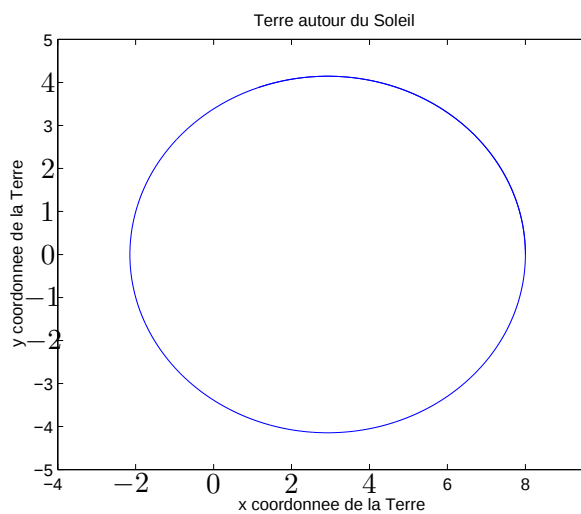
$$\begin{cases} \frac{dV_x}{dt} = - \frac{G M X(t)}{(\sqrt{X(t)^2 + Y(t)^2})^3} \\ \frac{dV_y}{dt} = - \frac{G M Y(t)}{(\sqrt{X(t)^2 + Y(t)^2})^3} \\ V_x(t) = \frac{dX}{dt} \\ V_y(t) = \frac{dY}{dt} \end{cases}$$

On résout par DF les équations en utilisant les algorithmes de Euler-Cromer. Discrétisons le système et notons $V_x = v$ et $V_y = u$.

$$\begin{cases} v_{n+1} = v_n - \frac{G M X_n}{(\sqrt{X_n^2 + Y_n^2})^3} \Delta t \\ u_{n+1} = u_n - \frac{G M Y_n}{(\sqrt{X_n^2 + Y_n^2})^3} \Delta t \\ X_{n+1} = X_n + \Delta t v_{n+1} \\ Y_{n+1} = Y_n + \Delta t u_{n+1} \end{cases}$$

On impose les conditions initiales et on trouve facilement les solutions en utilisant la réceptivité des relations obtenues et appliquant une seule boucle par rapport à n . On présente la trajectoire

de la Terre. Il n'est pas difficile d'ajouter la Lune. Dans ce cas on obtient le système de 8 equations que peut être aussi facilement résolu.



3 Principes de la méthode aux Différences Finies pour les EDP.

Pour discrétiser l'espace-temps, on introduit un pas d'espace Δx , et un pas du temps Δt . On définit un maillage ou des coordonnées discrètes de l'espace et du temps sont

$$(t^n, x_i) = (n\Delta t, i\Delta x).$$

Le principe de la méthode des différence finies est de remplacer les dérivées par des différences finies en utilisant des formules de Taylor dans lesquelles on néglige les restes.

3.1 Formule de Taylor

$$u(t, x + \Delta x) = u(t, x) + \frac{\partial u}{\partial x} \Delta x + \frac{1}{2} \frac{\partial^2 u}{\partial x^2} (\Delta x)^2 + \dots + \frac{1}{n!} \frac{\partial^n u}{\partial x^n} (\Delta x)^n + O((\Delta x)^{n+1}) \quad (1)$$

$$u(t + \Delta t, x) = u(t, x) + \frac{\partial u}{\partial t} \Delta t + \frac{1}{2} \frac{\partial^2 u}{\partial t^2} (\Delta t)^2 + \dots + \frac{1}{n!} \frac{\partial^n u}{\partial t^n} (\Delta t)^n + O((\Delta t)^{n+1}) \quad (2)$$

$$u(t, x - \Delta x) = u(t, x) - \frac{\partial u}{\partial x} \Delta x + \frac{1}{2} \frac{\partial^2 u}{\partial x^2} (\Delta x)^2 + \dots + \frac{(-1)^n}{n!} \frac{\partial^n u}{\partial x^n} (\Delta x)^n + O((\Delta x)^{n+1}) \quad (3)$$

$$u(t - \Delta t, x) = u(t, x) - \frac{\partial u}{\partial t} \Delta t + \frac{1}{2} \frac{\partial^2 u}{\partial t^2} (\Delta t)^2 + \dots + \frac{(-1)^n}{n!} \frac{\partial^n u}{\partial t^n} (\Delta t)^n + O((\Delta t)^{n+1}) \quad (4)$$

3.2 Discrétisation des dérivées premières

- **Dérivée progressive temporelle :**

Pour obtenir l'approximation de la dérivée on exprime la dérivée $\frac{\partial u}{\partial t}$ de la formule (2):

$$\frac{\partial u}{\partial t}(t_n, x_i) = \frac{u(t_n + \Delta t, x_i) - u(t_n, x_i)}{\Delta t} + T_1(u) = \frac{u(t_{n+1}, x_i) - u(t_n, x_i)}{\Delta t} + T_1(u)$$

Ici l'erreur de troncature est

$$T_1(u) = -\frac{1}{2!} \frac{\partial^2 u}{\partial t^2} (\Delta t) - \frac{1}{3!} \frac{\partial^3 u}{\partial t^3} (\Delta t)^2 + O((\Delta t)^3).$$

On observe que $T_1(u) = O(\Delta t)$ et on obtient l'approximation avec la précision $O(\Delta t)$ ou d'ordre 1 en temps:

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

- **Dérivée progressive spatiale :**

$$\frac{\partial u}{\partial x}(t_n, x_i) = \frac{u(t_n, x_i + \Delta t) - u(t_n, x_i)}{\Delta x} + T_1'(u) = \frac{u(t_n, x_{i+1}) - u(t_n, x_i)}{\Delta x} + T_1'(u)$$

Ici l'erreur de troncature est

$$T_1'(u) = -\frac{1}{2!} \frac{\partial^2 u}{\partial x^2}(\Delta x) - \frac{1}{3!} \frac{\partial^3 u}{\partial x^3}(\Delta x)^2 + O((\Delta x)^3).$$

On observe que $T_1'(u) = O(\Delta x)$ et on obtient l'approximation avec la précision $O(\Delta x)$ ou d'ordre 1 en espace:

$$\frac{\partial u}{\partial t}(t_n, x_i) \approx \frac{u_{i+1}^n - u_i^n}{\Delta t}$$

• **Dérivée rétrograde temporelle:**

Pour obtenir l'approximation de la dérivée on exprime la dérivée $\frac{\partial u}{\partial t}$ de la formule (4):

$$\frac{\partial u}{\partial t}(t_n, x_i) = \frac{u(t_n, x_i) - u(t_n - \Delta t, x_i)}{\Delta t} + T_2(u) = \frac{u(t_n, x_i) - u(t_{n-1}, x_i)}{\Delta t} + T_2(u)$$

Ici l'erreur de troncature est

$$T_2(u) = \frac{1}{2!} \frac{\partial^2 u}{\partial t^2}(\Delta t) - \frac{1}{3!} \frac{\partial^3 u}{\partial t^3}(\Delta t)^2 + O((\Delta t)^3)$$

On observe que $T_2(u) = O(\Delta t)$ et on obtient la précision $O(\Delta t)$:

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

• **Dérivée rétrograde spatiale:**

$$\frac{\partial u}{\partial x}(t_n, x_i) = \frac{u(t_n, x_i) - u(t_n, x_i - \Delta x)}{\Delta x} + T_2'(u) = \frac{u(t_n, x_i) - u(t_n, x_{i-1})}{\Delta x} + T_2'(u)$$

Ici l'erreur de troncature est

$$T_2'(u) = \frac{1}{2!} \frac{\partial^2 u}{\partial x^2}(\Delta x) - \frac{1}{3!} \frac{\partial^3 u}{\partial x^3}(\Delta x)^2 + O((\Delta x)^3)$$

On observe que $T_2(u) = O(\Delta x)$ et on obtient la précision $O(\Delta x)$:

$$\frac{\partial u}{\partial x}(t_n, x_i) \approx \frac{u_i^n - u_{i-1}^n}{\Delta x}$$

• **Dérivée centrée temporelle:**

Pour obtenir l'approximation de la dérivée on soustraie (4) de (2) et on exprime la dérivée $\frac{\partial u}{\partial x}$ du résultat:

$$\frac{\partial u}{\partial t}(t_n, x_i) = \frac{u(t_n + \Delta t, x_i) - u(t_n - \Delta t, x_i)}{2\Delta t} + T_3(u) = \frac{u(t_{n+1}, x_i) - u(t_{n-1}, x_i)}{2\Delta t} + T_3(u)$$

Ici l'erreur de troncature est

$$T_3(u) = -\frac{1}{6} \frac{\partial^3 u}{\partial t^3} (\Delta t)^2 + O((\Delta t)^3)$$

On observe que $T_3(u) = O(\Delta t^2)$ et on obtient la précision $O(\Delta t)^2$:

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

• **Dérivée centrée spatiale:**

$$\frac{\partial u}{\partial x}(t_n, x_i) = \frac{u(t_n, x_i + \Delta x) - u(t_n, x_i - \Delta x)}{2\Delta x} + T'_3(u) = \frac{u(t_n, x_{i+1}) - u(t_n, x_{i-1})}{2\Delta x} + T'_3(u)$$

Ici l'erreur de troncature est

$$T'_3(u) = -\frac{1}{6} \frac{\partial^3 u}{\partial x^3} (\Delta x)^2 + O((\Delta x)^3)$$

On observe que $T'_3(u) = O(\Delta x^2)$ et on obtient la précision $O(\Delta x)^2$:

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

3.2.1 Discrétisation des dérivées secondes

On ajoute les deux séries de Taylor (2) et (4) et après arrangement, on obtient l'approximation suivante du second ordre:

• **Dérivée temporelle:**

$$\frac{\partial^2 u}{\partial t^2}(t_n, x_i) = \frac{u(t + \Delta t, x) - 2u(t, x) + u(t - \Delta t, x))}{(\Delta t)^2} + T_4(u),$$

où l'erreur de troncature

$$T_4(u) = -\frac{1}{12} \frac{\partial^4 u}{\partial t^4} (\Delta t)^2 + O((\Delta t)^4)$$

et observant que $T_4(u) = O(\Delta t^2)$ on obtient la précision $O(\Delta t)^2$:

$$\frac{\partial^2 u}{\partial t^2}(t_n, x_i) = \frac{u(t^{n+1}, x_i) - 2u(t_n, x_i) + u(t^{n-1}, x_i))}{(\Delta t)^2} + T_4(u) \approx \frac{u_i^{n+1} - 2u_i^n + u_i^{n-1}}{(\Delta t)^2}$$

• **Dérivée seconde spatiale:**

$$\frac{\partial^2 u}{\partial x^2}(t_n, x_i) = \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2} + T_5(u) \approx \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2}$$

où l'erreur de troncature est

$$T_5(u) = -\frac{1}{12} \frac{\partial^4 u}{\partial x^4} (\Delta x)^2 + O((\Delta x)^4)$$

et on obtient la précision $O(\Delta x)^2$:

4 Stratégie de base dans les approches de discrétisation

4.1 Maillage

On doit résoudre numériquement l'équation de chaleur sur l'intervalle $x \in [0, L]$:

$$\begin{cases} \frac{\partial u}{\partial t} - \beta \frac{\partial^2 u}{\partial x^2} = 0 \\ u(0, x) = g(x) & \text{condition initiale} \\ u(t, 0) = h(t) & \text{condition aux limites} \\ u(t, L) = \xi(t) & \text{condition aux limites} \end{cases} \quad (21)$$

Ici, nous utilisons des conditions aux limites de Dirichlet.

Discretisons les variables spatiale et temporelle

x : $x_0 = 0, \dots, x_i = i\Delta x, \dots, x_{N+1} = L$.

t : $t_0 = 0, \dots, t^n = n\Delta t, \dots, t_{M+1} = T$.

On note u_i^n la valeur d'une solution calculée à l'aide d'un schéma numérique aux points $x = x_i, t = t^n$.

On note $u(t^n, x_i)$ la valeur de la solution exacte calculée aux points $x = x_i, t = t^n$.

Donc

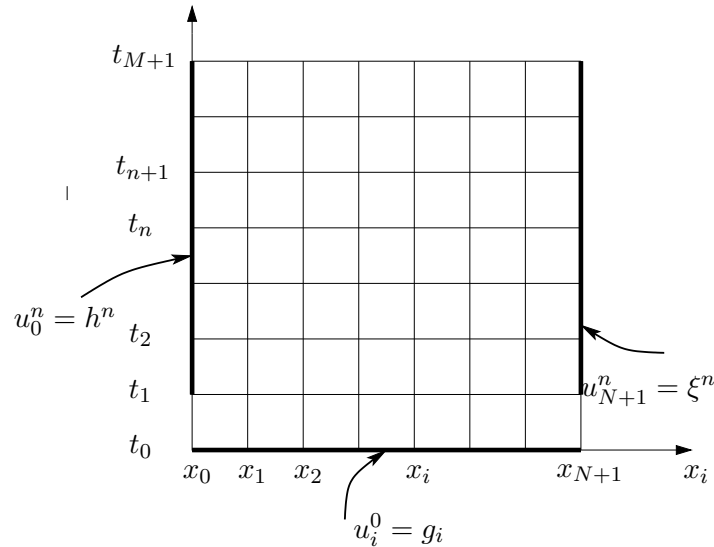
$$\begin{cases} i = 0, 1, 2, \dots, N + 1 \\ n = 0, 1, 2, \dots, M + 1 \\ \text{On cherche } u_i^n \end{cases} \quad (22)$$

Conditions aux limites:

$$\begin{cases} u_0^n = h(n\Delta t) = h^n \\ u_{N+1}^n = \xi(n\Delta t) = \xi^n \\ n = 1, 2, \dots, M + 1 \end{cases} \quad (23)$$

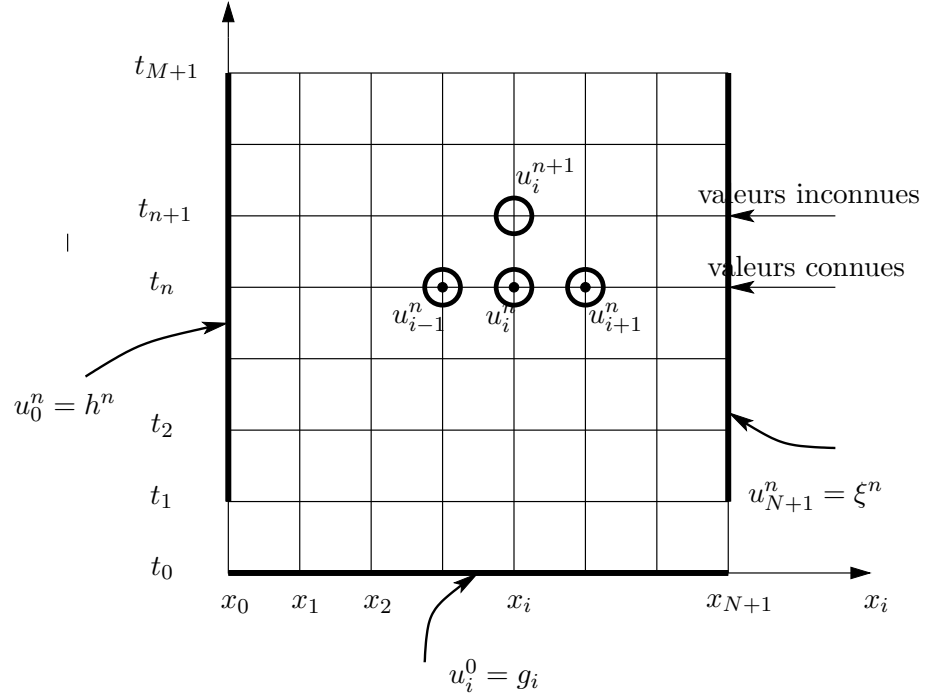
Conditions initiales :

$$\{ u_i^0 = g(i\Delta x) = g_i, i = 0, 1, 2, \dots, N + 1 \} \quad (24)$$



4.2 Méthodes explicites

4.2.1 Méthode d'Euler progressive



Utilisons la dérivée temporelle progressive et la dérivée spatiale centrée. On obtient l'équation discrète qui représente une approximation de l'équation originale continue:

$$\left\{ \begin{array}{l} \frac{u_i^{n+1} - u_i^n}{\Delta t} = \beta \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2} \\ u_i^0 = g_i \\ u_0^n = h^n \\ u_{N+1}^n = \xi^n \end{array} \right. \quad (25)$$

On appelle schéma numérique explicite, un schéma pour lequel une seule inconnue apparaît dans l'équation aux différences finies, d'une telle manière qu'il permet son évaluation en fonction des valeurs connues, ici les u_{i-1}^n , u_i^n , u_{i+1}^n par une simple récurrence:

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

4.2.2 Implementation du schéma explicite d'Euler progressive

On écrit les conditions initiales et aux limites avec deux boucles séparées, puis on construit deux boucles telles que l'une est à l'intérieur de l'autre:

Algorithme d'Euler explicite

Pour $i = 0, \dots, N + 1$

$$u_i^0 = g_i$$

Fin Pour

Pour $n = 1, \dots, M + 1$

$$u_0^n = h^n$$

$$u_{N+1}^n = \xi^n$$

Fin Pour

Pour $n = 0, \dots, M$

Pour $i = 1, \dots, N$

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

Fin Pour

Fin Pour

En effet, on tourne l'algorithme à la main pour $u_i^{n+1} = u_i^n + \frac{\Delta t \beta}{(\Delta x)^2} (u_{i+1}^n - 2u_i^n + u_{i-1}^n)$:

$n = 0$ toutes les valeurs u_i^0 sont connues de la condition initiale: $u_i^0 = g_i$

$$\left\{ \begin{array}{ll} i = 1 & u_1^1 = \boxed{u_1^0} + \frac{\Delta t \beta}{(\Delta x)^2} (\boxed{u_2^0} - 2\boxed{u_1^0} + \boxed{u_0^0}) \\ i = 2 & u_2^1 = \boxed{u_2^0} + \frac{\Delta t \beta}{(\Delta x)^2} (\boxed{u_3^0} - 2\boxed{u_2^0} + \boxed{u_1^0}) \\ & \dots \\ i = N & u_N^1 = \boxed{u_N^0} + \frac{\Delta t \beta}{(\Delta x)^2} (\boxed{u_{N+1}^0} - 2\boxed{u_N^0} + \boxed{u_{N-1}^0}) \end{array} \right.$$

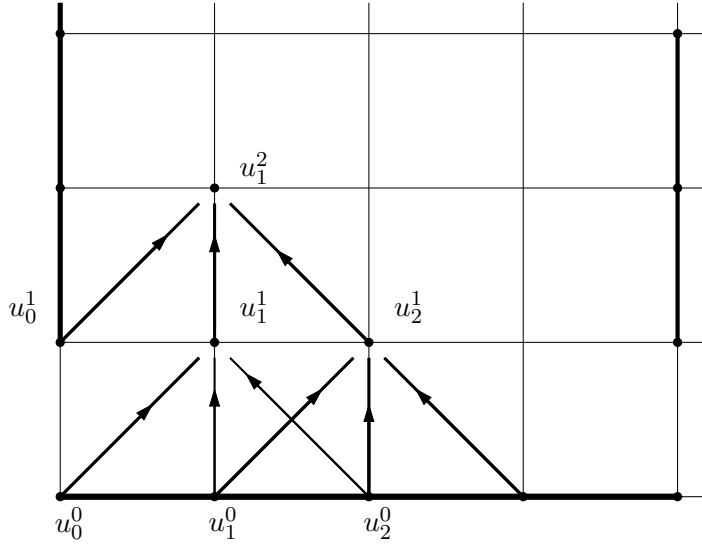
$n = 1$ les valeurs u_i^1 ($i=1, \dots, N$) sont calculées à l'iteration $n = 0$, u_0^1 et u_{N+1}^1 sont connues de conditions aux limites: $u_0^1 = h^1$, $u_{N+1}^1 = \xi^1$.

$$\left\{ \begin{array}{ll} i = 1 & u_1^2 = u_1^1 + \frac{\Delta t \beta}{(\Delta x)^2} (u_2^1 - 2u_1^1 + \boxed{u_0^1}) \\ i = 2 & u_2^2 = u_2^1 + \frac{\Delta t \beta}{(\Delta x)^2} (\boxed{u_3^1} - 2u_2^1 + u_1^1) \\ & \dots \\ i = N & u_N^2 = u_N^1 + \frac{\Delta t \beta}{(\Delta x)^2} (\boxed{u_{N+1}^1} - 2u_N^1 + u_{N-1}^1) \end{array} \right.$$

$n = M$

$$\left\{ \begin{array}{ll} i = 1 & u_1^M = u_1^{M-1} + \frac{\Delta t \beta}{(\Delta x)^2} (u_2^{M-1} - 2u_1^{M-1} + \boxed{u_0^{M-1}}) \\ i = 2 & u_2^M = u_2^{M-1} + \frac{\Delta t \beta}{(\Delta x)^2} (\boxed{u_3^{M-1}} - 2u_2^{M-1} + u_1^{M-1}) \\ & \dots \\ i = N & u_N^M = u_N^{M-1} + \frac{\Delta t \beta}{(\Delta x)^2} (\boxed{u_{N+1}^{M-1}} - 2u_N^{M-1} + u_{N-1}^{M-1}) \end{array} \right.$$

On peut visualiser cette explication par un dessin suivant:



On voit qu'on commence par calculer $u_1^1, u_2^1, \dots, u_i^1, \dots$ à l'aide de la condition initiale $u_i^0 = g_i$. Puis on passe à $n = 2$. On calcule u_1^2 en utilisant u_0^1 (condition aux limites), u_1^1, u_2^1 . Ensuite on calcule $u_2^2 \dots$ etc.

4.2.3 Forme matricielle de l'équation discrete

On présente ici deux formes matricielles linéaire de l'équation discrete avec les conditions aux limites de Dirichlet et $C = \frac{\Delta t \beta}{(\Delta x)^2}$:

$$\begin{pmatrix} u_1^{n+1} \\ u_2^{n+1} \\ u_3^{n+1} \\ \vdots \\ u_N^{n+1} \end{pmatrix} = \begin{pmatrix} 1-2C & C & 0 & 0 & 0 \\ C & 1-2C & C & 0 & 0 \\ & & C & 1-2C & C \\ & & 0 & C & 1-2C \end{pmatrix} \cdot \begin{pmatrix} u_1^n \\ u_2^n \\ u_3^n \\ \vdots \\ u_N^n \end{pmatrix} - \begin{pmatrix} C \cdot u_0^n \\ 0 \\ 0 \\ \vdots \\ 0 \\ C \cdot u_{N+1}^n \end{pmatrix}$$

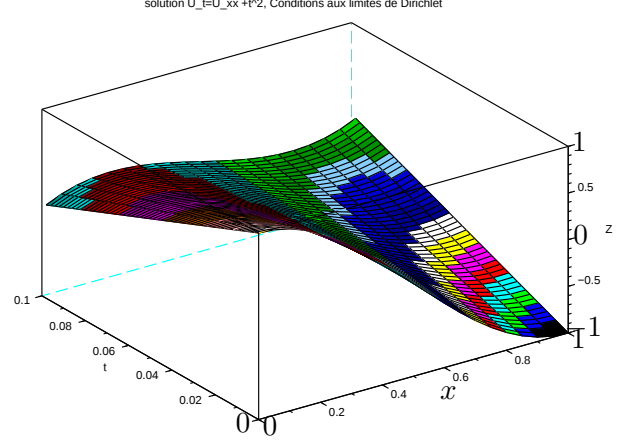
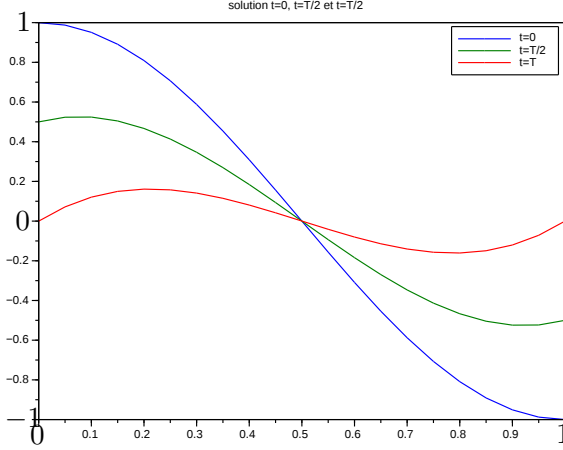
où

$$\begin{pmatrix} u_0^{n+1} \\ u_1^{n+1} \\ u_2^{n+1} \\ \vdots \\ u_{N+1}^{n+1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1-2C & C & 0 & 0 \\ 0 & C & 1-2C & C & 0 \\ & & C & 1-2C & C \\ & & 0 & C & 1-2C \\ & & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} u_0^n \\ u_1^n \\ u_2^n \\ \vdots \\ u_{N+1}^n \end{pmatrix}$$

4.2.4 Visualisation des solutions

On trace en 2 et 3 dimensions les solutions de l'équation suivante avec les conditions aux limites de Dirichlet:

$$\begin{cases} u_t = u_{xx} + t^2 \\ u(0, x) = \cos(\pi x) \\ u(t, 0) = 1 - 10t \\ u(t, 1) = 1 + 10t \end{cases} \quad (26)$$



4.3 Méthodes implicites

4.3.1 Méthode de Crank-Nicolson

On doit résoudre numériquement l'équation de chaleur:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \beta \frac{\partial^2 u}{\partial x^2} + f(t^n, x_i).$$

Le schéma explicite simple donne:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \beta \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2} + f(t^n, x_i)$$

La méthode de Crank-Nicolson consiste à remplacer chaque fonction u_i^n à l'instant n par la fonction moyenne temporelle $(u_i^n + u_i^{n+1})/2$.

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{\beta}{(\Delta x)^2} \left[\frac{1}{2}(u_{i+1}^n + u_{i+1}^{n+1}) - 2\frac{1}{2}(u_i^n + u_i^{n+1}) + \frac{1}{2}(u_{i-1}^n + u_{i-1}^{n+1}) \right] + \frac{1}{2}(f_i^n + f_i^{n+1})$$

On peut réécrire cette équation sous la forme :

$$B_i u_{i-1}^{n+1} + D_i u_i^{n+1} + A_i u_{i+1}^{n+1} = K_i^n, \quad (27)$$

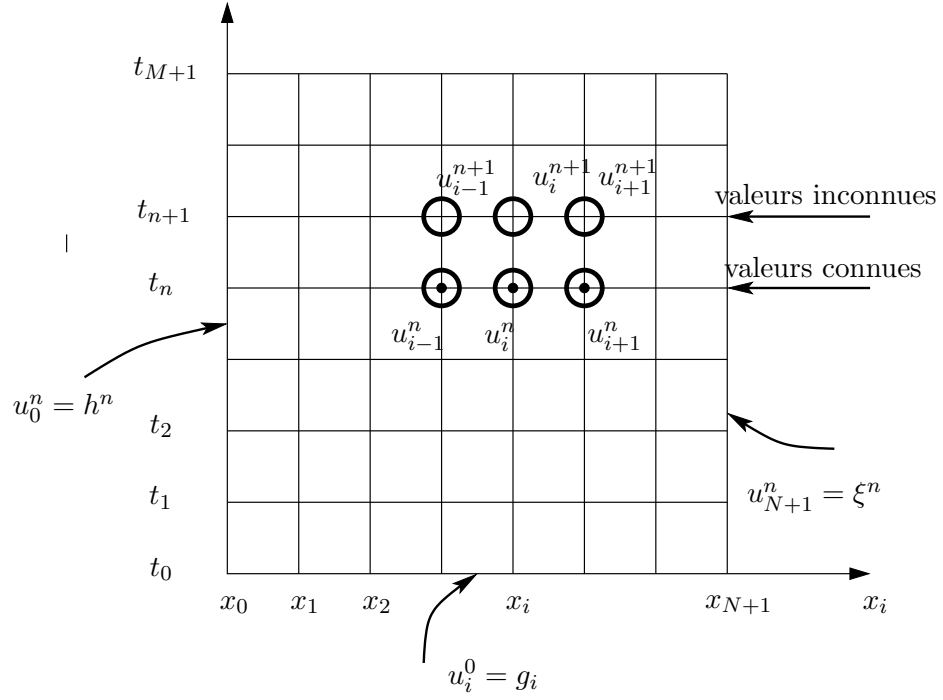
où

$$A_i = -\frac{\Delta t \beta}{2(\Delta x)^2}, \quad (28)$$

$$B_i = -\frac{\Delta t \beta}{2(\Delta x)^2}, \quad (29)$$

$$D_i = 1 + \frac{\Delta t \beta}{(\Delta x)^2}, \quad (30)$$

$$K_i^n = \left(1 - \frac{\Delta t \beta}{(\Delta x)^2}\right) u_i^n + \frac{\Delta t \beta}{2(\Delta x)^2} (u_{i+1}^n + u_{i-1}^n) + \frac{\Delta t}{2} (f_i^n + f_i^{n+1}) \quad (31)$$



On réécrit l'équation sous la forme de système linéaire :

$$i = 1, \quad B_1 u_0^{n+1} + D_1 u_1^{n+1} + A_1 u_2^{n+1} = K_1^n$$

$$i = 2, \quad B_2 u_1^{n+1} + D_2 u_2^{n+1} + A_2 u_3^{n+1} = K_2^n$$

.....

$$i = N, \quad B_N u_{N-1}^{n+1} + D_N u_N^{n+1} + A_N u_{N+1}^{n+1} = K_N^n$$

On utilise les conditions aux limites $u_0^{n+1} = h^{n+1}$, $u_{N+1}^{n+1} = \xi^{n+1}$ et on obtient la forme matricielle suivante:

$$\begin{pmatrix} D_1 & A_1 & 0 & 0 & 0 \\ B_2 & D_2 & A_2 & 0 & 0 \\ 0 & B_3 & D_3 & A_3 & 0 \\ & & & & \\ & & & B_{N-1} & D_{N-1} & A_{N-1} \\ & & & 0 & B_N & D_N \end{pmatrix} \cdot \begin{pmatrix} u_1^{n+1} \\ u_2^{n+1} \\ u_3^{n+1} \\ \vdots \\ u_N^{n+1} \end{pmatrix} = \begin{pmatrix} K_1^n - B_1 h^{n+1} \\ K_2^n \\ K_3^n \\ \vdots \\ K_{N-1}^n \\ K_N^n - A_N \xi^{n+1} \end{pmatrix}$$

Or

$$M \cdot U^{n+1} = K^n$$

Notre but est de trouver $U^{n+1} = (u_N^{n+1}, u_{N-1}^{n+1}, u_{N-2}^{n+1}, \dots, u_1^{n+1})$, exprimées en termes des $(u_N^n, u_{N-1}^n, u_{N-2}^n, \dots, u_1^n)$. On ne peut plus calculer directement u_i^{n+1} . Pour cela il faut inverser la matrice car

$$U^{n+1} = M^{-1} K^n.$$

Pour obtenir la matrice inverse d'une **matrice tridiagonale** on applique l'Algorithme de Thomas.

4.3.2 Forme matricielle de l'équation discrète et Algorithme de Thomas

On cherche la solution du système

$$\begin{cases} d_1 u_1 + a_1 u_2 = c_1 \\ b_2 u_1 + d_2 u_2 + a_2 u_3 = c_2 \\ b_3 u_2 + d_3 u_3 + a_3 u_4 = c_3 \\ \vdots \\ b_{m-1} u_{m-2} + d_{m-1} u_{m-1} + a_{m-1} u_m = c_{m-1} \\ b_m u_{m-1} + d_m u_m = c_m \end{cases} \quad (32)$$

qui peut être écrite aussi en forme matricielle:

$$\begin{pmatrix} d_1 & a_1 & 0 & 0 & 0 \\ b_2 & d_2 & a_2 & 0 & 0 \\ 0 & b_3 & d_3 & a_3 & 0 \\ & & & & \\ & & & b_{m-1} & d_{m-1} & a_{m-1} \\ & & & 0 & b_m & d_m \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_m \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_{m-1} \\ c_m \end{pmatrix}$$

Déduisons l'algorithme de Thomas de façon générale. On multiplie 1ere équation sur b_2 , 2eme sur d_1 , on déduit 1ere de 2eme et on obtient:

$$\begin{aligned} b_2(d_1 u_1 + a_1 u_2) &= b_2 c_1 \\ d_1(b_2 u_1 + d_2 u_2 + a_2 u_3) &= d_1 c_2 \\ u_2(d_2 - \frac{b_2 a_1}{d_1}) + a_2 u_3 &= c_2 - \frac{b_2 c_1}{d_1} \end{aligned}$$

Notons

$$d_2^* = d_2 - \frac{b_2 a_1}{d_1}, \quad c_2^* = c_2 - \frac{b_2 c_1}{d_1}.$$

On continue la même procedure pour les équation 2 et 3, 3 et 4,... Finalement on obtient le système:

$$\begin{cases} d_1 u_1 + a_1 u_2 = c_1 \\ d_2^* u_2 + a_2 u_3 = c_2^* \\ d_3^* u_3 + a_3 u_4 = c_3^* \\ d_{m-1}^* u_{m-1} + a_{m-1} u_m = c_{m-1}^* \\ d_m^* u_m = c_m^* \end{cases} \quad (33)$$

Les coefficients se calculent de facon iterative.

$$\begin{cases} d_i^* = d_i - \frac{b_i a_{i-1}}{d_{i-1}^*}, & i = 2, 3, \dots, m \\ d_1^* = d_1 \end{cases} \quad (34)$$

$$\begin{cases} c_i^* = c_i - \frac{b_i c_{i-1}^*}{d_{i-1}^*}, & i = 2, 3, \dots, m \\ c_1^* = c_1 \end{cases} \quad (35)$$

La solution se calcule aussi de facon iterative en commençant par **l'indice maximal** $i = m$ **et en descendant**:

$$\begin{cases} u_m = \frac{c_m^*}{d_m^*} \\ u_i = \frac{c_i^* - a_i u_{i+1}}{d_i^*}, & i = m-1, \dots, 1 \end{cases} \quad (36)$$

Revenons à l'équation de chaleur écrite de facon générale. Faisons les identifications:

$$\begin{cases} d_i = D_i, & i = 1, \dots, N \\ a_i = A_i, & i = 1, \dots, N-1 \\ b_i = B_i, & i = 2, \dots, N \end{cases} \quad (37)$$

$$\begin{pmatrix} K_1^n - B_1 h^{n+1} \\ K_2^n \\ K_3^n \\ \vdots \\ K_{N-1}^n \\ K_N^n - A_N \xi^{n+1} \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_{m-1} \\ c_m \end{pmatrix}$$

Les solutions pour les fonctions u_i^{n+1} en terms des fonctions u_i^n sont données par les formules récurrentes:

$$\begin{cases} u_N^{n+1} = \frac{K_N^{*n}}{D_N^*} \\ u_i^{n+1} = \frac{K_i^{*n} - A_i u_{i+1}^{n+1}}{D_i^*}, & i = N-1, \dots, 1 \\ D_i^* = D_i - \frac{B_i A_{i-1}}{D_{i-1}^*}, & i = 2, \dots, N, \quad D_1^* = D_1 \\ K_i^{*n} = K_i^n - \frac{B_i K_{i-1}^{*n}}{D_{i-1}^*}, & i = 2, \dots, N-1, \quad K_1^{*n} = K_1^n - B_1 h^{n+1} \\ K_N^{*n} = K_N^n - A_N \xi^{n+1} - \frac{B_N K_{N-1}^{*n}}{D_{N-1}^*} \end{cases} \quad (38)$$

Si on utilise les propriétés de l'équation de chaleur:

$$B_i = A_i = A = -\frac{\Delta t \beta}{2(\Delta x)^2}, \quad D_i = D = 1 + \frac{\Delta t \beta}{(\Delta x)^2}$$

on obtient

$$\left\{ \begin{array}{l} K_1^{*n} = K_1^n - B_1 h^{n+1} \quad K_i^{*n} = K_i^n - \frac{AK_{i-1}^{*n}}{D_{i-1}^*}, \quad i = 2, \dots, N-1 \\ K_N^{*n} = K_N^n - A\xi^{n+1} - \frac{AK_{N-1}^{*n}}{D_{N-1}^*} \\ D_1^* = D_1 = D, \quad D_i^* = D - \frac{A^2}{D_{i-1}^*}, \quad i = 2, \dots, N \\ u_N^{n+1} = \frac{K_N^{*n}}{D_N^*} \\ u_i^{n+1} = \frac{K_i^{*n} - Au_{i+1}^{n+1}}{D_i^*}, \quad i = N-1, \dots, 1 \end{array} \right. \quad (39)$$

5 Equations paraboliques. Résolution numérique de l'équation de la chaleur par Crank-Nicolson

5.1 Algorithme de Crank-Nicolson

On fait une boucle temporelle et l'algorithme de Thomas à l'intérieur. Pour corriger les valeurs K_1^n et K_N^n on utilise les symboles de Kroneker δ_{ij} : $\delta_{ij} = 1$ si $i = j$ et $\delta_{ij} = 0$ si $i \neq j$.

On suppose que les coefficients A_i , B_i et D_i de la matrice M sont précisés.

Algorithme de Crank-Nicolson. Conditions aux limites de Dirichlet.

$$A_i = -\frac{\Delta t \beta}{2(\Delta x)^2}; \quad B_i = -\frac{\Delta t \beta}{2(\Delta x)^2}; \quad D_i = 1 + \frac{\Delta t \beta}{(\Delta x)^2}$$

Pour $i = 0, N+1$

$$u_i^0 = g_i$$

Fin Pour

Pour $n = 1, M+1$

$$u_0^n = h^n$$

$$u_{N+1}^n = \xi^n$$

Fin Pour

Pour $n = 0, M$

Pour $i = 1, N$

$$K_i^n = (1 - \frac{\Delta t \beta}{(\Delta x)^2})u_i^n + \frac{\Delta t \beta}{2(\Delta x)^2}(u_{i+1}^n + u_{i-1}^n) + \frac{\Delta t}{2}(f_i^n + f_i^{n+1}) - \delta_{1i} \cdot B_1 \cdot h^{n+1} - \delta_{Ni} \cdot A_N \cdot \xi^{n+1}$$

Fin Pour

$$D_1^* = D_1, \quad K_1^{*n} = K_1^n$$

Pour $i = 2, N$

$$D_i^* = D_i - \frac{B_i \cdot A_{i-1}}{D_{i-1}^*}, \quad K_i^{*n} = K_i^n - \frac{B_i \cdot K_{i-1}^{*n}}{D_{i-1}^*}$$

Fin Pour

$$u_N^{n+1} = \frac{K_N^{*n}}{D_N^*}$$

Pour $i = N - 1, -1, 1$

$$u_i^{n+1} = \frac{K_i^{*n} - A_i \cdot u_{i+1}^{n+1}}{D_i^*}$$

Fin Pour

Fin Pour

En effet , pour $n = 0$ la valeur

$$K_i^0 = (1 - \frac{\Delta t \beta}{(\Delta x)^2})u_i^0 + \frac{\Delta t \beta}{2(\Delta x)^2}(u_{i+1}^0 + u_{i-1}^0) + \frac{\Delta t}{2}(f_i^0 + f_i^1) - \delta_{1i} B_1 h^1 - \delta_{Ni} A_N \xi^1$$

est connue, car u_i^0 représente les conditions initiales.

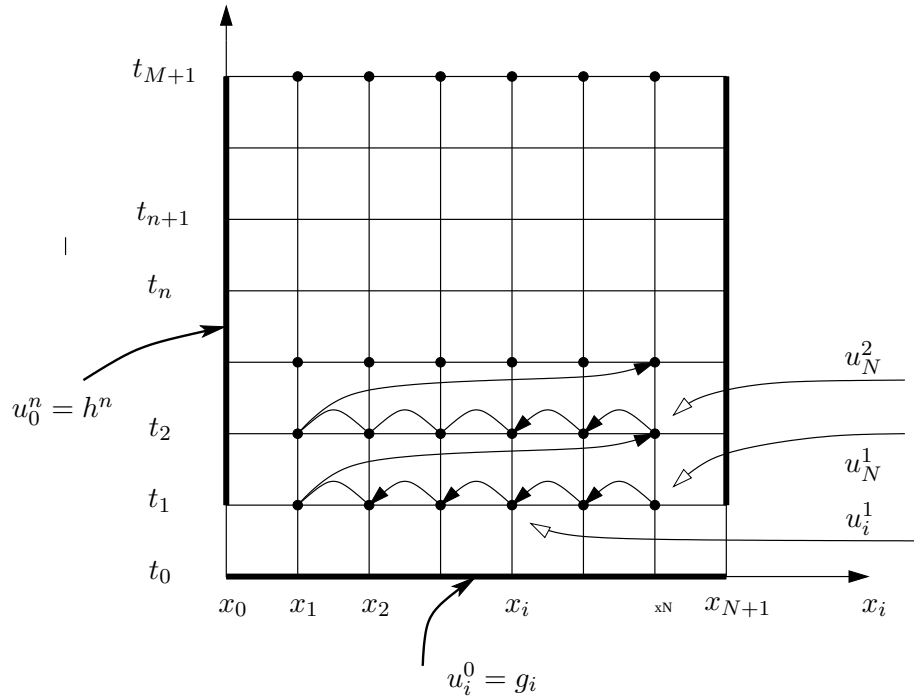
On calcule les valeurs u_i^1 : d'abord $u_N^1, u_{N-1}^1, \dots, u_1^1$ comme indiqué par les fleches sur la figure.

Puis pour $n = 1$ la valeur

$$K_i^1 = (1 - \frac{\Delta t \beta}{(\Delta x)^2})u_i^1 + \frac{\Delta t \beta}{2(\Delta x)^2}(u_{i+1}^1 + u_{i-1}^1) + \frac{\Delta t}{2}(f_i^1 + f_i^2) - \delta_{1i} B_1 h^2 - \delta_{Ni} A_N \xi^2$$

se calcule à l'aide de u_i^1 déjà calculées.

On calcule sur le niveau $n = 2$ les valeurs u_i^2 : d'abord $u_N^2, u_{N-1}^2, \dots, u_1^2$ comme indiqué par les fleches sur la figure.



5.2 Implémentation en MATLAB ou SCILAB.

- **Discrétisation de l'espace-temps**

$$\circ \quad x = (0 : N + 1) \cdot \Delta x \quad \text{ou} \quad x = \text{linspace}(0, L, N + 2)$$

Par conséquent les coordonnées portent les vraies valeurs :

$$x(1) = 0,$$

$$x(2) = \Delta x,$$

$$x(3) = 2\Delta x, \dots,$$

$$x(N + 2) = (N + 1) * \Delta x \equiv L$$

Il y a $N + 2$ composantes.

$$\circ \quad t = (0 : M + 1) \cdot \Delta t \quad \text{ou} \quad t = \text{linspace}(0, T, M + 2)$$

Par conséquent les coordonnées portent les vraies valeurs : $t(1) = 0$,

$$t(2) = \Delta t,$$

$$t(3) = 2\Delta t, \dots,$$

$$t(M + 2) = (M + 1) * \Delta t \equiv T$$

Il y a $M + 2$ composantes.

- **Programmation structurée**

```
function[f]=condition_initiale(x)
```

```
f = ...
```

```
endfunction
```

- **On déplacer de 1 tout les indices fixes**

```
Pour i = 1 : N + 2
```

```
u(1,i)=condition_initiale(x(i))
```

```
Fin Pour
```

```
figure;
```

```
plot(x,u(1,:));
```

```
title('condition initiale')
```

```
Pour n = 2 : M + 2
```

```
u(n,1)=Condition1_limite(t(n))
```

```
u(n,N + 2)=Condition2_limite(t(n))
```

```
Fin Pour
```

```
figure;
```

```
plot(t,u(:, N+2),t,u(:,1));
title('condition aux limites')
```

Le schéma est toujours stable.

- **Programme principal**

Algorithme de Crank-Nicolson pour MATLAB. Condition aux limites de Dirichlet.

$$A(i) = -\frac{\Delta t \beta}{2(\Delta x)^2}; \quad B(i) = -\frac{\Delta t \beta}{2(\Delta x)^2}; \quad D(i) = 1 + \frac{\Delta t \beta}{(\Delta x)^2}$$

Pour $i = 1 : N + 2$

$$u(1, i) = g(x(i))$$

Fin Pour

Pour $n = 2 : M + 2$

$$u(n, 1) = h(t(n))$$

$$u(n, N + 2) = \xi(t(n))$$

Fin Pour

Pour $n = 1 : M + 1$

Pour $i = 2, N + 1$

$$K(n, i) = (1 - \frac{\Delta t \beta}{(\Delta x)^2})u(n, i) + \frac{\Delta t \beta}{2(\Delta x)^2}(u(n, i + 1) + u(n, i - 1)) + \frac{\Delta t}{2}(f(t(n), x(i)) + f(t(n + 1), x(i))) - \delta_{2,i} \cdot B(2) \cdot h(t(n + 1)) - \delta_{N+1,i} \cdot A(N + 1) \cdot \xi(t(n + 1))$$

Fin Pour

$$Detoile(2) = D(2), \quad Ketoile(2, n) = K(2, n)$$

Pour $i = 3 : N + 1$

$$Detoile(i) = D(i) - \frac{B(i) \cdot A(i-1)}{Detoile(i-1)}, \quad Ketoile(n, i) = K(n, i) - \frac{B(i) \cdot Ketoile(n, i-1)}{Detoile(i-1)}$$

Fin Pour

$$u(n + 1, N + 1) = \frac{Ketoile(n, N+1)}{Detoile(N+1)}$$

Pour $i = N : -1, 2$

$$u(n + 1, i) = \frac{Ketoile(n, i) - A(i) \cdot u(n+1, i+1)}{Detoile(i)}$$

Fin Pour

Fin Pour

5.3 Visualisation des solutions

Considérons le problème suivant:

- On étudie la distribution de la température le long d'une tige métallique. Une extrémité est maintenue à la température $u(x = 0) = 0$, l'autre est maintenue à la température $u(x = L) = 50$. Il n'y a pas des sources extérieures de la chaleur ($f(x, t) = 0$). La température le long d'une tige vérifie l'équation:

$$\begin{cases} \frac{\partial u}{\partial t} - \beta \frac{\partial^2 u}{\partial x^2} = 0 \\ u(0, x) = g(x) & \text{condition initiale} \\ u(t, 0) = 0 & \text{condition aux limites} \\ u(t, L) = T_0 & \text{condition aux limites} \end{cases} \quad (40)$$



Pour trouver les solutions on a utilisé la méthode de Crank-Nicolson avec les données suivantes

$$\begin{cases} L = 20, \quad T = 10, \quad g(x) = 0 \\ \Delta x = 0.2, \quad \Delta t = 0.1 \\ N = 100, \quad M = 100 \end{cases}$$

La figure 1 présente la distribution de la température aux instants: $t = 0$, $t = T$ et $t = T/2$ pour les conditions initiales: $u(t = 0, x) = 0$

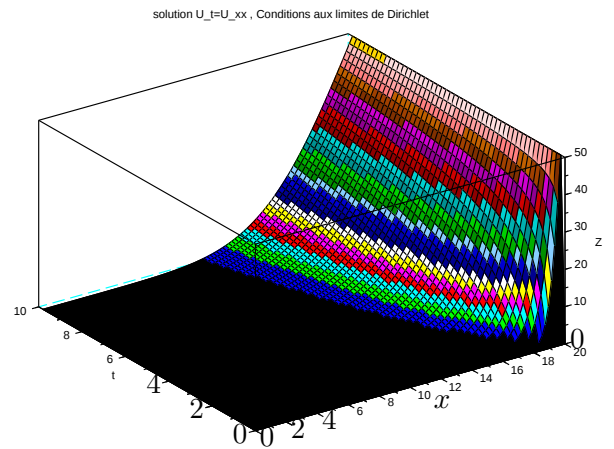
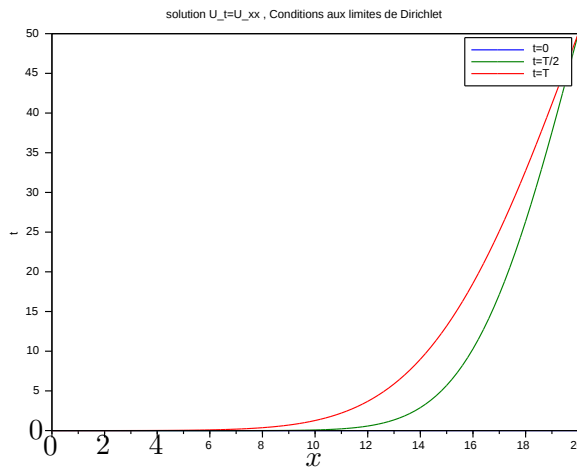


Fig. 1

La figure 2 présente la distribution de la température aux instants: $t = 0$, $t = T$ et $t = T/2$ pour les conditions initiales:

$$u(t = 0, x) = \frac{50}{\sqrt{\pi}} e^{-\frac{(x-10)^2}{2}}$$

La température extrémité $x = L$ varie en fonction du temps $u(x = L) = 5t$.

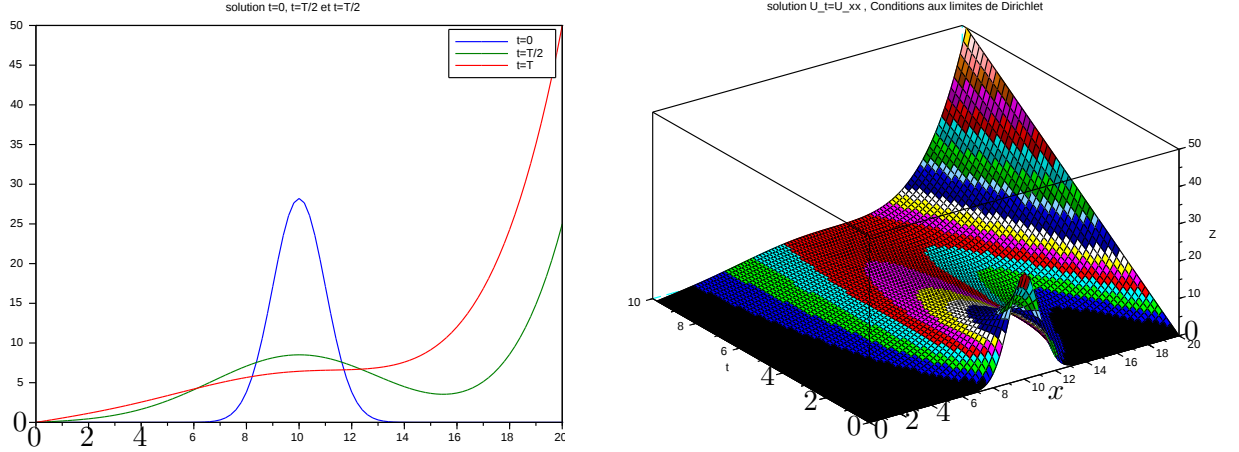
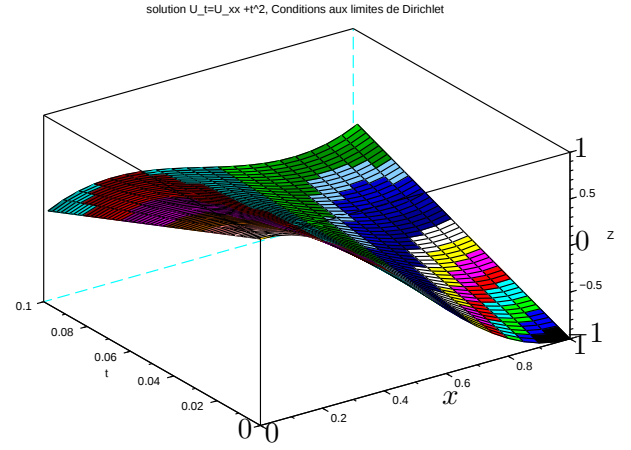
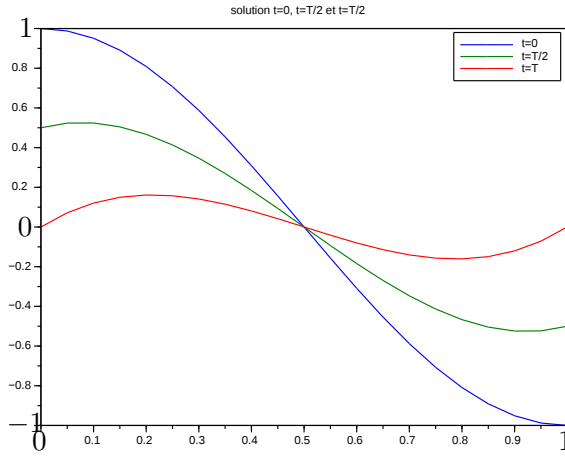


Fig. 1

5.4 Comparaisons des conditions aux limites

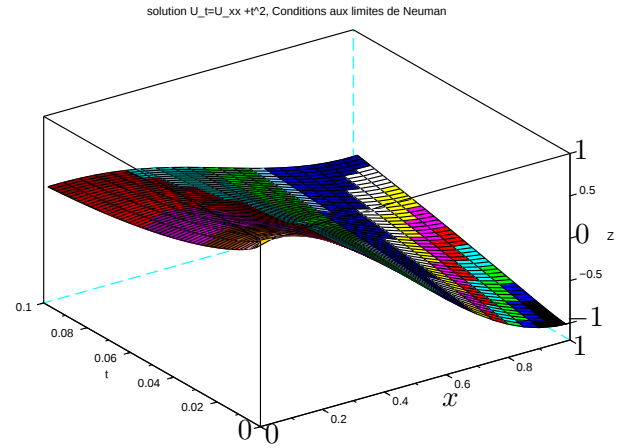
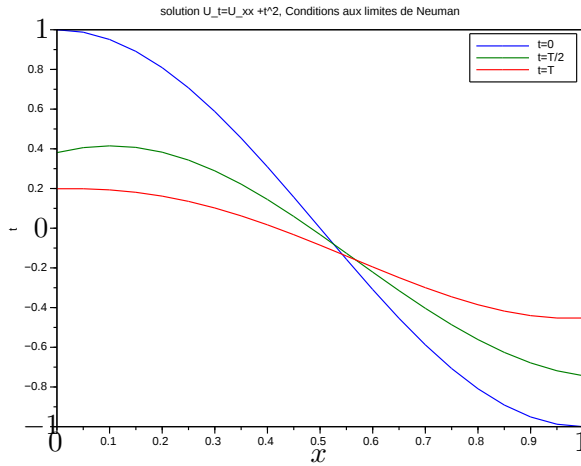
- On trace en 2 et 3 dimensions la solution de l'équation suivante avec les conditions aux limites de Dirichlet pour $T = 0.1$:

$$\left\{ \begin{array}{l} u_t = u_{xx} + t^2 \\ u(0, x) = \cos(\pi x) \\ u(t, 0) = 1 - 10t \\ u(t, 1) = 1 + 10t \\ x \in [0, 1] \\ T = 0.1 \end{array} \right. \quad (41)$$

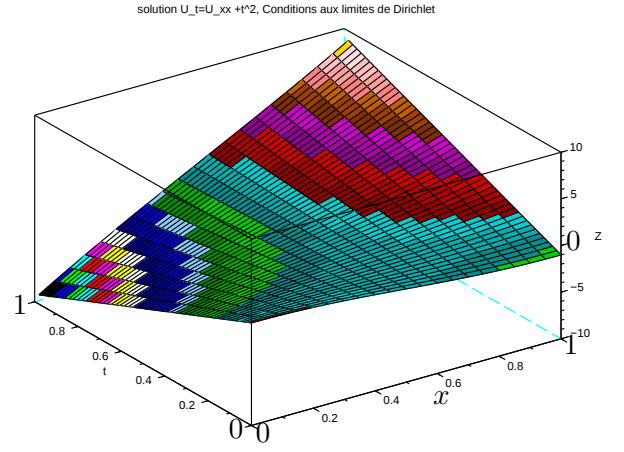
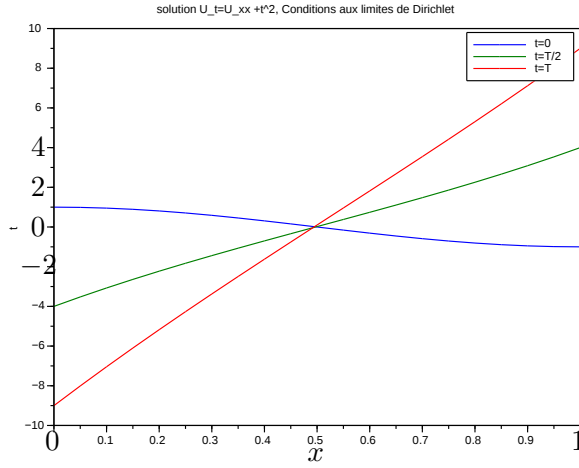


- On trace en 2 et 3 dimensions la solution de l'équation suivante avec les conditions aux limites de Neumann pour $T = 0.1$:

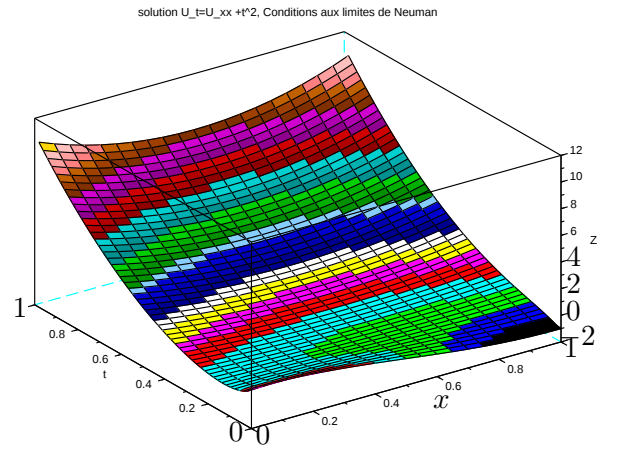
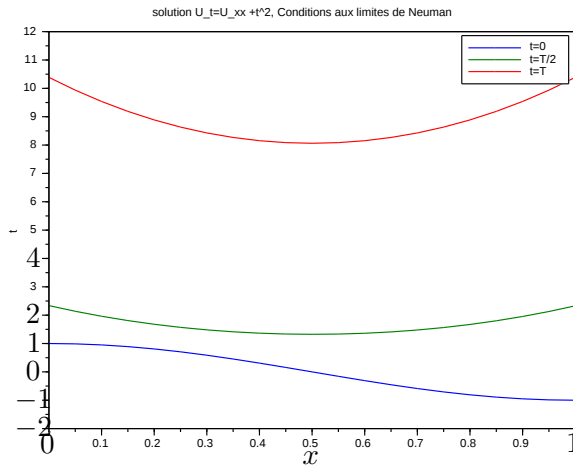
$$\begin{cases} u_t = u_{xx} + t^2 \\ u(0, x) = \cos(\pi x) \\ u_x(t, 0) = 1 - 10t \\ u_x(t, 1) = 1 + 10t \\ x \in [0, 1] \\ T = 0.1 \end{cases} \quad (42)$$



- On trace en 2 et 3 dimensions la solution de la même équation avec les conditions aux limites de Dirichlet pour $T = 1$:



- On trace en 2 et 3 dimensions la solution de la même équation avec les conditions aux limites de Neumann pour $T = 1$:



5.5 Méthode d'Euler rétrograde ou implicite.

Appliquons pour les dérivées : la dérivée temporelle rétrograde et la dérivée seconde spatiale. On obtient:

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

Il est facile de vérifier que ce schéma est implicite et bien défini, c'est-à-dire qu'on peut calculer les valeurs u_i^{n+1} en fonction des u_i^n . On introduit les notations:

$$C = -\frac{\Delta t \beta}{(\Delta x)^2}, \quad (43)$$

$$D = 1 + \frac{\Delta t \beta}{(\Delta x)^2}, \quad (44)$$

$$Q_i^n = u_i^n + f_i^n \quad (45)$$

On réécrit l'équation sous la forme :

$$Cu_{i-1}^{n+1} + Du_i^{n+1} + Cu_{i+1}^{n+1} = Q_i^n, \text{ soit} \quad (46)$$

$$\begin{pmatrix} D & C & 0 & 0 & 0 \\ C & D & C & 0 & 0 \\ 0 & C & D & C & 0 \\ & & & C & D & C \\ & & & 0 & C & D \end{pmatrix} \begin{pmatrix} u_1^{n+1} \\ u_2^{n+1} \\ u_3^{n+1} \\ \vdots \\ \vdots \\ u_N^{n+1} \end{pmatrix} = \begin{pmatrix} Q_1^n - Ch^{n+1} \\ Q_2^n \\ Q_3^n \\ \vdots \\ Q_{N-1}^n \\ Q_N^n - C\xi^{n+1} \end{pmatrix}$$

Pour calculer les valeurs u_i^{n+1} en fonction des u_i^n il faut inverser la matrice. On applique l'algorithme de Thomas.

6 Traitement des conditions aux limites de Neumann

6.1 Conditions aux limites de Neumann d'ordre 1

On peut remplacer les conditions aux limites de Dirichlet par des conditions aux limites de Neumann. Il existe deux manières différentes de discrétiser les conditions aux limites de Neumann:

$$\frac{\partial u}{\partial x}(t, 0) = h(t) \quad \frac{\partial u}{\partial x}(t, L) = \xi(t)$$

On peut les discrétiser de façon suivante:

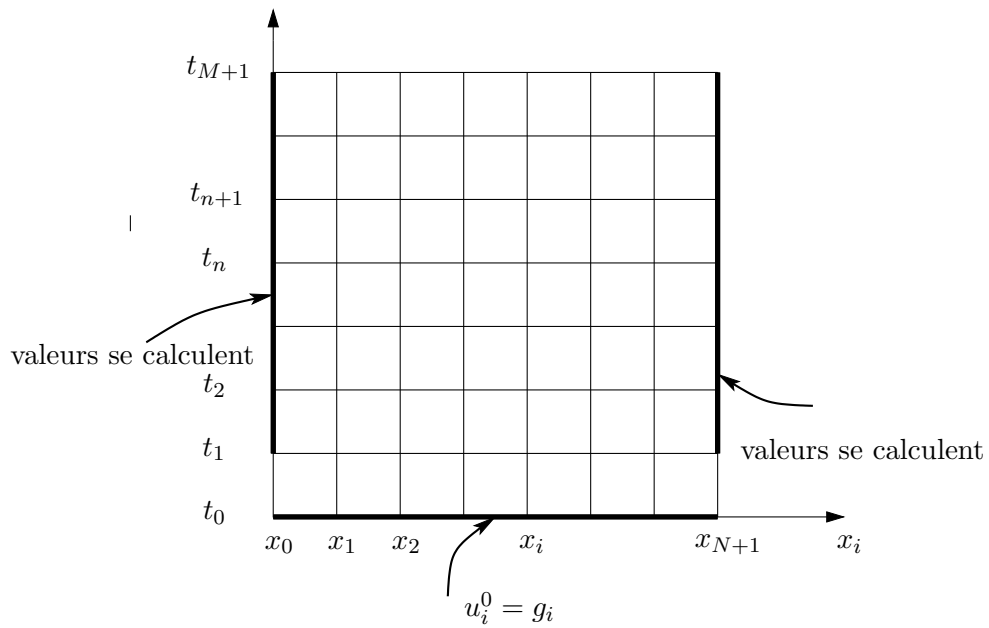
$$\frac{u_1^n - u_0^n}{\Delta x} = h^n \quad \frac{u_{N+1}^n - u_N^n}{\Delta x} = \xi^n$$

soit

$$u_0^n = u_1^n - \Delta x h^n \quad u_{N+1}^n = u_N^n + \Delta x \xi^n$$

6.2 Implementation des conditions aux limites de Neumann d'ordre 1 en utilisant la méthode d'Euler explicite

On peut visualiser le maillage:



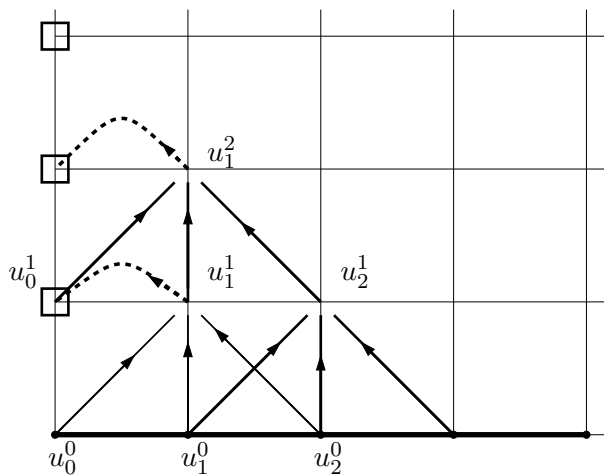
Algorithme d'Euler explicite avec les conditions aux limites de Neumann

```

Pour     $i = 0, \dots, N + 1$ 
 $u_i^0 = g_i$ 
Fin Pour
Pour     $n = 0, \dots, M$ 
  Pour     $i = 1, \dots, N$ 
 $u_i^{n+1} = u_i^n + \frac{\Delta t \beta}{(\Delta x)^2} (u_{i+1}^n - 2u_i^n + u_{i-1}^n)$ 
  Fin Pour
 $u_0^{n+1} = u_1^{n+1} - \Delta x h^{n+1}$ 
 $u_{N+1}^{n+1} = u_N^{n+1} + \Delta x \xi^{n+1}$ 
Fin Pour

```

On visualise cette explication par un dessin suivant:



On voit qu'on commence par calculer $u_1^1, u_2^1, \dots, u_i^1, \dots$ à l'aide de la condition initiale $u_i^0 = g_i$.
Puis on calcule u_0^1 à l'aide de la condition aux limites de Neumann $u_0^n = u_1^n - \Delta x h^n$.
Puis on passe à $n = 2$. On calcule u_1^2 en utilisant u_0^1, u_1^1, u_2^1 . Ensuite on calcule $u_2^2 \dots$ etc.

6.3 Implementation des conditions aux limites de Neumann d'ordre 2 en utilisant la méthode d'Euler explicite

Si le schéma est du deuxième ordre, discrétisation de la condition de Neumann du premier ordre engendre une perte de précision près du bord. C'est pourquoi on propose une autre discrétisation (du deuxième ordre):

$$\frac{u_1^n - u_{-1}^n}{2\Delta x} = h^n \quad \frac{u_{N+2}^n - u_N^n}{2\Delta x} = \xi^n$$

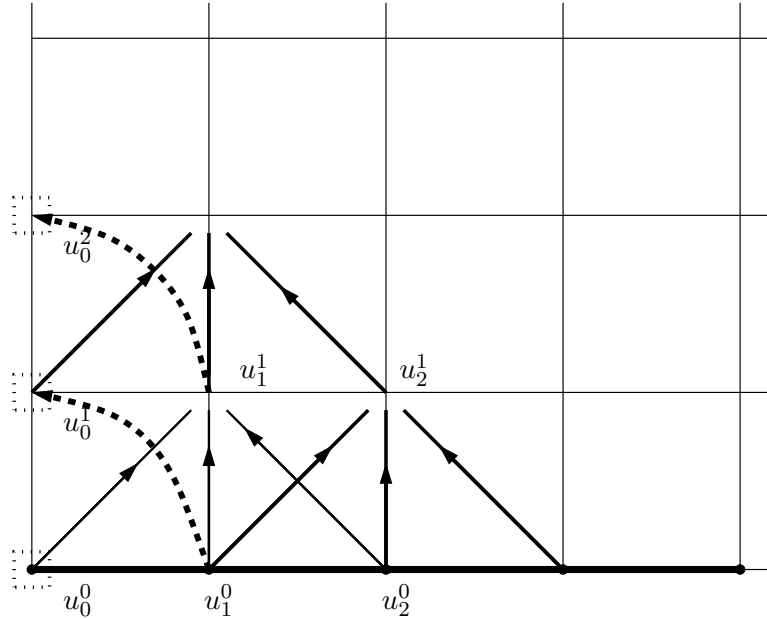
qui est plus précise, mais nécessite l'ajout de 2 points fictifs, et il reste maintenant $N + 2$ valeurs à calculer, à savoir $(u_i^n)_{0 \leq i \leq N+1}$.

En effet pour $i = 0$ et $i = N + 1$ nous avons les équations:

$$\begin{cases} u_0^{n+1} = u_0^n + \frac{\Delta t \beta}{(\Delta x)^2} (u_1^n - 2u_0^n + \boxed{u_{-1}^n}) \\ u_{N+1}^{n+1} = u_{N+1}^n + \frac{\Delta t \beta}{(\Delta x)^2} (\boxed{u_{N+2}^n} - 2u_{N+1}^n + u_N^n) \end{cases}$$

qui deviendront après l'injection $u_{-1}^n = u_1^n - 2\Delta x h^n$ et $u_{N+2}^n = u_N^n + 2\Delta x \xi^n$

$$\begin{cases} u_0^{n+1} = u_0^n + \frac{\Delta t \beta}{(\Delta x)^2} (u_1^n - 2u_0^n + u_1^n - 2\Delta x h^n) = u_0^n + \frac{\Delta t \beta}{(\Delta x)^2} (2u_1^n - 2u_0^n - 2\Delta x h^n) \\ u_{N+1}^{n+1} = u_{N+1}^n + \frac{\Delta t \beta}{(\Delta x)^2} (u_N^n - 2u_{N+1}^n + u_N^n + 2\Delta x \xi^n) = u_{N+1}^n + \frac{\Delta t \beta}{(\Delta x)^2} (2u_N^n - 2u_{N+1}^n + 2\Delta x \xi^n) \end{cases}$$



Algorithme d'Euler explicite avec les conditions aux limites de Neumann d'ordre 2

Pour $i = 0, \dots, N + 1$

$$u_i^0 = g_i$$

Fin Pour

Pour $n = 0, \dots, M$

Pour $i = 1, \dots, N$

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

Fin Pour

$$u_0^{n+1} = u_0^n + \frac{\Delta t \beta}{(\Delta x)^2} (2u_1^n - 2u_0^n - 2\Delta x h^n)$$

$$u_{N+1}^{n+1} = u_{N+1}^n + \frac{\Delta t \beta}{(\Delta x)^2} (2u_N^n - 2u_{N+1}^n + 2\Delta x \xi^n)$$

Fin Pour

6.4 Implementation des conditions aux limites de Neumann en utilisant la méthode de Crank-Nicolson

La méthode de Crank-Nicolson conduit à l'équation de la forme :

$$B_i u_{i-1}^{n+1} + D_i u_i^{n+1} + A_i u_{i+1}^{n+1} = K_i^n, \quad (47)$$

On réécrit l'équation sous la forme de système linéaire :

$$i = 1, \quad B_1 u_0^{n+1} + D_1 u_1^{n+1} + A_1 u_2^{n+1} = K_1^n$$

$$i = 2, \quad B_2 u_1^{n+1} + D_2 u_2^{n+1} + A_2 u_3^{n+1} = K_2^n$$

.....

$$i = N, \quad B_N u_{N-1}^{n+1} + D_N u_N^{n+1} + A_N u_{N+1}^{n+1} = K_N^n$$

On utilise les conditions aux limites $u_0^{n+1} = u_1^{n+1} - \Delta x h^{n+1}$, $u_{N+1}^{n+1} = u_N^{n+1} + \Delta x \xi^{n+1}$ et on obtient la forme matricielle suivante:

$$\begin{pmatrix} B_1 + D_1 & A_1 & 0 & 0 & 0 \\ B_2 & D_2 & A_2 & 0 & 0 \\ 0 & B_3 & D_3 & A_3 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ B_{N-1} & D_{N-1} & A_{N-1} & 0 & 0 \\ 0 & B_N & D_N + A_N & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} u_1^{n+1} \\ u_2^{n+1} \\ u_3^{n+1} \\ \vdots \\ u_N^{n+1} \end{pmatrix} = \begin{pmatrix} K_1^n + \Delta x B_1 h^{n+1} \\ K_2^n \\ K_3^n \\ \vdots \\ K_{N-1}^n \\ K_N^n - \Delta x A_N \xi^{n+1} \end{pmatrix}$$

Or

$$M \cdot U^{n+1} = K^n$$

Notre but est de trouver $U^{n+1} = (u_N^{n+1}, u_{N-1}^{n+1}, u_{N-2}^{n+1}, \dots, u_1^{n+1})$, exprimées en termes des

$(u_N^n, u_{N-1}^n, u_{N-2}^n, \dots, u_1^n)$. On ne peut plus calculer directement u_i^{n+1} . Pour cela il faut inverser la matrice M car

$$U^{n+1} = M^{-1} K^n.$$

Pour obtenir la matrice inverse d'une **matrice tridiagonale** on applique l'Algorithme de Thomas.

Algorithme de Crank-Nicolson. Condition aux limites de Neumann.

Pour $i = 0 : N + 1$

$$u_i^0 = g_i$$

Fin Pour

$$A_i = -\frac{\Delta t \beta}{2(\Delta x)^2}$$

$$B_i = -\frac{\Delta t \beta}{2(\Delta x)^2}$$

$$D_i = 1 + \frac{\Delta t \beta}{(\Delta x)^2}$$

$$D_1 = D_1 + B_1, \quad D_N = D_N + A_N$$

Pour $n = 0 : M$

Pour $i = 1 : N$

$$K_i^n = (1 - \frac{\Delta t \beta}{(\Delta x)^2})u_i^n + \frac{\Delta t \beta}{2(\Delta x)^2}(u_{i+1}^n + u_{i-1}^n) + \frac{\Delta t}{2}(f_i^n + f_i^{n+1}) - \delta_{1i} B_1 \cdot h^{n+1} \cdot \Delta x - \delta_{Ni} A_N \cdot \xi^{n+1} \cdot \Delta x$$

Fin Pour

$$Detoile_1 = D_1, \quad Ketoile_1^n = K_1^n$$

Pour $i = 2 : N$

$$Detoile_i = D_i - \frac{B_i \cdot A_{i-1}}{Detoile_{i-1}}$$

$$Ketoile_i^n = K_i^n - \frac{B_i \cdot Ketoile_{i-1}^n}{Detoile_{i-1}}$$

Fin Pour

$$u_N^{n+1} = \frac{Ketoile_N^n}{Detoile_N}$$

Pour $i = N - 1 : -1 : 1$

$$u_i^{n+1} = \frac{Ketoile_i^n - A_i \cdot u_{i+1}^{n+1}}{Detoile_i}$$

Fin Pour

$$u_0^{n+1} = u_1^{n+1} - \Delta x \cdot h^{n+1}$$

$$u_{N+1}^{n+1} = u_N^{n+1} + \Delta x \cdot \xi^{n+1}$$

Fin Pour

Algorithme de Crank-Nicolson pour MATLAB. Condition aux limites de Neumann.

Pour $i = 1 : N + 2$

$$u(1, i) = g(x(i))$$

Fin Pour

$$A(i) = -\frac{\Delta t \beta}{2(\Delta x)^2}$$

$$B(i) = -\frac{\Delta t \beta}{2(\Delta x)^2}$$

$$D(i) = 1 + \frac{\Delta t \beta}{(\Delta x)^2}$$

$$D(2) = D(2) + B(2), \quad D(N + 1) = D(N + 1) + A(N + 1)$$

Pour $n = 1 : M + 1$

Pour $i = 2 : N + 1$

$$K(n, i) = (1 - \frac{\Delta t \beta}{(\Delta x)^2})u(n, i) + \frac{\Delta t \beta}{2(\Delta x)^2}(u(n, i + 1) + u(n, i - 1)) + \frac{\Delta t}{2}(f(t(n), x(i)) + f(t(n + 1), x(i))) - \delta_{2,i} \cdot B(2) \cdot h(t(n + 1)) \cdot \Delta x - \delta_{N+1,i} \cdot A(N + 1) \cdot \xi(t(n + 1)) \cdot \Delta x$$

Fin Pour

$$Detoile(2) = D(2), \quad Ketoile(n, 2) = K(n, 2)$$

Pour $i = 3 : N + 1$

$$Detoile(i) = D(i) - \frac{B(i) \cdot A(i-1)}{Detoile(i-1)}$$

$$Ketoile(n, i) = K(n, i) - \frac{B(i) \cdot Ketoile(n, i-1)}{Detoile(i-1)}$$

Fin Pour

$$u(n + 1, N + 1) = \frac{Ketoile(n, N+1)}{Detoile(N+1)}$$

Pour $i = N : -1 : 2$

$$u(n + 1, i) = \frac{Ketoile(n, i) - A(i) \cdot u(n + 1, i + 1)}{Detoile(i)}$$

Fin Pour

$$u(n + 1, 1) = u(n + 1, 2) - \Delta x \cdot h(t(n + 1))$$

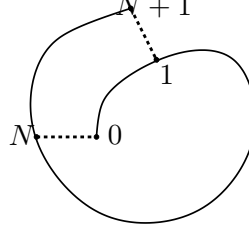
$$u(n + 1, N + 2) = u(n + 1, N + 1) + \Delta x \cdot \xi(t(n + 1))$$

Fin Pour

6.5 Conditions aux limites périodiques

La condition aux limites périodiques s'écrit:

$$u(x_0, t) = u(x_{N+1}, t).$$



Finalement on écrit les conditions de la forme

$$\begin{cases} u_0^n = u_N^n \\ u_1^n = u_{N+1}^n \end{cases}$$

On écrit les conditions initiales et aux limites avec deux boucles séparées, puis on construit deux boucles telles que l'une est à l'intérieur de l'autre:

Algorithme d'Euler explicite avec les conditions aux limites périodiques

```

Pour     $i = 0, \dots, N + 1$ 
 $u_i^0 = g_i$ 
Fin Pour

Pour     $n = 0 \dots M$ 
  Pour     $i = 1 \dots N$ 
 $u_i^{n+1} = u_i^n + \frac{\Delta t \beta}{(\Delta x)^2} (u_{i+1}^n - 2u_i^n + u_{i-1}^n)$ 
  Fin Pour
 $u_0^{n+1} = u_N^{n+1}$ 
 $u_{N+1}^{n+1} = u_1^{n+1}$ 
Fin Pour

```

En effet, on tourne l'algorithme à la main:

$n = 0$ toutes les valeurs u_i^0 sont connues de la condition initiale: $u_i^0 = g_i$ ($0 \leq i \leq N + 1$).

$$\begin{cases} i = 1 & u_1^1 = u_1^0 + \frac{\Delta t \beta}{(\Delta x)^2} (u_2^0 - 2u_1^0 + u_0^0) \\ i = 2 & u_2^1 = u_2^0 + \frac{\Delta t \beta}{(\Delta x)^2} (u_3^0 - 2u_2^0 + u_1^0) \\ \dots & \dots \\ i = N & u_N^1 = u_N^0 + \frac{\Delta t \beta}{(\Delta x)^2} (u_{N+1}^0 - 2u_N^0 + u_{N-1}^0) \end{cases}$$

$$u_0^1 = u_N^1$$

$$u_{N+1}^1 = u_1^1$$

$n = 1$ les valeurs u_i^1 ($i=0, \dots, N$) sont calculées à l'iteration $n = 0$, u_0^1 et u_{N+1}^1 sont connues de conditions aux limites périodiques: $u_0^1 = u_N^1$, $u_{N+1}^1 = u_1^1$.

$$\left\{ \begin{array}{ll} i = 1 & u_1^2 = u_1^1 + \frac{\Delta t \beta}{(\Delta x)^2} (u_2^1 - 2u_1^1 + u_0^1) \\ i = 2 & u_2^2 = u_2^1 + \frac{\Delta t \beta}{(\Delta x)^2} (u_3^1 - 2u_2^1 + u_1^1) \\ & \dots \\ i = N & u_N^2 = u_N^1 + \frac{\Delta t \beta}{(\Delta x)^2} (\boxed{u_{N+1}^1} - 2u_N^1 + u_{N-1}^1) \end{array} \right.$$

$$u_0^2 = u_N^2$$

$$u_{N+1}^2 = u_1^2$$

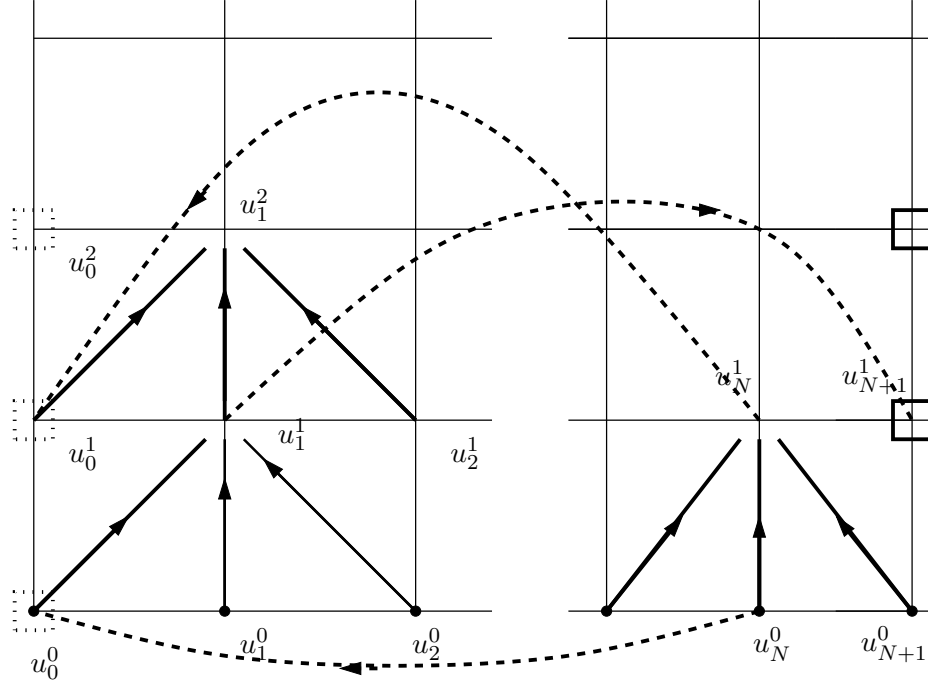
$$n = M$$

$$\left\{ \begin{array}{ll} i = 1 & u_1^M = u_1^{M-1} + \frac{\Delta t \beta}{(\Delta x)^2} (u_2^{M-1} - 2u_1^{M-1} + u_0^{M-1}) \\ i = 2 & u_2^M = u_2^{M-1} + \frac{\Delta t \beta}{(\Delta x)^2} (u_3^{M-1} - 2u_2^{M-1} + u_1^{M-1}) \\ & \dots \\ i = N & u_N^M = u_N^{M-1} + \frac{\Delta t \beta}{(\Delta x)^2} (\boxed{u_{N+1}^{M-1}} - 2u_N^{M-1} + u_{N-1}^{M-1}) \end{array} \right.$$

$$u_0^M = u_N^M$$

$$u_{N+1}^M = u_1^M$$

On peut visualiser cette explication par un dessin suivant:



7 Schémas aux plusieurs niveaux temporels

7.1 Schéma de Leap-Frog

Considérons la méthode de Leap-Frog:

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

La dérivée temporelle centrée et la dérivée spatiale centrée conduit au schéma complètement symétrique par rapport à n et i . Ce schéma est incapable de calculer des solutions approchées car il est instable. On explique la notion de la stabilité plus tard.

Cependant le méthode de Leap-Frog s'utilise souvent pour la resolution des équations hyperboliques et les équations non linéaires du type de Korteweg-de Vries. Expliquons l'application de méthode de Leap-Frog en prenant comme exemple l'équation de chaleur.

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

On commence par $n = 1$. Comment calculer u_i^1 ? On fait le premier pas avec le schéma d'Euler implicite, et ensuite on continue en Leap-Frog.

En effet, le schéma d'Euler explicite donne:

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

$$n = 0 \quad \frac{u_i^1 - u_i^0}{\Delta t} = \beta \frac{u_{i+1}^0 - 2u_i^0 + u_{i-1}^0}{(\Delta x)^2},$$

$$u_i^1 = u_i^0 - \frac{\Delta t \beta}{(\Delta x)^2} (u_{i+1}^0 - 2u_i^0 + u_{i-1}^0).$$

On continue en Leap-Frog:

$n = 1$ les valeurs u_i^1 ($i=1, \dots, N$) sont calculées par la méthode d'Euler, u_0^1 et u_{N+1}^1 sont connues de conditions aux limites: $u_0^1 = h^1$, $u_{N+1}^1 = \xi^1$.

$$\left\{ \begin{array}{ll} i = 1 & u_1^2 = u_1^0 + \frac{2\Delta t \beta}{(\Delta x)^2} (u_2^1 - 2u_1^1 + u_0^1) \\ i = 2 & u_2^2 = u_2^0 + \frac{2\Delta t \beta}{(\Delta x)^2} (u_3^1 - 2u_2^1 + u_1^1) \\ \vdots & \vdots \\ i = N & u_N^2 = u_N^0 + \frac{2\Delta t \beta}{(\Delta x)^2} (u_{N+1}^1 - 2u_N^1 + u_{N-1}^1) \end{array} \right.$$

$n = M$

$$\left\{ \begin{array}{ll} i = 1 & u_1^M = u_1^{M-2} + \frac{2\Delta t \beta}{(\Delta x)^2} (u_2^{M-1} - 2u_1^{M-1} + u_0^{M-1}) \\ i = 2 & u_2^M = u_2^{M-2} + \frac{2\Delta t \beta}{(\Delta x)^2} (u_3^{M-1} - 2u_2^{M-1} + u_1^{M-1}) \\ \vdots & \vdots \\ i = N & u_N^M = u_N^{M-2} + \frac{2\Delta t \beta}{(\Delta x)^2} (u_{N+1}^{M-1} - 2u_N^{M-1} + u_{N-1}^{M-1}) \end{array} \right.$$

7.2 Schéma de Lax-Wendroff

On commence par la decomposition de la fonction cherchée en series de Taylor d'ordre 2:

$$u_i^{n+1} = u_i^n + \frac{\partial u}{\partial t} \Delta t + \frac{1}{2} \frac{\partial^2 u}{\partial t^2} (\Delta t)^2 + \dots$$

On remplace $\frac{\partial u}{\partial t}$ par $\beta \frac{\partial^2 u}{\partial x^2}$, $\frac{\partial^2 u}{\partial t^2}$ par $\beta^2 \frac{\partial^4 u}{\partial x^4}$.

On discrétise les dérivées spatiales:

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

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

On obtient le schéma explicite:

$$u_i^{n+1} = u_i^n + \beta \Delta t \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2} + \frac{\beta^2 (\Delta t)^2}{2} \frac{u_{i+2}^n - 4u_{i+1}^n + 6u_i^n - 4u_{i-1}^n + u_{i-2}^n}{(\Delta x)^4}$$

8 Résumé sur les schémas de discrétisation.

- Euler explicite: Forward in time, centered in space:

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

- Euler implicite: Backward in time, centered in space:

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

- Crank-Nicolson:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{\beta}{(\Delta x)^2} \left[\frac{1}{2}(u_{i+1}^n + u_{i+1}^{n+1}) - 2\frac{1}{2}(u_i^n + u_i^{n+1}) + \frac{1}{2}(u_{i-1}^n + u_{i-1}^{n+1}) \right]$$

- Lax-Friedrich:

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

- Lax-Wendroff:

$$u_i^{n+1} = u_i^n + \beta \Delta t \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2} + \frac{\beta^2 (\Delta t)^2}{2} \frac{u_{i+2}^n - 4u_i^n + 6u_{i-1}^n - 4u_{i-2}^n + u_{i-3}^n}{(\Delta x)^4}$$

- Leap-Frog:

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

- Du Fort-Frankel

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

- Gear:

$$\frac{3u_i^{n+1} - 4u_i^n + u_i^{n-1}}{2\Delta t} = \beta \frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}}{(\Delta x)^2}$$

9 Consistance

Une représentation aux différences finies sera dite consistante si l'on peut montrer que l'erreur de troncature tend vers zéro lorsque les pas du maillage tendent vers zéro. En d'autres termes la consistance exprime que les équations discrétisées doivent tendre vers les équations aux dérivées partielles originales lorsque les pas du maillage tendent vers zéro.

Définition de Consistance

Soit l'équation aux dérivées partielles

$$F(u, \frac{\partial u}{\partial t}, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}, \frac{\partial^2 u}{\partial t^2}, \dots) = 0.$$

Le schéma aux différences finies est dit consistant avec l'équation aux dérivées partielles $F(\cdot) = 0$ si pour toute solution suffisamment régulière, l'erreur de troncature du schéma, défini par l'équation discrete

$$F(u(t^n + m\Delta t, x_i + j\Delta x)) = 0,$$

tend vers zéros, uniformément par rapport à (t, x) , lorsque Δt et Δx tendent vers zéros indépendamment.

On dit que le schéma est précis à l'ordre p en espace et à l'ordre q en temps si l'erreur de troncature tend vers zéro comme $O((\Delta x)^p + (\Delta t)^q)$ lorsque Δt et Δx tendent vers zéro.

Un exemple de l'équation discrete $F(u(t^n + m\Delta t, x_i + j)) = 0$, de cette définition est

$$F(\frac{u_i^{n+1} - u_i^n}{\Delta t}, \frac{u_{i+1}^n - u_i^n}{\Delta x}, \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2}, \frac{u_i^{n+1} - 2u_i^n + u_i^{n-1}}{(\Delta t)^2}, \dots) = 0$$

Concrètement on calcule l'erreur de troncature d'un schéma en remplaçant u_{i+j}^{n+m} faisant partie de l'équation discrete associée à ce schéma par $u(t^n + m\Delta t, x_i + j\Delta x)$.

9.1 Consistance de la méthode d'Euler progressif

On vérifie consistance à l'équation de la chaleur et on calcule l'erreur de troncature du schéma suivant:

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

On effectue le remplacement:

$$\frac{u(t^n + \Delta t, x_i) - u(t^n, x_i)}{\Delta t} - \beta \frac{u(t^n, x_i + \Delta x) - 2u(t^n, x_i) + u(t^n, x_i - \Delta x)}{(\Delta x)^2} = 0$$

On utilise la formule de Taylor et on obtient:

$$\frac{\partial u}{\partial t}(t^n, x_i) - \beta \frac{\partial^2 u}{\partial x^2}(t^n, x_i) - \frac{\beta}{12} \frac{\partial^4 u}{\partial x^4}(\Delta x)^2 + \frac{1}{2!} \frac{\partial^2 u}{\partial t^2}(\Delta t) + O((\Delta t)^2) + O((\Delta x)^4) = 0$$

Si $u(t^n, x_i)$ est la solution de l'équation de chaleur ($u_t = \beta u_{xx}$), on obtient aisément la consistance ainsi que la précision à l'ordre 1 en temps et 2 en espace. En effet

$$\begin{aligned} & \frac{\partial u}{\partial t} - \beta \frac{\partial^2 u}{\partial x^2} - \frac{\beta}{12} \frac{\partial^4 u}{\partial x^4}(\Delta x)^2 + \frac{1}{2!} \frac{\partial^2 u}{\partial t^2}(\Delta t) = \\ & = -\frac{\beta}{12} \frac{\partial^4 u}{\partial x^4}(\Delta x)^2 + \frac{1}{2!} \frac{\partial^2 u}{\partial t^2}(\Delta t) \rightarrow 0, \quad \Delta x, \Delta t \rightarrow 0 \end{aligned}$$

Il est possible d'augmenter l'ordre de précision. En effet

$$\frac{\partial}{\partial t} \frac{\partial u}{\partial t} = \beta \frac{\partial}{\partial t} \frac{\partial^2 u}{\partial x^2} = \beta \frac{\partial^2}{\partial x^2} \frac{\partial u}{\partial t} = \beta^2 \frac{\partial^4 u}{\partial x^4}$$

On réécrit l'erreur de troncature

$$T = \left(\frac{\beta^2 \Delta t}{2} - \frac{\beta (\Delta x)^2}{12} \right) \frac{\partial^4 u}{\partial x^4} + O((\Delta t)^2, (\Delta x)^4).$$

Si on choisit

$$\Delta t = \frac{(\Delta x)^2}{6\beta}$$

alors ce schéma est précis à l'ordre 2 en temps et 4 en espace. On a augmenté la précision sans changer la complexité du calcul.

9.2 Consistance du schéma de DuFort-Frankel

On calcule l'erreur de troncature du schéma:

$$\begin{aligned} & \frac{u_i^{n+1} - u_i^{n-1}}{2\Delta t} - \frac{\beta}{(\Delta x)^2} (u_{i+1}^n - u_i^{n-1} - u_i^{n+1} + u_{i-1}^n) = 0 \\ & \frac{u(t^n + \Delta t, x_i) - u(t^n - \Delta t, x_i)}{\Delta t} - \beta \frac{u(t^n, x_i + \Delta x) - u(t^n + \Delta t, x_i) - u(t^n - \Delta t, x_i) + u(t^n, x_i - \Delta x)}{(\Delta x)^2} = \\ & = \frac{\partial u}{\partial t} - \beta \frac{\partial^2 u}{\partial x^2} + \frac{1}{12} \frac{\partial^3 u}{\partial t^3} (\Delta t)^2 + \beta \frac{\partial^2 u}{\partial t^2} \left(\frac{\Delta t}{\Delta x} \right)^2 - \frac{\beta}{12} \frac{\partial^4 u}{\partial x^4} (\Delta x)^2 \end{aligned}$$

Tout se passe bien si le rapport $(\frac{\Delta t}{\Delta x})^2$ tend vers zéro. Le schéma est d'ordre $O((\frac{\Delta t}{\Delta x})^2, (\Delta x)^2)$.

Par contre si, lorsque Δt et Δx tendent vers zéro en conservant le rapport $(\frac{\Delta t}{\Delta x})^2 = \alpha^2$ constant, l'équation aux différence finies étudiées sera consistante avec une autre équation, qui sera d'ailleurs de type hyperbolique:

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

10 Stabilité

L'instabilité se manifeste par des oscillations non bornées de la solution numérique.

Définissons une norme pour une solution numérique $u(t^n, x_i)$. Nous reprenons les normes classiques sur R^N que nous pondérons simplement par le pas Δx . La norme L^p :

$$\|u^n\|_p = \left(\sum_{i=0}^N \Delta x |u_i^n|^p \right)^{1/p}$$

où le cas limite $p = \infty$ doit être compris dans le sens

$$\|u^n\|_\infty = \max_{1 \leq j \leq N} |u_i^n|$$

Nous allons travailler dans l'espace discrete muni de la norme L^2 :

$$\|u^n\|_2 = \sqrt{\sum_{i=0}^N \Delta x |u_i^n|^2}$$

Définition de Stabilité

Un schéma aux différences finies est dit stable pour la norme L^2 s'il existe une constante $K > 0$ indépendante de Δx et Δt (lorsque ces valeurs tendent vers zéro) telle que

$$\|u^n\| \leq K \|u^0\| \text{ pour tout } n \geq 0,$$

quelle que soit la donnée initiale u^0 .

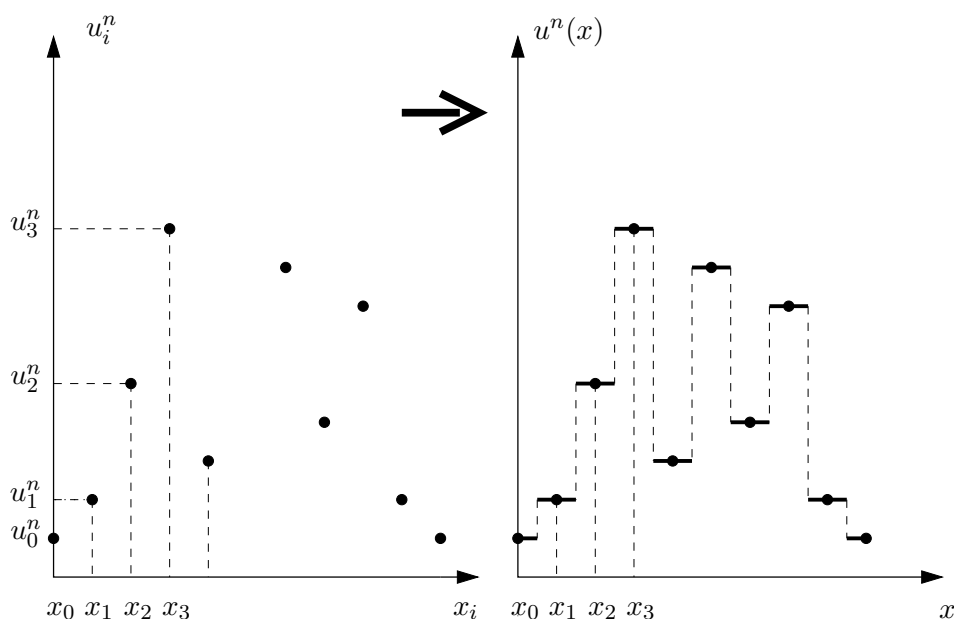
Si l'inégalité n'a lieu que pour des pas Δt et Δx restreints à une inégalité, on dit que le schéma est conditionnellement stable.

On va étudier la stabilité en norme L^2 . La norme L^2 se prête très bien à l'étude de la stabilité grâce à l'outil très puissant de l'analyse de Fourier. Nous supposons désormais que les conditions aux limites sont les conditions aux limites périodiques:

$$u(t, x + 1) = u(t, x)$$

pour $\forall x \in [0, 1]$. A chaque vecteur u_i^n ($0 \leq i \leq N$) on associe une fonction $u^n(x)$, constante par morceaux, périodique de période 1, définie sur $[0, 1]$ par

$$u^n(x) = u_i^n \text{ si } x_{i-1/2} < x < x_{i+1/2}.$$



D'après l'analyse de Fourier, toute fonction de $L^2[0, 1]$ peut se decomposer en une somme de Fourier:

$$u^n(x) = \sum_{k \in \mathbb{Z}} \hat{u}^n(k) e^{2i\pi kx} \quad \hat{u}^n(k) = \int_0^1 u^n(x) e^{-2i\pi kx} dx.$$

On a aussi la formule de Plancherel

$$\int_0^1 |u^n(x)|^2 dx = \sum_{k \in \mathbb{Z}} |\hat{u}^n(k)|^2$$

Expliquons maintenant la méthode sur l'exemple du schéma explicite d'Euler:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \beta \frac{u^n(x + \Delta x) - 2u^n(x) + u^n(x - \Delta x)}{(\Delta x)^2}$$

On applique la transformation de Fourier et on obtient

$$\hat{u}^{n+1}(k) = (1 - \frac{\beta \Delta t}{(\Delta x)^2} (-e^{2i\pi k \Delta x} + 2 - e^{-2i\pi k \Delta x})) \hat{u}^n(k)$$

On peut réécrire cette expression de la forme:

$$\hat{u}^{n+1}(k) = G(k) \hat{u}^n(k) = (G(k))^{n+1} \hat{u}^0(k) \quad \text{avec} \quad G(k) = 1 - \frac{4\beta \Delta t}{(\Delta x)^2} (\sin(\pi k \Delta x))^2$$

Ce facteur G s'appelle facteur d'amplification. Pour que la solution soit stable il faut que G ne soit pas supérieur à 1 pour tous les modes de Fourier de la solution u^n . En effet le coefficient de Fourier $\hat{u}^{n+1}(k)$ est borné par la condition initiale $|\hat{u}^{n+1}(k)| \leq |\hat{u}^0(k)|$ si le facteur d'amplification $|G(k)| \leq 1$. En effet

$$|\hat{u}^{n+1}(k)| = |(G(k))^{n+1} \hat{u}^0(k)| \leq |\hat{u}^0(k)|$$

à condition que $|G(k)| \leq 1$. Le facteur d'amplification $|G(k)| \leq 1$ si

$$|1 - \frac{4\beta \Delta t}{(\Delta x)^2} (\sin(\pi k \Delta x))^2| \leq 1 \implies 2\beta \Delta t (\sin(\pi k \Delta x))^2 \leq (\Delta x)^2 \implies 2\beta \Delta t \leq (\Delta x)^2$$

Si la condition $2\beta \Delta t \leq (\Delta x)^2$ est satisfaite on déduit de la formule de Plancherel:

$$\|u^n\|_2^2 = \int_0^1 |u^n(x)|^2 dx = \sum_{k \in \mathbb{Z}} |\hat{u}^n(k)|^2 \leq \sum_{k \in \mathbb{Z}} |\hat{u}^0(k)|^2 = \int_0^1 |u^0(x)|^2 dx = \|u^0\|_2^2$$

Donc

$$\|u^n\|_2 \leq \|u^0\|_2$$

ce qui n'est rien d'autre que la stabilité L^2 .

Un schéma instable est totalement inutilisable. En effet les données initiales peuvent entraîner les fluctuations très fortes. Même si on part d'une donnée initiale spécialement préparée de manière à ce qu'aucun des modes de Fourier instable ne soit pas excité par elle, les inévitables erreurs d'arrondi vont créer des composantes non nulles de la solution sur ces modes instables. La croissance exponentielle des modes instables entraîne qu'après quelques pas en temps ces petits modes deviennent énormes et polluent complètement le reste de la solution numérique.

10.1 Stabilité d'un point du vue pratique

1) On injecte dans le schéma des différences finies un mode de Fourier ou un ansatz sous la forme d'une onde plane:

$$u_i^n = G^n(k) e^{2i\pi k x_i}, \quad x_i = i\Delta x$$

2) On analyse la valeur du facteur d'implification $G^n(k)$.

3) On applique la condition de stabilité de Von Neumann:

$$|G(k)| \leq 1 \text{ pour tout mode } k \in \mathbb{Z}$$

Si la condition de stabilité de Von Neumann est satisfaite (avec éventuellement des restrictions sur Δt et Δx) alors le schéma est stable pour la norme L^2 .

10.2 Stabilité de méthode de Crank-Nicolson

On injecte dans l'équation discrétisée

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{\beta}{(\Delta x)^2} \left[\frac{1}{2}(u_{i+1}^n + u_{i+1}^{n+1}) - 2\frac{1}{2}(u_i^n + u_i^{n+1}) + \frac{1}{2}(u_{i-1}^n + u_{i-1}^{n+1}) \right]$$

une onde plane: $u_i^n = G^n(k)e^{2i\pi k x_i}$. On obtient:

$$\frac{G^{n+1}(k)e^{2i\pi k x_i} - G^n(k)e^{2i\pi k x_i}}{\Delta t} = \frac{\beta}{(\Delta x)^2} \left(\frac{1}{2}(G^n(k)e^{2i\pi k(x_i+\Delta x)} + G^{n+1}(k)e^{2i\pi k(x_i+\Delta x)}) - \right. \\ \left. G^n(k)e^{2i\pi k x_i} - G^{n+1}(k)e^{2i\pi k x_i} + \frac{1}{2}(G^n(k)e^{2i\pi k(x_i-\Delta x)} + G^{n+1}(k)e^{2i\pi k(x_i-\Delta x)}) \right)$$

Le facteur d'amplification $G(k)$ est donné par

$$G(k) = \frac{1 - \frac{2\beta\Delta t}{(\Delta x)^2} \sin^2(\pi k \Delta x)}{1 + \frac{2\beta\Delta t}{(\Delta x)^2} \sin^2(\pi k \Delta x)}$$

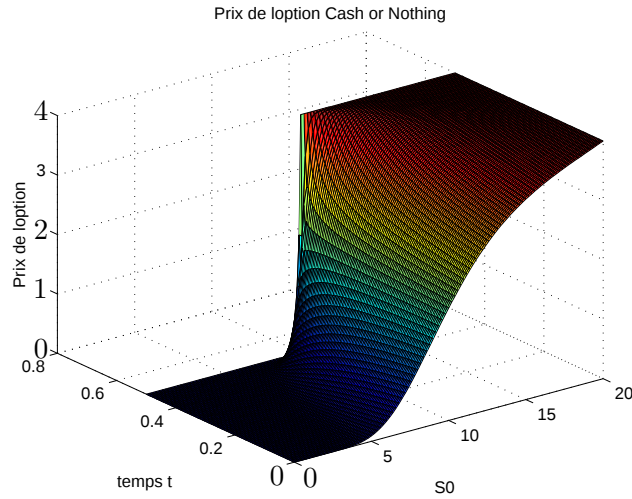
Pour toute valeur de k , on a $|G(k)| \leq 1$, ce qui donne une méthode qui est inconditionnellement convergente. La méthode de Crank- Nicolson est une amélioration significative par rapport à la méthode d'Euler. Elle est remarquable si le front ou la queue de condition initiale ne présente pas de discontinuité, mais elle génère des oscillations rapides si les discontinuités dans les conditions initiales sont présentes.

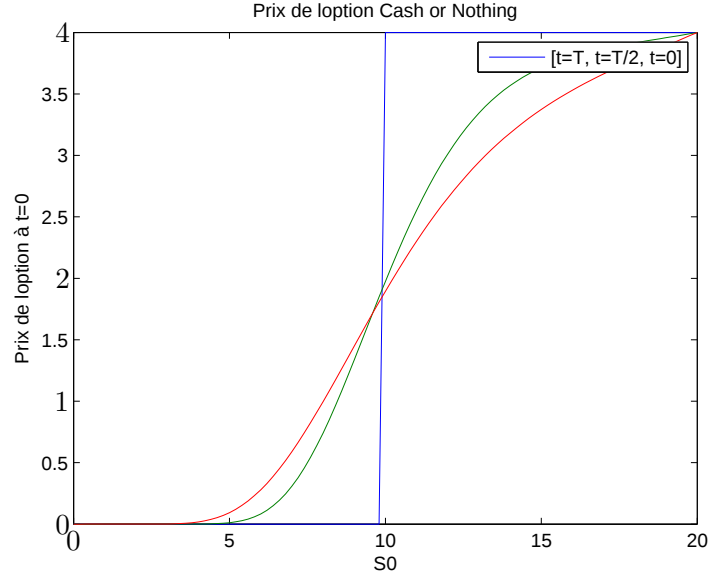
En effet, sur le graphe à 3 dimensions on voit ces fluctuations. Ils interviennent au domaine de discontinuité de conditions finales de l'option "Cash or nothing" avec fonction payoff :

$$V(t = T, S) = 4Heaviside(S - K).$$

Sur le graphe à 2 dimensions on voit les solutions $V(S_t, t)$ pour $t = 0$, $t = \frac{T}{2}$, $t = T$ (fonction payoff). La surface - solution est la solution de l'équation de Black et Scholes avec la condition finale et les conditions aux limites :

$$\begin{cases} \frac{\partial V}{\partial t} - rV + rS \frac{\partial V}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} = 0 \\ V(t = T, S) = 4Heaviside(S - K) \\ V(t, S = 0) = 0 \\ V(t, S \rightarrow \infty) = 4 \end{cases} \quad (48)$$





Remarque: L'analyse de Fourier repose sur le choix des conditions aux limites périodiques. La stabilité L^2 peut être aussi démontrée dans le cas de conditions aux limites de Dirichlet. Il suffit d'effectuer la prolongation de la fonction $u^n(x)$ sur l'intervalle $[-1, 0]$ de façon paire et puis appliquer la décomposition en série de Fourier de la fonction périodique sur $[-1, 1]$.

11 Convergence

La solution numérique u_i^n doit tendre vers la solution exacte $u(i\Delta x, n\Delta t)$ de l'équation différentielle en tout point x_i et en tout point t_n quand Δx et Δt tendent vers zéro.

Définition de Convergence

Si l'erreur entre la solution numérique et la solution exacte satisfasse la condition suivante

$$\forall T > 0, \quad \lim_{\Delta x, \Delta t \rightarrow 0} \left(\sup_{t_n \leq T} \|u(i\Delta x, n\Delta t) - u_i^n\| \right) = 0$$

le schéma est convergent.

11.1 Théorème de Lax

Théorème de Lax dit:

Soit $u(x, t)$ la solution suffisamment régulière de l'équation de la chaleur.

Soit u_i^n la solution de l'équation discrète obtenue par un schéma aux différences finies avec la condition initiale

$$u_i^0 = g(x_i).$$

On suppose que le schéma est linéaire, à deux niveaux, consistant, et stable pour la norme $\|\cdot\|$. Alors le schéma est convergent au sens où

$$\forall T > 0, \quad \lim_{\Delta x, \Delta t \rightarrow 0} \left(\sup_{t_n \leq T} \|u(i\Delta x, n\Delta t) - u_i^n\| \right) = 0$$

De plus, si le schéma est précis à l'ordre p en espace et à l'ordre q en temps, alors pour tout temps $T > 0$ il existe une constante $C_T > 0$ telle que

$$\sup_{t_n \leq T} \|u(i\Delta x, n\Delta t) - u_i^n\| \leq C_T((\Delta x)^p + (\Delta t)^q)$$

On peut dire plus simplement:

Pour un schéma linéaire consistant et stabilité implique convergence.

Démonstration du théorème de Lax

Un schéma linéaire à deux niveaux peut s'écrire sous la forme condensée:

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

où M est la matrice d'itération (carré de taille N).

En effet l'équation de la chaleur discrete avec les conditions aux limites de Dirichlet nuls admet la forme:

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

soit

$$u_i^{n+1} = au_{i-1}^n + u_i^n(1 - 2a) + au_{i+1}^n = 0, \quad a = \frac{\beta \Delta t}{(\Delta x)^2}$$

et s'écrit sous la forme matricielle

$$\begin{pmatrix} u_1^{n+1} \\ u_2^{n+1} \\ u_3^{n+1} \\ \vdots \\ u_N^{n+1} \end{pmatrix} = \begin{pmatrix} 1-2a & a & 0 & 0 & 0 \\ a & 1-2a & a & 0 & 0 \\ & & & & \\ & & a & 1-2a & a \\ 0 & C & & 1-2a & \end{pmatrix} \cdot \begin{pmatrix} u_1^n \\ u_2^n \\ u_3^n \\ \vdots \\ u_N^n \end{pmatrix}.$$

Soit $U_i^n \equiv u(t^n, x_i)$ la solution analytique de l'équation de la chaleur:

$$\frac{\partial U}{\partial t} = \beta \frac{\partial^2 U}{\partial x^2}.$$

Comme le schéma est consistant, il existe un vecteur ϵ^n tel que

$$U^{n+1} = MU^n + \Delta t \epsilon^n, \quad \text{avec} \quad \lim_{\Delta x, \Delta t \rightarrow 0} \|\epsilon^n\| = 0 \quad (2)$$

et la convergence de l'erreur de troncature ϵ^n est uniforme pour tout les temps $0 \leq t_n \leq T$. Si le schéma est précis à l'ordre p en espace et à l'ordre q en temps alors l'erreur de troncature

$$\|\epsilon^n\| \leq C((\Delta x)^p + (\Delta t)^q)$$

On pose

$$\epsilon_i^n = u_i^n - U_i^n$$

et on obtient par soustraction le (2) du (1):

$$\epsilon^{n+1} = M\epsilon^n - \Delta t \epsilon^n$$

d'où par récurrence

$$\epsilon^n = M^n \epsilon^0 - \Delta t \sum_{k=1}^n M^{n-k} \epsilon^{k-1}$$

Or, la stabilité du schéma veut dire que

$$\|u^n\| = M^n \|u^0\| \leq K \|u^0\|$$

pour toute donnée initiale, c'est-à-dire $\|M^n\| \leq K$ où la constante K ne dépend pas de n ; D'autre part $\epsilon^0 = 0$ donc on obtient

$$\epsilon^n \leq \Delta t \sum_{k=1}^n \|M^{n-k}\| \cdot \|\epsilon^{k-1}\| \leq \Delta t \cdot n \cdot K \cdot C \cdot ((\Delta x)^p + (\Delta t)^q)$$

ce qu'il fallait démontrer si $C_T = TKC$.

D'un point de vue pratique, le théorème de Lax est très rassurant: si l'on utilise un schéma consistant et que l'on n'observe pas d'oscillations numérique (c'est-à-dire qu'il est stable), alors la solution numérique est proche de la solution exacte (le schéma converge).

12 Propagation des erreurs.

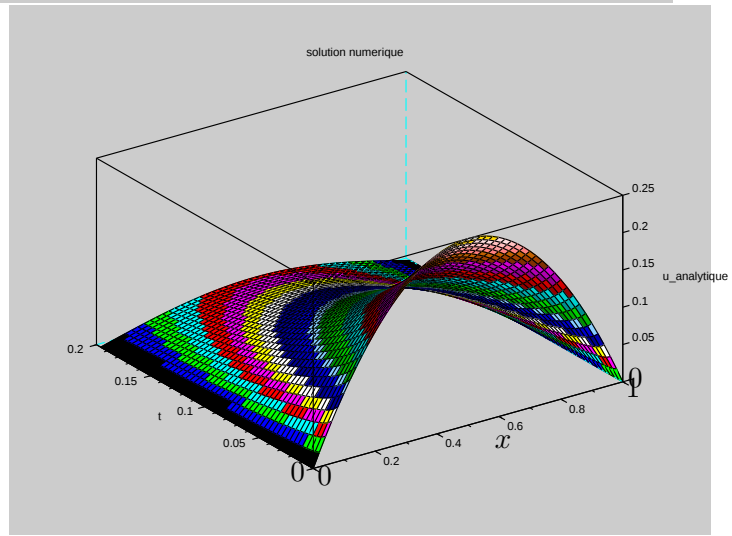
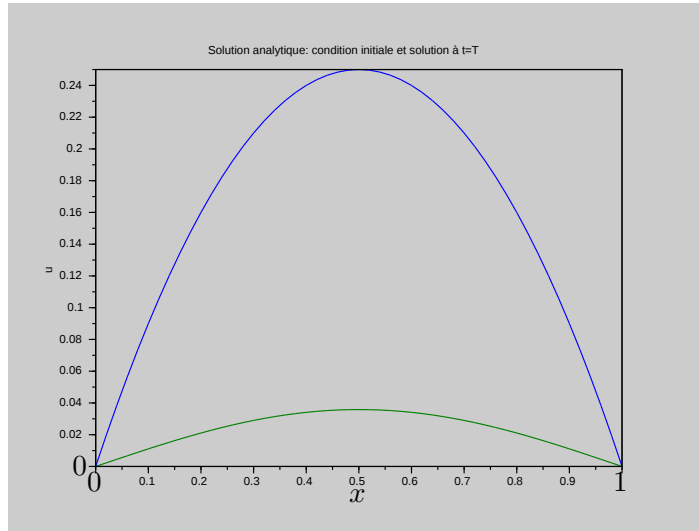
Dans cette section nous présentons les résultats des expériences numériques. Nous comparons les solutions numériques obtenus par les schémas d'Euler explicite et Crank-Nicolson avec la solution analytique. Considérons l'équation de la chaleur sur $[0, 1]$ avec la condition initiales $u(t=0, x) = x(1-x)$

$$\begin{cases} \frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} \\ u(t=0, x) = x(1-x) \\ u(t, x=0) = 0 \\ u(t, x=1) = 0 \end{cases} \quad (49)$$

La solution analytique de cette équation s'écrit de la forme:

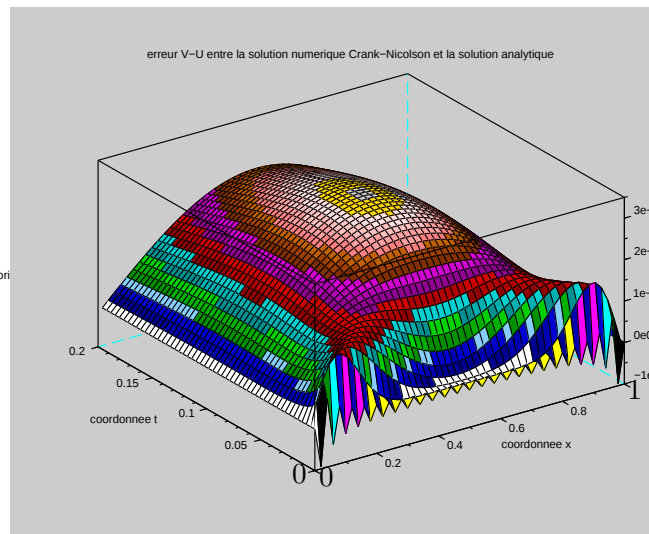
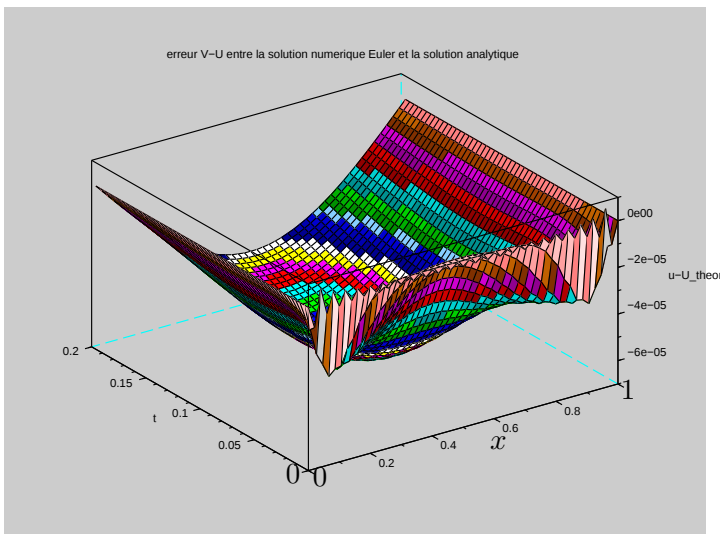
$$u(t, x) = \sum \frac{4(1 - (-1)^n)}{(\pi n)^3} \sin(\pi n x) e^{-\pi^2 n^2 t}$$

On trace la solution exacte en 2 et 3 dimensions:



Choisissons la discrétisation suivantes de l'espace temps:

$$\begin{cases} T = 0.2 & M + 1 = 1000 & \Delta t = T/(M + 1) = 0.0002 \\ L = 1 & N + 1 = 50 & \Delta x = L/(N + 1) = 0.02 \end{cases} \quad (50)$$



L'approximation d'Euler explicite est $(\Delta t) \sim 0.02\%$ $(\Delta x)^2 \sim 0.04$

L'approximation de Crank-Nicolson implicite est $(\Delta t)^2 \sim 4 \cdot 10^{-6}\%$ $(\Delta x)^2 \sim 0.04$

- Sur le graphe á gauche obtenu par Euler explicite l'erreur maximal est d'ordre $6 \cdot 10^{-5}$ ce que correspond á $(6 \cdot 10^{-5})/0.24 \sim 0.025\%$ et se trouve en accord avec l'approximation.
- Sur le graphe á gauche obtenu par Euler explicite l'erreur maximal est d'ordre $6 \cdot 10^{-5}$ ce que correspond á $(3 \cdot 10^{-6})/0.24 \sim 0.012\%$ et se trouve en accord avec l'approximation.

13 Analyse de Fourier des EDP. Relation de dispersion

Chaque fonction $u \in L^2$ peut étre décomposée en intégrale de Fourier:

$$u(x, t) = \frac{1}{2\pi} \int e^{ikx} \hat{u}(k, t) dk, \quad \hat{u}(k, t) = \int e^{-ikx} u(x, t) dx$$

Une composante de Fourier $v(x, t) = e^{i(kx+\omega t)}$ (l'onde plane) est une solution de chaque équation à condition qu'on impose une relation entre k et ω . Cette relation s'appelle la relation de dispersion. Le rapport $\frac{\omega}{k}$ s'appelle la vitesse de propagation de phase

$$v_\varphi = \frac{\omega}{k}$$

Considérons les cinq types des EDP les plus importantes:

- L'équation de chaleur:

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

- L'équation d'onde:

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0$$

- L'équation de Schrodinger

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

- L'équation de Burgers

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = 0$$

- L'équation de KdV linéarisée

$$\frac{\partial u}{\partial t} + u_0 \frac{\partial u}{\partial x} + b \frac{\partial^3 u}{\partial x^3} = 0$$

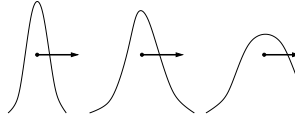
- L'équation de KdV

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + b \frac{\partial^3 u}{\partial x^3} = 0$$

- Injectons $v(x, t) = e^{i(kx+\omega t)}$ dans l'équation de chaleur. On obtient:

$$\omega = ik^2, \quad u(x, t) = e^{(ikx-k^2t)}$$

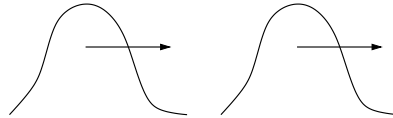
La solution diminue dans le temps. Donc l'équation est diffusive.



- Injectons $v(x, t) = e^{i(kx+\omega t)}$ dans l'équation d'onde. On obtient:

$$\omega = k, \quad u(x, t) = e^{ik(x-t)}$$

la vitesse de phase $v_\varphi = 1$ est une constante donc la solution se propage sans deformation.



- Injectons $v(x, t) = e^{i(kx+\omega t)}$ dans l'équation de Schrodinger. On obtient:

$$\omega = -k^2, \quad u(x, t) = e^{i(kx-k^2t)}$$

La solution n'est pas diffusive mais dispersive. C'est-à-dire la solution tend se briser aux paquets d'ondes qui se déplacent avec les vitesses différentes. En effet, la vitesse de phase $v_\varphi = \frac{\omega}{k} = -k$, dépend du vecteur d'onde k (ou longueur d'onde), chaque composante de Fourier de propagation avec les vitesses différentes, on dit que la solution se disperse.



- Injectons $v(x, t) = e^{i(kx+\omega t)}$ dans l'équation de Burgers. On obtient:

$$\omega + vk = 0, \quad v_{phase} = v$$

Cela signifie que la vitesse de propagation de phase dépend de l'amplitude.

L'équation de Burgers nonlinéaire décrit la propagation d'une vague. Le sommet d'une vague se propage avec la vitesse plus grande que celle du sol. Ce phénomène, la formation des choc, explique pourquoi les vagues tombent et se brisent.

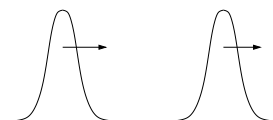


- Injectons $v(x, t) = e^{i(kx+\omega t)}$ dans l'équation de KdV linéarisée. On obtient:

$$\omega = k^3$$

La solution est dispersive.

• L'équation de KdV est une sorte de remarquable équilibre entre les phénomènes de formation des choc et de la dispersion. Les solutions de KdV - solitons sont très stables.



14 Equations hyperboliques

- Equation d'advection

$$\left\{ \begin{array}{l} \frac{\partial u}{\partial t} + V \frac{\partial u}{\partial x} = 0 \\ u(t=0, x) = g(x) \\ u(t, 0) = h(t), \quad \text{si } V > 0 \\ u(t, L) = \xi(t), \quad \text{si } V < 0 \end{array} \right. \quad (51)$$

Cette équation décrit la propagation d'une fluctuation sans deformation. La fonction $g(x - Vt)$ est une solution de l'équation.

Si $V > 0$ l'onde se propage à droite

Si $V < 0$ l'onde se propage à gauche.

C'est pourquoi on fixe deux conditions aux limites différentes.

- Equation d'ondes d'ordre 2

L'équation des ondes modélise des phénomènes de propagation d'ondes ou des vibrations. L'équation décrit des ondes se propageant à gauche et à droite.

$$\left\{ \begin{array}{l} \frac{\partial^2 u}{\partial t^2} = V^2 \frac{\partial^2 u}{\partial x^2} \\ u(t=0, x) = g(x) \\ u(0, t) = u(L, t) \\ \frac{\partial u}{\partial t}(t=0, x) = g_1(x) \end{array} \right. \quad (52)$$

14.1 Equation d'advection

On a l'équation:

$$\frac{\partial u}{\partial t} + V \frac{\partial u}{\partial x} = 0$$

14.1.1 Schéma upwind

- Discrétisation:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \left\{ \begin{array}{l} -V \frac{u_i^n - u_{i-1}^n}{\Delta x}, \quad V > 0 \\ u_0^n = g_i \\ u_0^n = h^n \end{array} \right. \quad \frac{u_i^{n+1} - u_i^n}{\Delta t} = \left\{ \begin{array}{l} -V \frac{u_{i+1}^n - u_i^n}{\Delta x}, \quad V < 0 \\ u_0^n = g_i \\ u_{N+1}^n = \xi^n \end{array} \right.$$

14.1.2 Implémentation du schéma explicite upwind

Pour implémenter le schéma upwind on a pas besoin une seule condition aux limites: en $x = 0$ si $V > 0$ et en $x = L$ si $V < 0$.

En effet, soit $V > 0$ et

$$C = \frac{V \Delta t}{\Delta x}$$

On réécrit le schéma de la forme

$$u_i^{n+1} = u_i^n(1 + C) + Cu_{i-1}^n$$

On écrit les conditions initiales et aux limites avec deux boucles séparées, puis on construit deux boucles telles que l'une est à l'intérieur de l'autre::

Algorithme du schéma explicite upwind

Pour $i = 0, \dots, N + 1$

$$u_i^0 = g_i$$

Fin Pour

Pour $n = 1, \dots, M + 1$

$$u_0^n = h^n$$

Fin Pour

Pour $n = 0, \dots, M$

Pour $i = 1, \dots, N$

$$u_i^{n+1} = u_i^n(1 + C) + Cu_{i-1}^n$$

Fin pour

Fin pour

En effet, on tourne l'algorithme à la main:

$n = 0$ toutes les valeurs u_i^0 sont connues de la condition initiale: $u_i^0 = g_i$

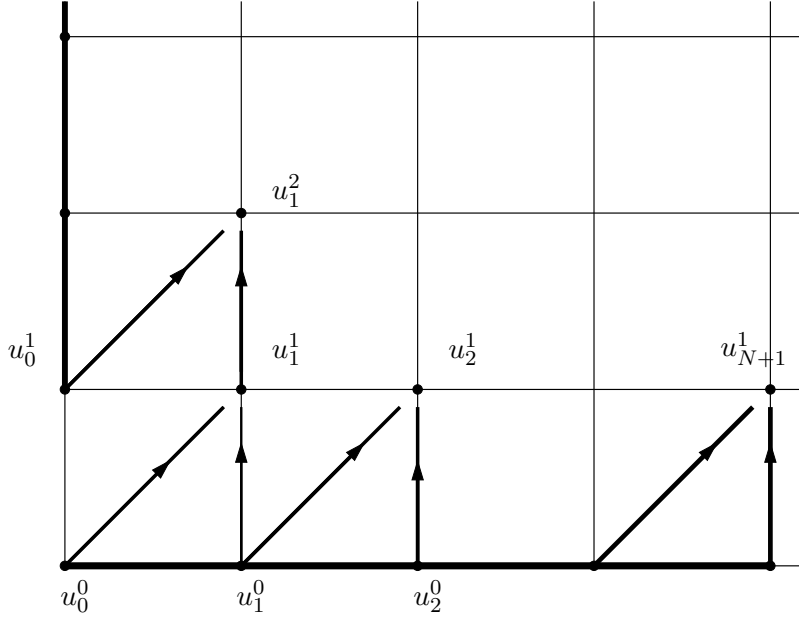
$$\begin{cases} i = 1 & u_1^1 = u_1^0(1 + C) + Cu_0^0 \\ & \dots \\ i = N + 1 & u_{N+1}^1 = u_{N+1}^0(1 + C) + Cu_N^0 \end{cases}$$

$n = 1$ les valeurs u_i^1 ($i = 1, \dots, N + 1$) sont calculées à l'iteration $n = 0$, u_0^1 sont connues de conditions aux limites: $u_0^1 = h^1$.

$$\begin{cases} i = 1 & u_1^2 = u_1^1(1 + C) + Cu_0^1 \\ i = 2 & u_2^2 = u_2^1(1 + C) + Cu_1^1 \\ & \dots \\ i = N + 1 & u_{N+1}^2 = u_{N+1}^1(1 + C) + Cu_N^1 \end{cases}$$

$$n = M \dots$$

Donc on peut tout calculer et visualiser l'algorithme par le dessin suivant:



- Consistance.

L'ordre de precision est Δx en espace et Δt en temps.

- Stabilité.

On injecte dans l'équation discrétisée pour $V > 0$

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

une onde plane: $u_i^n = G^n(k)e^{2i\pi k x_i}$. On obtient:

$$\frac{G^{n+1}(k)e^{2i\pi k x_i} - G^n(k)e^{2i\pi k x_i}}{\Delta t} = -\frac{V}{\Delta x}(G^n(k)e^{2i\pi k x_i} - G^n(k)e^{2i\pi k(x_i - \Delta x)})$$

Le facteur d'amplification $G(k)$ est donné par

$$G(k) = 1 - \frac{V\Delta t}{\Delta x}(1 - e^{-2i\pi k \Delta x})$$

On en déduit

$$|G| = \sqrt{1 - 2\frac{V\Delta t}{\Delta x}(1 - \frac{V\Delta t}{\Delta x})\sin^2(\pi k \Delta x)}$$

Pour la stabilité il faut que $|G(k)| \leq 1$, ce qui donne

$$\frac{V\Delta t}{\Delta x} < 1$$

Si la variable V est une fonction $V(x, t)$ la condition de stabilité se transforme à

$$\frac{\max |V(x, t)| \Delta t}{\Delta x} \leq 1$$

14.1.3 Schéma explicite centré

- Discrétisation:

$$\begin{cases} \frac{u_i^{n+1} - u_i^n}{\Delta t} = -V \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} \\ u_i^0 = g_i \\ u_0^n = h^n \end{cases} \quad (53)$$

- Consistance.

L'ordre de précision est $(\Delta x)^2$ en espace et Δt en temps.

- Stabilité.

On injecte dans l'équation discrétisée pour

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

une onde plane: $u_i^n = G^n(k)e^{2i\pi k x_i}$. On obtient:

$$\frac{G^{n+1}(k)e^{2i\pi k x_i} - G^n(k)e^{2i\pi k x_i}}{\Delta t} = -\frac{V}{\Delta x} (G^n(k)e^{2i\pi k(x_i + \Delta x)} - G^n(k)e^{2i\pi k(x_i - \Delta x)})$$

Le facteur d'amplification $G(k)$ est donné par

$$G(k) = 1 - i \frac{V \Delta t}{\Delta x} \sin(2\pi k \Delta x)$$

On en déduit

$$|G| = 1 + \left(\frac{V \Delta t}{\Delta x} \sin(2\pi k \Delta x) \right)^2 \geq 1.$$

Le schéma est instable.

14.1.4 Schéma de Lax-Friedrichs

Si l'on tient absolument à rester centré et explicite, le schéma de Lax-Friedrichs est un schéma simple, robuste, mais pas très précis.

- Discrétisation:

$$\begin{cases} \frac{2u_i^{n+1} - u_{i+1}^n - u_{i-1}^n}{2\Delta t} = -V \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} \\ u_i^0 = g_i \\ u_0^n = h^n \end{cases} \quad (54)$$

- Consistance.

Le schéma de Lax-Friedrichs est consistant avec l'équation d'advection si le rapport $\frac{\Delta t}{\Delta x}$ est gardé constant lorsque Δx et Δt tendent vers zéro.

L'ordre de précision est Δx en espace et Δt en temps.

- Stabilité.

On injecte dans l'équation discrétisée

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

une onde plane: $u_i^n = G^n(k)e^{2i\pi k x_i}$. On obtient:

$$\frac{2G^{n+1}(k)e^{2i\pi k x_i} - G^n(k)e^{2i\pi k(x_i+\Delta x)} - G^n(k)e^{2i\pi k(x_i-\Delta x)}}{2\Delta t} = -\frac{V}{2\Delta x}(G^n(k)e^{2i\pi k(x_i+\Delta x)} - G^n(k)e^{2i\pi k(x_i-\Delta x)})$$

Le facteur d'amplification $G(k)$ est donné par

$$G(k) = \cos(2\pi k \Delta x) - i \frac{V \Delta t}{\Delta x} \sin(2\pi k \Delta x)$$

On en déduit

$$|G|^2 = \cos^2(2\pi k \Delta x) + \left(\frac{V \Delta t}{\Delta x}\right)^2 \sin^2(2\pi k \Delta x)$$

Le schéma est stable à condition que

$$\frac{|V| \Delta t}{\Delta x} \leq 1$$

- Conditions aux limites numérique.

Il faut fixer $u(t, L)$.

14.1.5 Schéma de Lax-Wendroff

Un schéma centré, explicite, plus précis est le schéma de Lax-Wendroff.

- Discrétisation:

$$\left\{ \begin{array}{l} \frac{u_i^{n+1} - u_i^n}{\Delta t} = -V \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} + \frac{V^2 \Delta t}{2} \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2} \\ u_i^0 = g_i \\ u_0^n = h^n \end{array} \right. \quad (55)$$

- Consistance.

L'ordre de precision est $(\Delta x)^2$ en espace et Δt en temps.

- Stabilité.

On injecte dans l'équation discrétisée

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

une onde plane: $u_i^n = G^n(k)e^{2i\pi k x_i}$. Le facteur d'amplification $G(k)$ est donné par

$$G(k) = 1 - 2\left(\frac{V \Delta t}{\Delta x}\right)^2 \sin^2(\pi k \Delta x) - i \frac{V \Delta t}{\Delta x} \sin(2\pi k \Delta x)$$

On en déduit

$$|G|^2 = 1 - 4\left(\frac{V \Delta t}{\Delta x}\right)^2 \sin^4(\pi k \Delta x) \left(1 - \left(\frac{V \Delta t}{\Delta x}\right)^2\right)$$

Ainsi, le schéma est stable dès que

$$\frac{|V| \Delta t}{\Delta x} \leq 1$$

- Condition aux limites numérique.

Il faut fixer $u(t, L)$.

14.1.6 Schéma de Leap-Frog

- Discrétisation:

$$\left\{ \begin{array}{l} \frac{u_i^{n+1} - u_i^{n-1}}{2\Delta t} = -V \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} \\ u_i^0 = g_i \\ u_0^n = h^n \end{array} \right. \quad (56)$$

Le schéma a besoin u_i^1 pour se lancer:

$$\frac{u_i^1 - u_i^0}{\Delta t} = v_i$$

On peut aussi calculer u_i^1 en utilisant le schéma upwind ou un autre de deux niveaux temporels.

- Consistance.

L'ordre de précision est $(\Delta x)^2$ en espace et $(\Delta t)^2$ en temps.

- Stabilité.

C'est un schéma aux 3 niveaux temporels.

On décompose $u^n(x)$ en une somme de Fourier:

$$u^n(x) = \sum_{k \in \mathbb{Z}} \hat{u}^n(k) e^{2i\pi k x}$$

On applique la transformation de Fourier à

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

et on obtient :

$$\hat{u}^{n+1}(k) - \hat{u}^{n-1}(k) = -2iC \sin(2i\pi k \Delta x) \hat{u}^n(k), \quad C = \frac{V\Delta t}{\Delta x}$$

On cherche la solution $\hat{u}^n(k)$ de la forme

$$\hat{u}^n(k) = G^n$$

On en déduit que G vérifie l'équation caractéristique:

$$G^2 + bG - 1 = 0, \quad b = 2iC \sin(2\pi k \Delta x)$$

La solution générale s'écrit de la forme

$$\hat{u}^n(k) = A_1 \lambda_1^n + A_2 \lambda_2^n,$$

$$\hat{u}^n(k) = (A_1 + nA_2) \lambda^n, \quad \text{si } \Delta = 0$$

où λ_1 et λ_2 sont les racines de l'équation caractéristique.

$$\lambda_{1,2} = -iC \sin(2\pi k \Delta x) \pm \sqrt{\Delta}/2, \quad \Delta = 4(1 - C^2 \sin^2(2\pi k \Delta x))$$

On note aussi Δ le discriminant.

Si $n = 0$ $\hat{u}^0 = A_1 + A_2$

Si $n = 1$ $\hat{u}^1 = A_1\lambda_1 + A_2\lambda_2$

On en déduit

$$A_2 = \frac{\hat{u}^0\lambda_1 - \hat{u}^1}{\lambda_1 - \lambda_2}, \quad A_1 = \frac{\hat{u}^1 - \hat{u}^0\lambda_2}{\lambda_1 - \lambda_2}$$

On peut exprimer les solutions explicitement en fonction de $\hat{u}^0, \hat{u}^1$:

$$\hat{u}^n = \begin{cases} \frac{\lambda_1\lambda_2^n - \lambda_2\lambda_1^n}{\lambda_1 - \lambda_2}\hat{u}^0 + \frac{\lambda_1^n - \lambda_2^n}{\lambda_1 - \lambda_2}\hat{u}^1, & si \Delta \neq 0 \\ (1-n)\lambda_1^n\hat{u}^0 + n\lambda_1^{n-1}\hat{u}^1, & si \Delta = 0 \end{cases}$$

Si $C > 1$, le module de la somme des deux racines est égale à $2C \sin(2\pi k\Delta x)$. Il existe donc $k, \Delta x$ t.q. $\sin(2\pi k\Delta x) = 1$ et le module de la somme des deux racines est égale à $2C > 2$. Le module de l'une des deux racine est plus grand que 1 et le schéma est instable.

Si $C = 1$, on peut avoir $\Delta = 0$ pour certaines valeurs de k et Δx . Dans ce cas, $\lambda_1 = \lambda_2 = i$ et

$$\hat{u}^n = (n\hat{u}^1 + i(n-1)\hat{u}^0)i^{n-1}$$

dépend de n et n'est pas bornée. Le schéma est instable.

Considérons le cas où C est majoré par une constante $M < 1$. Dans ce cas, les racines sont de même module

$$|\lambda_1| = |\lambda_2| = 1$$

De plus, $|\lambda_1 - \lambda_2| = \sqrt{\Delta} > 2\sqrt{1 - M^2} > 0$. On en déduit de l'expression explicite de $\hat{u}^n$

$$|\hat{u}^n| \leq \frac{\hat{u}^0 + \hat{u}^1}{2(1 - M^2)}$$

Ainsi, sous la condition

$$\frac{|V|\Delta t}{\Delta x} < M < 1,$$

le schéma saute - mouton est stable.

- Condition aux limites numérique.

Il faut s'occuper de $u(t, L)$.

- Méthode 1 consiste à établir une condition aux limites fixe en $i = 0$ et une interpolation pour $i = N + 1$:

$$\begin{cases} u_0^n = h^n \\ u_{N+1}^n = -u_{N-1}^n + 2u_N^n \end{cases} \quad (57)$$

- Méthode 2 consiste à établir les conditions aux limites périodiques:

$$\begin{cases} u_{N+1}^n = u_1^n \\ u_0^n = u_N^n \end{cases} \quad (58)$$

14.1.7 Schéma de Crank-Nicolson

- Discrétisation:

$$\left\{ \begin{array}{l} \frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{1}{2} \left(-V \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} - V \frac{u_{i+1}^{n+1} - u_{i-1}^{n+1}}{2\Delta x} \right) \\ u_i^0 = g_i \\ u_0^n = h^n \end{array} \right. \quad (59)$$

- Consistance.

L'ordre de précision est $(\Delta x)^2$ en espace et $(\Delta t)^2$ en temps.

- Stabilité.

Le schéma est inconditionnellement stable en norme L^2 .

- Condition aux limites numérique.

Il faut fixer $u(t, L)$.

On va résoudre numériquement l'équation d'onde avec second membre:

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

Le schéma explicite avec la dérivée centrée donne:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + V \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} = f(t^n, x_i)$$

Ce schéma n'est pas stable. On doit utiliser la méthode de Crank-Nicolson. La méthode de Crank-Nicolson consiste à remplacer chaque fonction u_i^n à l'instant n par la fonction moyenne $(u_i^n + u_i^{n+1})/2$.

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + \frac{V}{2\Delta x} \left[\frac{1}{2}(u_{i+1}^n + u_{i+1}^{n+1}) - \frac{1}{2}(u_{i-1}^n + u_{i-1}^{n+1}) \right] = \frac{1}{2}(f_i^n + f_i^{n+1})$$

On peut réécrire cette équation sous la forme :

$$B_i u_{i-1}^{n+1} + D_i u_i^{n+1} + A_i u_{i+1}^{n+1} = K_i^n, \quad (60)$$

où

$$A_i = \frac{V\Delta t}{4\Delta x}, \quad (61)$$

$$B_i = -\frac{V\Delta t}{4\Delta x} \quad (62)$$

$$D_i = 1 \quad (63)$$

$$K_i^n = \frac{V\Delta t}{4\Delta x} (u_{i+1}^n - u_{i-1}^n) + u_i^n + \frac{\Delta t}{2} (f_i^n + f_i^{n+1}) \quad (64)$$

On note $\frac{c\Delta t}{4\Delta x} = a$

On réécrit l'équation sous la forme de système linéaire:

$$i = 1, \quad -a u_0^{n+1} + u_1^{n+1} + a u_2^{n+1} = K_1^n$$

$$i = 2, \quad -au_1^{n+1} + u_2^{n+1} + au_3^{n+1} = K_2^n$$

.....

$$i = N - 1, \quad -au_{N-2}^{n+1} + u_{N-1}^{n+1} + au_N^{n+1} = K_{N-1}^n$$

$$i = N, \quad -au_{N-1}^{n+1} + u_N^{n+1} + au_{N+1}^{n+1} = K_N^n$$

1) Conditions aux limites fixes.

On utilise une condition aux limites fixe:

$$u_0^{n+1} = h^{n+1}$$

et une condition aux limites artificielles:

$$u_{N+1}^{n+1} = u_N^{n+1}$$

On obtient la forme matricielle suivante:

$$\begin{pmatrix} 1 & a & 0 & 0 & 0 & 0 \\ -a & 1 & a & 0 & 0 & 0 \\ 0 & -a & 1 & a & 0 & 0 \\ & & & & & -a & 1 & a \\ 0 & & & & & 0 & -a & a+1 \end{pmatrix} \cdot \begin{pmatrix} u_1^{n+1} \\ u_2^{n+1} \\ u_3^{n+1} \\ \vdots \\ u_N^{n+1} \end{pmatrix} = \begin{pmatrix} K_0^n \\ K_1^n \\ K_2^n \\ \vdots \\ K_{N-1}^n \\ K_N^n \end{pmatrix}$$

Or

$$M_1 \cdot U^{n+1} = K^n \quad \Rightarrow \quad U^{n+1} = (M_1)^{-1} K^n$$

2) Conditions aux limites périodiques

On utilise les conditions aux limites périodiques

$$u_{-1}^{n+1} = u_N^{n+1}, \quad u_{N+1}^{n+1} = u_0^{n+1}$$

C'est pour cette raison on a introduit un point fictif u_{-1}^n .

On réécrit l'équation sous la forme de système linéaire (avec un point fictif u_{-1}^{n+1}):

$$i = 0, \quad -au_{-1}^{n+1} + u_0^{n+1} + au_1^{n+1} = K_0^n$$

$$i = 1, \quad -au_0^{n+1} + u_1^{n+1} + au_2^{n+1} = K_1^n$$

$$i = 2, \quad -au_1^{n+1} + u_2^{n+1} + au_3^{n+1} = K_2^n$$

.....

$$i = N - 1, \quad -au_{N-2}^{n+1} + u_{N-1}^{n+1} + au_N^{n+1} = K_{N-1}^n$$

$$i = N, \quad -au_{N-1}^{n+1} + u_N^{n+1} + au_{N+1}^{n+1} = K_N^n$$

On obtient la forme matricielle suivante:

$$\begin{pmatrix} 1 & a & 0 & 0 & 0 & & & -a \\ -a & 1 & a & 0 & 0 & & & \\ 0 & -a & 1 & a & 0 & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ a & & & & & & -a & 1 & a \\ & & & & & & 0 & -a & 1 \end{pmatrix} \cdot \begin{pmatrix} u_0^{n+1} \\ u_1^{n+1} \\ u_2^{n+1} \\ \vdots \\ \vdots \\ u_N^{n+1} \end{pmatrix} = \begin{pmatrix} K_0^n \\ K_1^n \\ K_2^n \\ \vdots \\ K_{N-1}^n \\ K_N^n \end{pmatrix}$$

Or

$$M_2 \cdot U^{n+1} = K^n$$

Notre but est de trouver $U^{n+1} = (u_N^{n+1}, u_{N-1}^{n+1}, u_{N-2}^{n+1}, \dots, u_1^{n+1})$, exprimées en termes des $(u_N^n, u_{N-1}^n, u_{N-2}^n, \dots, u_1^n)$. On ne peut plus calculer directement u_i^{n+1} . Pour cela il faut inverser la matrice car

$$U^{n+1} = (M_2)^{-1} K^n.$$

Pour obtenir la matrice inverse d'une **matrice quasi-tridiagonale** on applique l'Algorithme de Thomas généralisé.

On doit remarquer que si on n'introduit pas le point fictif alors une seule condition périodique $u_{N+1}^{n+1} = u_0^{n+1}$ n'est pas suffisante pour résoudre le système de N équations avec $N + 1$ inconnues: $(u_0^{n+1}, u_1^{n+1}, u_2^{n+1}, \dots, u_N^{n+1})$.

$$i = 1, \quad -au_0^{n+1} + u_1^{n+1} + au_2^{n+1} = K_1^n$$

$$i = 2, \quad -au_1^{n+1} + u_2^{n+1} + au_3^{n+1} = K_2^n$$

.....

$$i = N - 1, \quad -au_{N-1}^{n+1} + u_N^{n+1} + au_{N+1}^{n+1} = K_{N-1}^n$$

$$i = N, \quad -au_{N-1}^{n+1} + u_N^{n+1} + au_{N+1}^{n+1} = K_N^n$$

14.2 Schémas de l'équation d'onde d'ordre 2

L'équation d'onde:

$$\left\{ \begin{array}{l} \frac{\partial^2 u}{\partial t^2} = V^2 \frac{\partial^2 u}{\partial x^2} \\ u(t=0, x) = g(x) \\ \frac{\partial u}{\partial t}(t=0, x) = v(x) \end{array} \right. \quad (65)$$

- Discrétisation 1 - conditions aux limites périodiques:

$$\left\{ \begin{array}{l} \frac{u_i^{n+1} + u_i^{n-1} - 2u_i^n}{(\Delta t)^2} = -V^2 \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{(\Delta x)^2} \\ u_i^0 = g_i \\ u_{N+1}^n = u_1^n \\ u_0^n = u_N^n \\ \frac{u_i^1 - u_i^0}{\Delta t} = v_i \end{array} \right. \quad (66)$$

Algorithme 1 du schéma explicite de l'équation d'onde d'ordre 2.

Conditions aux limites périodiques.

Pour $i = 0, \dots, N + 1$

$$u_i^0 = g_i$$

$$u_i^1 = u_i^0 + \Delta t v_i$$

Fin Pour

Pour $n = 0, \dots, M$

Pour $i = 1, \dots, N$

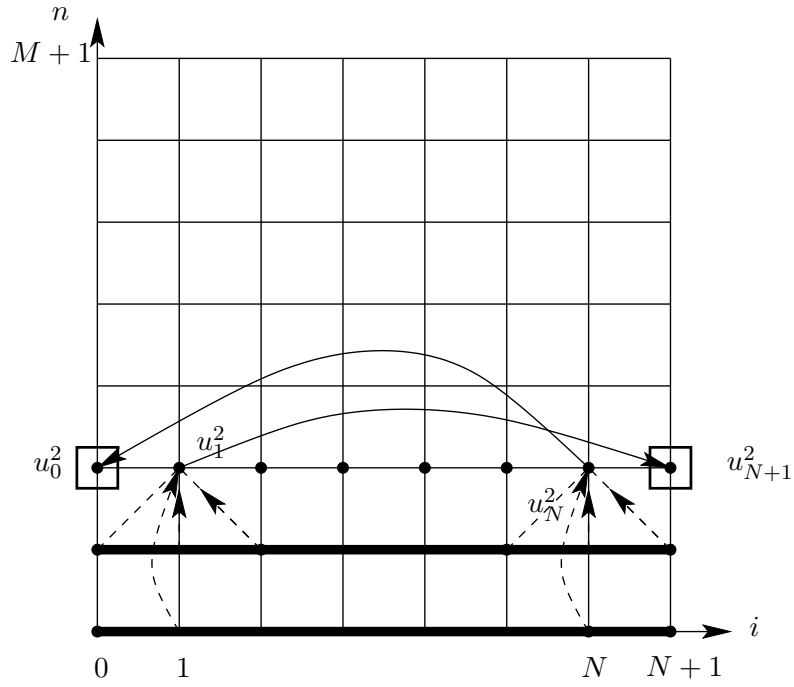
$$u_i^{n+1} = 2u_i^n(1 - C) + C(u_{i-1}^n + u_{i+1}^n) + u_i^{n-1}$$

Fin pour

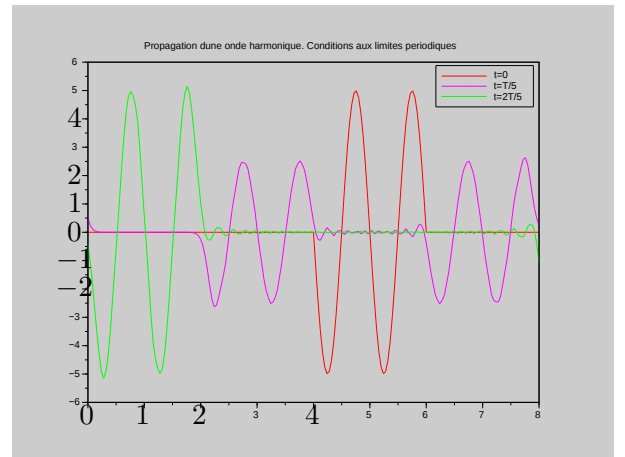
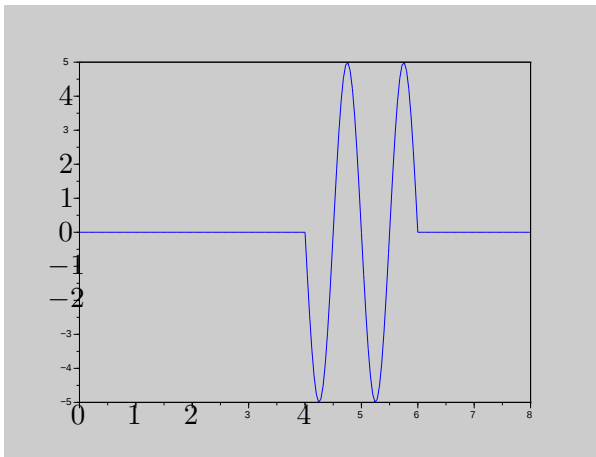
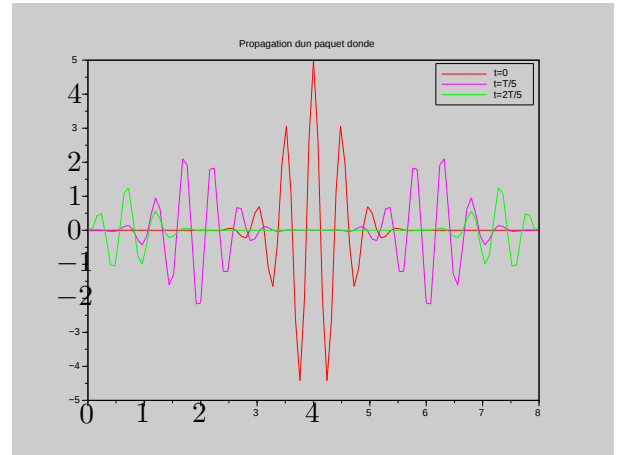
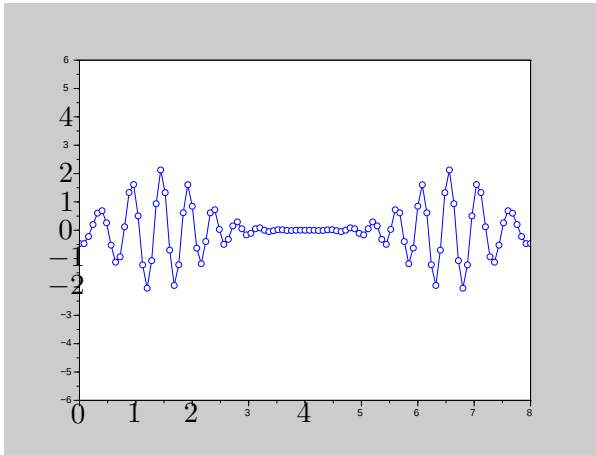
$$u_{N+1}^{n+1} = u_1^{n+1}$$

$$u_0^{n+1} = u_N^{n+1}$$

Fin pour



Les graphes suivants illustrent la propagation d'un paquet d'onde et d'une onde harmonique. On voit que la fluctuation initiale se sépare en deux. Ils se déplacent dans les sens inverses:



- Discrétisation 2 - conditions aux limites fixes:

$$\left\{ \begin{array}{l} \frac{u_i^{n+1} + u_i^{n-1} - 2u_i^n}{(\Delta t)^2} = -V^2 \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{(\Delta x)^2} \\ u_i^0 = g_i \\ u_{N+1}^n = 0 \\ u_0^n = 0 \\ \frac{u_i^1 - u_i^0}{\Delta t} = v_i \end{array} \right. \quad (67)$$

Algorithme 2 du schéma explicite de l'équation d'onde d'ordre 2.

Conditions aux limites fixes.

Pour $i = 0, \dots, N + 1$

$u_i^0 = g_i$

$u_i^1 = u_i^0 + \Delta t v_i$

Fin Pour

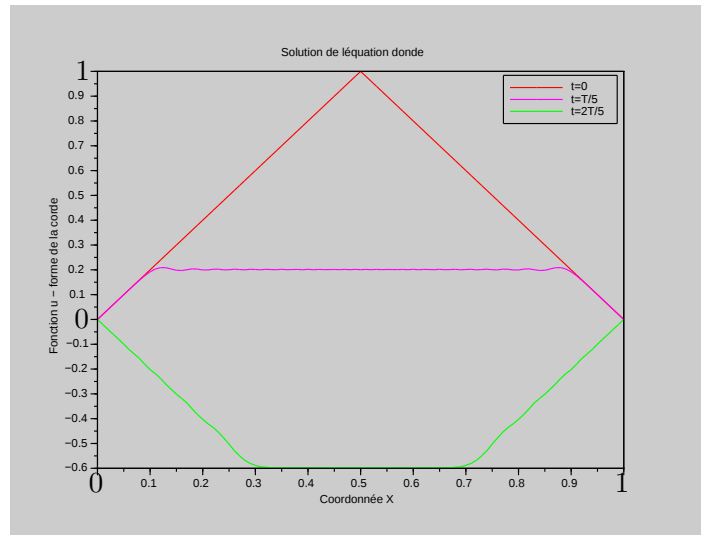
Pour $n = 1, \dots, M + 1$

```

 $u_0^n = 0$ 
 $u_{N+1}^{n+1} = 0$ 
Fin Pour
Pour  $n = 0, \dots, M$ 
  Pour  $i = 1, \dots, N$ 
     $u_i^{n+1} = 2u_i^n(1 - C) + C(u_{i-1}^n + u_{i+1}^n) + u_i^{n-1}$ 
  Fin pour
Fin pour

```

Présentons ici les solutions de l'équation d'ordre 1 qui représentent la vibration d'une corde (simulation de la guitare)



- Discrétisation 3 - condition aux limites fixe en $i = 0$ et l'interpolation pour $i = N + 1$:

$$\left\{ \begin{array}{l} \frac{u_i^{n+1} + u_i^{n-1} - 2u_i^n}{(\Delta t)^2} = -V^2 \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{(\Delta x)^2} \\ u_i^0 = g_i \\ u_0^n = h^n \\ u_{N+1}^n = -u_{N-1}^n + 2u_N^n \\ \frac{u_i^1 - u_i^0}{\Delta t} = v_i \end{array} \right. \quad (68)$$

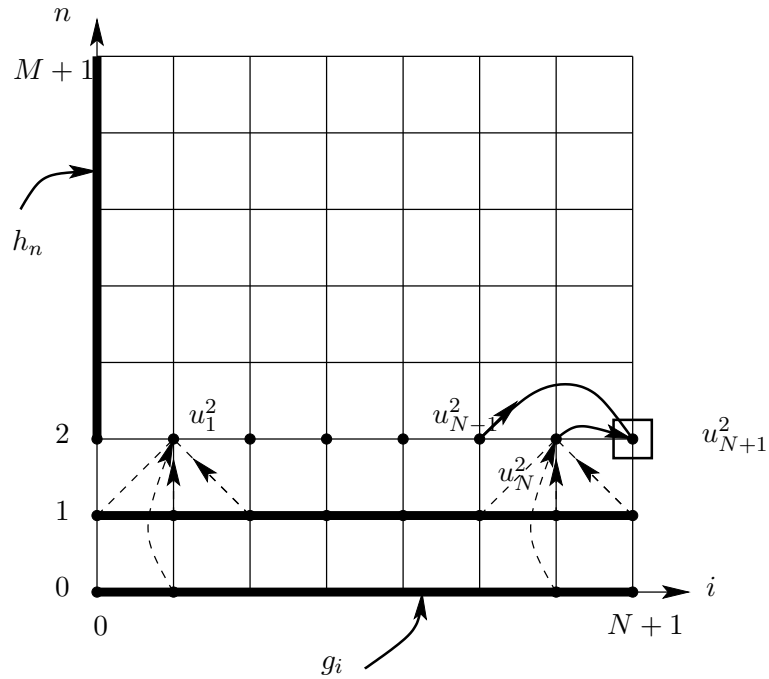
Algorithme 3 du schéma explicite de l'équation d'onde d'ordre 2

Interpolation à droite

```

Pour    $i = 0, \dots, N + 1$ 
 $u_i^0 = g_i$ 
 $u_i^1 = u_i^0 + \Delta t v_i$ 
Fin Pour
Pour    $n = 1, \dots, M + 1$ 
 $u_0^n = h^n$ 
Fin Pour
Pour    $n = 0, \dots, M$ 
  Pour    $i = 1, \dots, N$ 
 $u_i^{n+1} = 2u_i^n(1 - C) + C(u_{i-1}^n + u_{i+1}^n) + u_i^{n-1}$ 
  Fin pour
 $u_{N+1}^{n+1} = 2u_N^{n+1} - u_{N-1}^{n+1}$ 
Fin pour

```



14.3 Conditions aux limites numériques

Pour la résolution numérique on a besoin d'une condition aux limites supplémentaire: il faut fixer $u(t, L)$. c'est une condition aux limites artificielle, car on n'a pas d'en besoin pour la résolution analytique. Il y a plusieurs approches.

- Extrapolation.

On utilise la formule d'interpolation de Lagrange

$$f(x) = f(x_1) \frac{x - x_2}{x_1 - x_2} + f(x_2) \frac{x - x_1}{x_2 - x_1}$$

On peut faire l'interpolation en même niveau temporel:

L'interpolation d'ordre 0 :

$$u_N^n = u_{N-1}^n$$

L'interpolation d'ordre 1 (l'interpolation linéaire):

$$u_N^n = u_{N-2}^n \frac{x_N - x_{N-1}}{x_{N-2} - x_{N-1}} + u_{N-1}^n \frac{x_N - x_{N-2}}{x_{N-1} - x_{N-2}}$$

Si le maillage est uniforme la formule devient:

$$u_N^n = -u_{N-2}^n + 2u_{N-1}^n$$

- Méthode des caractéristiques

L'interpolation d'ordre 0 :

$$u_N^{n+1} = u_{N-1}^n$$

L'interpolation d'ordre 1 :

$$u_N^{n+1} = -u_{N-2}^n + 2u_{N-1}^n$$

15 Equations elliptiques

15.1 Equation de Poisson en coordonnées cartésiennes

On cherche une fonction u que vérifie l'équation de Poisson

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y)$$

Très souvent cette fonction représente le potentiel d'un champ électromagnétique ou gravitationnel.

- 1) Discretisons les variables: spatiales :

x : $x_0 = 0, \dots, x_i, \dots, x_{N_x+1} = L_x$.

y : $y_0 = 0, \dots, y_j, \dots, y_{N_y+1} = L_y$.

Donc les points (x_i, y_j) forment une grille uniforme du rectangle $[0, L_x] \times [0, L_y]$

$$\begin{cases} i = 0, 1, 2, \dots, N_x + 1 \\ j = 0, 1, 2, \dots, N_y + 1 \\ \Delta x = \frac{L_x}{N_x+1}, \quad \Delta y = \frac{L_y}{N_y+1} \\ V(x_i, y_j) \equiv V_{ij} \end{cases} \quad (69)$$

2) Discrétisons les conditions aux limites:

$$\left\{ \begin{array}{l} u_{0,j} = g_j \\ u_{i,0} = h_i \\ u_{N_x+1,j} = \mu_j \\ u_{i,N_y+1} = \nu_i \end{array} \right. \quad (70)$$

3) Pour discrétiser l'équation de Poisson on utilise:

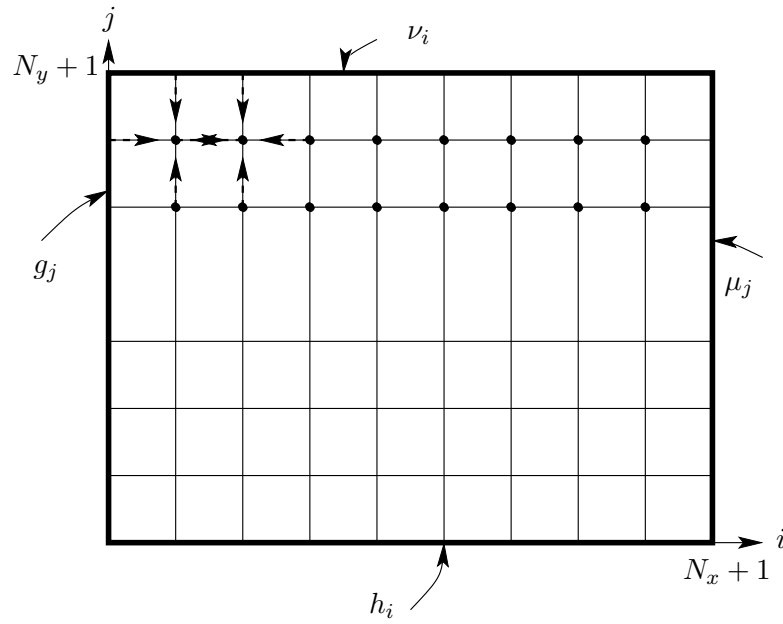
- pour $\frac{\partial^2 u}{\partial x^2}$ la seconde dérivée, centrée,

$$\frac{\partial^2 u}{\partial x^2} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{(\Delta x)^2}$$

- pour $\frac{\partial^2 u}{\partial y^2}$ la seconde dérivée, centrée,

$$\frac{\partial^2 u}{\partial y^2} = \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{(\Delta y)^2}$$

Méthode 1 de la résolution : Méthode itérative de Gauss-Seidel ou Jacobi.



On écrit l'équation de la forme (on prend $\Delta_x = \Delta_y = \Delta$, et $N_x = N_y = N$)

$$u_{ij} = \frac{1}{4}(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - f_{i,j}(\Delta)^2)$$

A chaque itération on calcule toutes les valeurs u_{ij} à l'aide de la formule établie. On répète le même calcul un grand nombre de fois et on met en place le critère d'arrêt:

$$\left| \sum_{i,j=1}^N |u_{ij}^{(k+1)}| - \sum_{i,j=1}^N |u_{ij}^{(k)}| \right| < \varepsilon$$

Ici on compare les valeurs de u de deux itérations successives: (k) et $(k + 1)$.

Algorithme de Gauss-Seidel pour l'équation de Poisson
--

```

Pour     $j = 0, \dots, N + 1$ 
 $u_{0,j} = g_j$ 
 $u_{N+1,j} = \mu_j$ 
Fin Pour

Pour     $i = 0, \dots, N + 1$ 
 $u_{i,0} = h_i$ 
 $u_{i,N+1} = \nu_i$ 
Fin Pour

Initialisation:
Pour     $i = 1 \dots N$ 
Pour     $j = 1 \dots N$ 
 $u_{i,j} = 0$ 
Fin Pour
Fin Pour
Pour     $k = 1 \dots 10000$ 
 $u_{i,j}^{old} = u_{i,j}$ 

Pour     $i = 1 \dots N$ 
Pour     $j = 1 \dots N$ 

$$u_{ij} = \frac{1}{4}(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - f_{i,j}(\Delta)^2)$$


Fin Pour
Fin Pour
Si     $|\sum_{i,j=1}^N |u_{ij}| - \sum_{i,j=1}^N |u_{ij}^{old}|| < \varepsilon$  et  $k > 50$  arrêt
Fin Pour

```

Méthode 2 de la résolution : Méthode de Gauss ou de decomposition LU.

On introduit la variable x selon la règle

$$x_k = u_{ij}, \quad k = j + N(i - 1)$$

Soit $N=3$.

On peut réécrire le système le la forme:

$$\frac{1}{\Delta^2} \begin{pmatrix} -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -4 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -4 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & -4 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & -4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \end{pmatrix} = \begin{pmatrix} f_{11} - \frac{u_{01}+u_{10}}{\Delta^2} \\ f_{12} - \frac{u_{02}}{\Delta^2} \\ f_{13} - \frac{u_{03}+u_{14}}{\Delta^2} \\ f_{12} - \frac{u_{20}}{\Delta^2} \\ f_{22} \\ f_{23} - \frac{u_{24}}{\Delta^2} \\ f_{31} - \frac{u_{30}+u_{41}}{\Delta^2} \\ f_{23} - \frac{u_{24}}{\Delta^2} \\ f_{33} - \frac{u_{34}+u_{43}}{\Delta^2} \end{pmatrix}$$

On a obtenu une matrice diagonale par blocks.

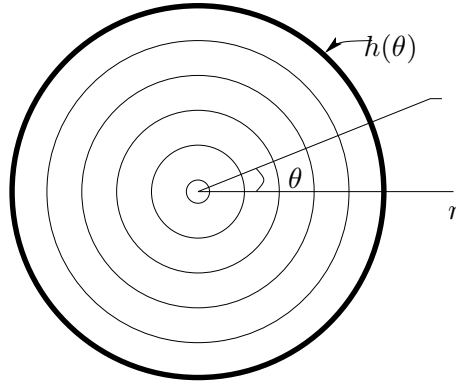
$$A = \frac{1}{\Delta^2} \begin{pmatrix} B & I & 0 \\ I & B & I \\ 0 & I & B \end{pmatrix}, \quad B = \begin{pmatrix} -4 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & -4 \end{pmatrix}$$

La matrice A est une matrice creuse et il existe des méthodes spatiales pour la résolution des systèmes linéaires avec les matrices creuses.

15.2 Equation de Poisson en coordonnées polaires

L'équation s'écrit de la façon suivantes:

$$\frac{1}{r} \frac{\partial u}{\partial r} + \frac{\partial^2 u}{\partial r^2} + \frac{1}{r^2} \frac{\partial^2 u}{\partial \theta^2} = f(r, \theta)$$



- 1) Pour surmonter le problème de singularité en $r=0$ on utilise le maillage special (staggered grid)

$$r_i = (i - 1/2)\Delta r$$

- 2) Discrétisons les variables: spatiales :

$$r: r_1 = \frac{\Delta r}{2}, \dots, r_i, \dots, r_{N_r+1} = R.$$

$$\theta: \theta_0 = 0, \dots, \theta_j, \dots, \theta_{N_\theta+1} = 2\pi.$$

- 3) On impose une condition aux limites (avec une fonction périodique $h(\theta)$).

$$\left\{ \begin{array}{l} u_{N_r+1,j} = h_j, \quad j = 0 \dots N_\theta + 1 \end{array} \right.$$

- 4) On impose les conditions périodiques (on introduit un point fictif):

$$\left\{ \begin{array}{l} u_{i,0} = u_{i,N_\theta+1} \\ u_{i,-1} = u_{i,N_\theta} \end{array} \right.$$

Discrétisons l'équation. On peut écrire l'équation de la forme:

$$\frac{2i}{2i-1} u_{i+1,j} - 2 \left(1 + \frac{4}{(2i-1)(\Delta\theta)^2} \right) u_{ij} + \frac{2(i-1)}{2i-1} u_{i-1,j} +$$

$$\frac{4}{(2i-1)^2} \frac{1}{(\Delta\theta)^2} u_{i,j+1} + \frac{4}{(2i-1)^2} \frac{1}{(\Delta\theta)^2} u_{i,j-1} = f_{ij}(\Delta r)^2$$

Prenons $i=1$. On voit que le coefficient devant $u_{0,j}$ s'annule. On n'a pas donc besoin une condition aux limites à $i=0$.

L'algorithme itératif (Gauss-Seidel) permet de calculer les valeurs u_{ij} .

Algorithme de Gauss-Seidel pour l'équation de Poisson en coordonnées polaires

Pour $j = 0, \dots, N_\theta + 1$

$u_{N_r+1,j} = h_j$

Fin Pour

Initialisation:

Pour $i = 1 \dots N_r$

Pour $j = 1 \dots N_\theta + 1$

$u_{i,j} = 0$

Fin Pour

Fin Pour

Pour $k = 1 \dots 10000$

$u_{i,j}^{old} = u_{i,j}$

Pour $i = N_r \dots 1$

Pour $j = 1 \dots N_\theta$

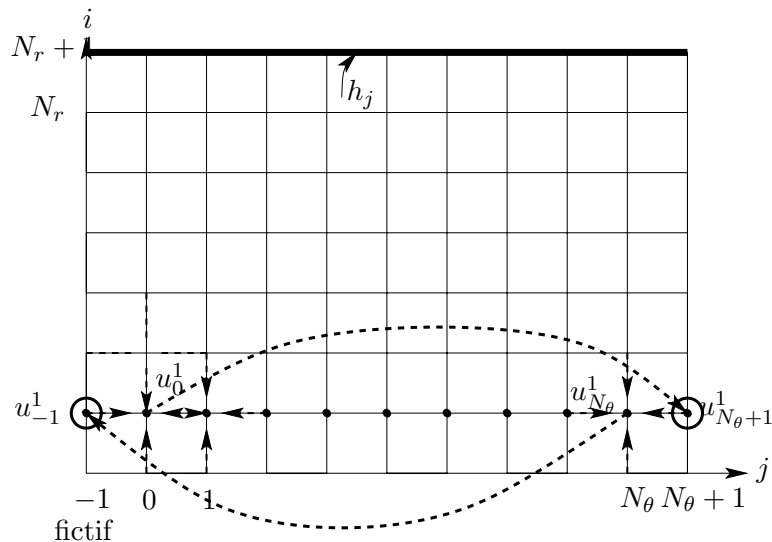
$$u_{ij} = \frac{\frac{2i}{2i-1} u_{i+1,j} + \frac{2(i-1)}{2i-1} u_{i-1,j} + \frac{4}{(2i-1)^2} \frac{1}{(\Delta\theta)^2} u_{i,j+1} + \frac{4}{(2i-1)^2} \frac{1}{(\Delta\theta)^2} u_{i,j-1} - f_{ij}(\Delta r)^2}{(1 + \frac{4}{(2i-1)(\Delta\theta)^2})}$$

Fin Pour

Fin Pour

Si $|\sum_{i,j=1}^{N_\theta, N_r} |u_{ij}| - \sum_{i,j=1}^{N_\theta, N_r} |u_{ij}^{old}|| < \varepsilon$ **et** $k > 50$ **arrêt**

Fin Pour



Remarque:

1) La valeur $u_{N_r,0}$ se calcule.

2) Les valeurs $u_{0,j}$, $j = 1 \dots N_\theta + 1$ en réalité n'existent pas. Nous aussi ne les calculons pas. Unique valeur qu'il existe est $u_{0,0}$. Cette valeur peut être calculée par l'interpolation:

$$u_{0,0} = 2u_{1,0} - u_{2,0}$$

16 Résolution numérique de l'équation de chaleur à 2 dimension

16.1 Méthode explicite

Discretisons l'espace-temps: discretisons les variables spatiales et temporelle

x : $x_0 = 0, \dots, x_i = i \cdot \Delta x, \dots, x_{N_x+1} = L$.

y : $y_0 = 0, \dots, y_j = j \cdot \Delta y, \dots, y_{N_y+1} = L$.

t : $t_0 = 0, \dots, t^n = n\Delta t, \dots, t_{M+1} = T$.

On note $u_{i,j}^n$ la valeur d'une solution calculée à l'aide d'un schéma numérique aux points $x = x_i$, y_j , $t = t^n$. Discutons les idées principales de la résolution numérique de l'équation de la chaleur:

$$\frac{\partial u}{\partial t} = D \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right), \quad (71)$$

où D est le coefficient de diffusion.

Les conditions aux limites :

$$\begin{cases} u(0, y, t) = f_1(y, t) \\ u(L_1, y, t) = f_2(y, t) \\ u(x, 0, t) = f_3(x, t) \\ u(x, L_2, t) = f_4(x, t) \end{cases} \quad (72)$$

et la condition initiale

$$\{ u(x, y, 0) = \phi(x, y) \quad (73)$$

On calcule la solution $U_{i,j}^{M+1}$ à $t = T$ et on ne cherche pas à calculer les valeurs aux instants temporels intermédiaires. Pour cela on introduit les matrices "old" et "new": $U_{old,i,j}$, $U_{new,i,j}$ on les update dans la boucle temporelle.

Algorithme d'Euler pour l'équation de la chaleur à 3 dimensions

Initialisation: Conditions initiales

Pour $j = 0, \dots, N_y + 1$

Pour $i = 0, \dots, N_x + 1$

$U_{initial_{i,j}} = \phi(x(i), y(j))$

Fin Pour

Fin Pour

Uold=Uinitial;

Pour $n = 1 : M + 1$

Pour $i = 1 : N_x$

Pour $j = 1 : N_y$

Uold=Uinitial;

Pour $n = 1 : 1 + M$

Pour $i = 1 : N_x$

Pour $j = 1 : N_y$

$$U_{new_{i,j}} = U_{old_{i,j}} + (-2 \cdot U_{old_{i,j}} + U_{old_{i-1,j}} + U_{old_{i+1,j}}) \cdot \Delta_t / (\Delta_x)^2 + (-2 \cdot U_{old_{i,j}} + U_{old_{i,j-1}} + U_{old_{i,j+1}}) \cdot \Delta_t / (\Delta_y)^2$$

end

$U_{new_{1,j}} = 0;$

$U_{new_{N_x+1,j}} = 0;$

end

$U_{new_{i,1}} = 0;$

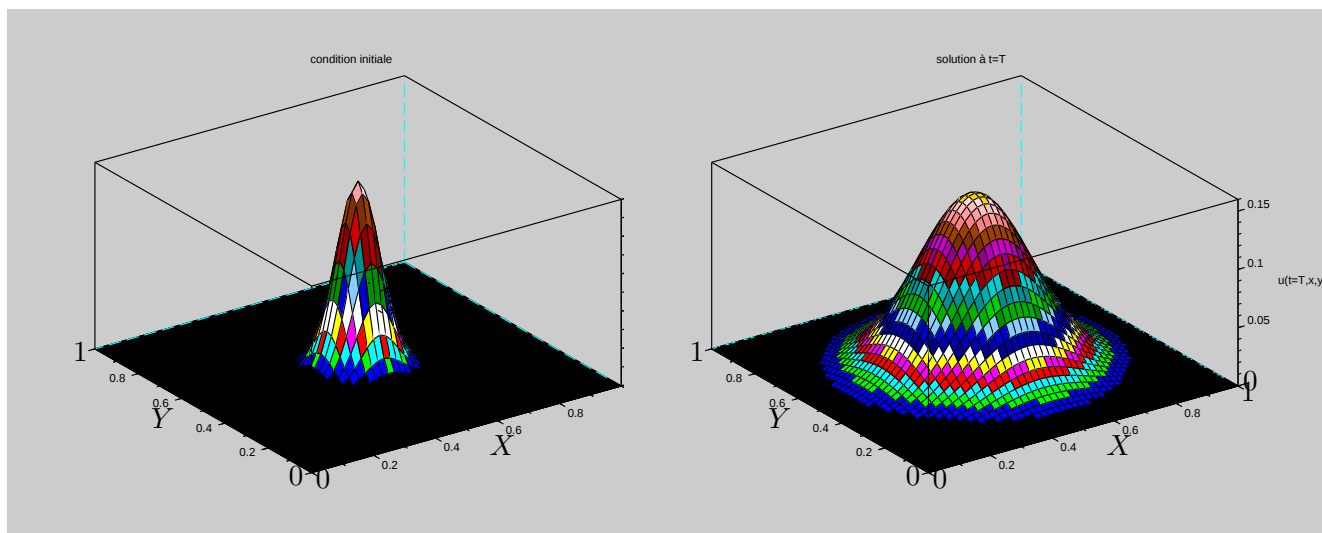
$U_{new_{i,N_y+1}} = 0;$

$u_{old} = u_{new};$

Fin Pour

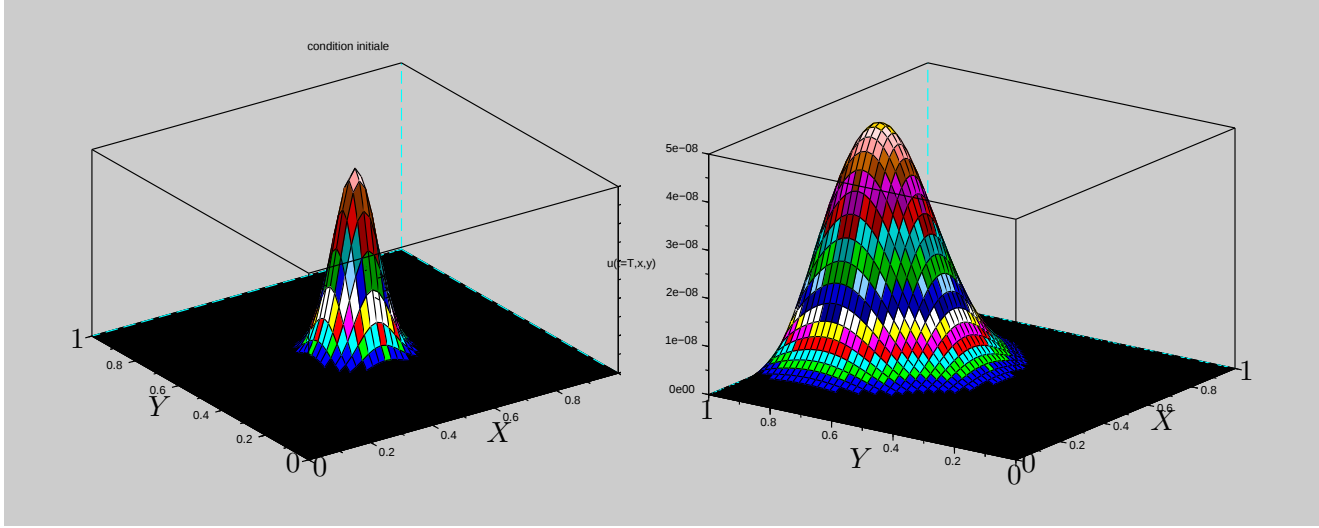
Présentons ici la solution numérique pour la condition initiales gaussienne

$$\phi(x, y) = e^{-180 \cdot (x-1/2)^2 - 180 \cdot (y-1/2)^2}$$



De la même façon on peut résoudre l'équation d'advection-diffusion et démontrer la propagation d'une fluctuation initiale.

$$\frac{\partial u}{\partial t} = D \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} \quad (74)$$



16.2 Méthode implicite

L'idée est la résolution alternativement des deux équations unidimensionnelles:

$$\frac{\partial u}{\partial t} = 2D \frac{\partial^2 u}{\partial x^2} \quad \text{et} \quad \frac{\partial u}{\partial t} = 2D \frac{\partial^2 u}{\partial y^2}$$

La résolution peut être faite en deux étapes par une technique de directions alternées (ADI):

$$\frac{u_{i,j}^{n+1/2} - u_{i,j}^n}{\Delta t} = \beta \frac{u_{i+1,j}^{n+1/2} - 2u_{i,j}^{n+1/2} + u_{i-1,j}^{n+1/2}}{2(\Delta x)^2} + \beta \frac{u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n}{2(\Delta x)^2}$$

Ce schéma est explicite par rapport à la variable j et implicite par rapport à la variable i .

$$\frac{u_{i,j}^{n+1} - u_{i,j}^{n+1/2}}{\Delta t} = \beta \frac{u_{i,j+1}^{n+1} - 2u_{i,j}^{n+1} + u_{i,j-1}^{n+1}}{2(\Delta y)^2} + \beta \frac{u_{i,j+1}^{n+1/2} - 2u_{i,j}^{n+1/2} + u_{i,j-1}^{n+1/2}}{2(\Delta y)^2}$$

Ce schéma est explicite par rapport à la variable i et implicite par rapport à la variable j .

La discrétisation des équations se fait en deux étapes et donne les équations suivantes:

$$-d_1 u_{i+1,j}^{n+1/2} + (1 + 2d_1) u_{i,j}^{n+1/2} - d_1 u_{i-1,j}^{n+1/2} = d_2 u_{i,j+1}^n + (1 - 2d_2) u_{i,j}^n + d_2 u_{i,j-1}^n \quad (75)$$

$$-d_2 u_{i,j+1}^{n+1} + (1 + 2d_2) u_{i,j}^{n+1} - d_2 u_{i,j-1}^{n+1} = d_1 u_{i+1,j}^{n+1/2} + (1 - 2d_1) u_{i,j}^{n+1/2} + d_1 u_{i-1,j}^{n+1/2} \quad (76)$$

Ici

$$d_1 = \frac{D\Delta t}{(\Delta x)^2} \quad \text{et} \quad d_2 = \frac{D\Delta t}{(\Delta y)^2}.$$

Pour chaque j on applique l'algorithme de Thomas à

$$A \boxed{u_{i+1,j}^{n+1/2}} + D \boxed{u_{i,j}^{n+1/2}} + B \boxed{u_{i-1,j}^{n+1/2}} = K_{ij}^n$$

pour trouver $u_{i,j}^{n+1/2}$.

Pour chaque i on applique l'algorithme de Thomas à

$$\tilde{A}u_{i,j+1}^{n+1} + \tilde{D}u_{i,j}^{n+1} + \tilde{B}u_{i,j-1}^{n+1} = \boxed{K_{ij}^{n+1/2}}$$

pour trouver $u_{i,j}^n$.

Le schéma de ADI est stable si

$$d_1 + d_2 \leq \frac{1}{2}.$$

17 EDP nonlinéaires

17.1 Résolution numérique de l'équation de Burgers à une dimension.

L'équation de Burgers nonlinéaire qui régit par exemple le mouvement d'un fluide visqueux a la forme:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \frac{\mu}{\rho} \frac{\partial^2 u}{\partial x^2}, \quad (77)$$

où u , ρ et μ sont respectivement la vitesse, la densité et le coefficient de viscosité d'un fluide.

Cette équation réécrite en terme des variables sans dimensions admet la forme:

$$\frac{\partial u'}{\partial t'} + u' \frac{\partial u'}{\partial x'} = \frac{1}{Re} \frac{\partial^2 u'}{\partial x'^2} \quad (78)$$

Ici on a introduit les variables sans dimensions:

$$u' = \frac{u}{u_0}, \quad t' = \frac{t}{L/u_0}, \quad y' = \frac{y}{L}, \quad \frac{1}{Re} = \frac{\mu}{\rho u_0 L}.$$

Par la suite on va écrire toujours u , x , t au lieu de u' , x' , t' .

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \frac{1}{Re} \frac{\partial^2 u}{\partial x^2}$$

Pour une collision de deux ondes de choc l'équation admet la solution exacte:

$$u(x, t) = -2 \frac{shx}{chx + e^{-\frac{t}{Re}}} \quad (79)$$

En effet sur le graphe de distribution des vitesses $u(x, t)$ vous voyez deux ondes de vitesse $u(-\infty, -\infty) = 2$ et $u(\infty, -\infty) = -2$ qui se rapprochent dans le temps.

17.1.1 Méthode de Mac-Cormack

La méthode de Mac-Cormack est explicite et n'exige que les dérivées d'ordre 1 par rapport au temps. La méthode consiste à calculer la vitesse en utilisant la schéma aux différences finies "progressives" en temps et en espace et l'introduction de la valeur moyenne de variation de la vitesse entre les moments t et $t + \Delta t$.

$$u_i^{n+1} = u_i^n + \left(\frac{\partial u}{\partial t}\right)_{moy} \Delta t = u_i^n + \frac{1}{2} \left(\left(\frac{\partial u}{\partial t}\right)_i^n + \left(\frac{\partial \tilde{u}}{\partial t}\right)_i^{n+1} \right) \Delta t \quad (80)$$

Pour calculer $\left(\frac{\partial u}{\partial t}\right)_i^n$ on discrétise l'équation en forme conservative:

$$\frac{\partial u}{\partial t} + \frac{\partial F}{\partial x} = -\frac{1}{Re} \frac{\partial^2 u}{\partial x^2}, \text{ ou } F = \frac{u^2}{2} \implies \quad (81)$$

$$\left(\frac{\partial u}{\partial t}\right)_i^n = \frac{F_{i+1}^n - F_i^n}{\Delta x} + \frac{1}{Re} \frac{1}{(\Delta x)^2} (u_{i-1}^n - 2u_i^n + u_{i+1}^n) \quad (82)$$

Pour calculer $\left(\frac{\partial \tilde{u}}{\partial t}\right)_i^n$ on utilise le schéma à deux étapes: Prédiction et Correction:

Prédiction d'une vitesse intermédiaire :

$$\tilde{u}_i^{n+1} = u_i^n + \left(\frac{\partial u}{\partial t}\right)_i^n \Delta t$$

Correction: Dans l'étape de correction on utilise la schéma aux différences finies "rétrogradées" et centrées en espace

$$\left(\frac{\partial \tilde{u}}{\partial t}\right)_i^{n+1} = -\frac{\tilde{F}_i^{n+1} - \tilde{F}_{i-1}^{n+1}}{\Delta x} + \frac{1}{Re} \frac{1}{(\Delta x)^2} (\tilde{u}_{i-1}^{n+1} - 2\tilde{u}_i^{n+1} + \tilde{u}_{i+1}^{n+1}), \quad \tilde{F}_i^{n+1} = \frac{1}{2} (\tilde{u}_i^{n+1})^2$$

On applique les conditions initiales et aux limites:

$$\begin{cases} u_i^0 = -2 \frac{shx_i}{chx_i + 1} \\ u_0^n = 2, \quad u_N^n = -2. \end{cases} \quad (83)$$

17.1.2 Méthode de Crank-Nicolson.

L'idée de la méthode est la discrétisation de l'équation afin d'obtenir après la linéarisation le système des équations linéaires avec une matrice tridiagonale. L'application de l'algorithme de Thomas permet d'achever la solution.

On applique Crank-Nicolson à (82):

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = -\frac{1}{2} \left(\frac{F_{i+1}^n - F_{i-1}^n}{2\Delta x} + \frac{F_{i+1}^{n+1} - F_{i-1}^{n+1}}{2\Delta x} \right) + \frac{1}{2Re} \left(\frac{(u_{i-1}^n - 2u_i^n + u_{i+1}^n)}{(\Delta x)^2} + \frac{(u_{i-1}^{n+1} - 2u_i^{n+1} + u_{i+1}^{n+1})}{(\Delta x)^2} \right)$$

Pour pouvoir réduire cette équation au système tridiagonal il faut linéariser les termes F_{i+1}^{n+1} et F_{i-1}^{n+1} . On utilise les formules:

$$F_i^{n+1} = F_i^n + \left(\frac{\partial F}{\partial u}\right)_i^n \left(\frac{\partial u}{\partial t}\right)_i^n \Delta t = F_i^n + u_i^n (u_i^{n+1} - u_i^n) \quad (84)$$

L'équation se réduit au système tridiagonal:

$$b_i^n u_{i-1}^{n+1} + d_i^n u_i^{n+1} + a_i^n u_{i+1}^{n+1} = K_i^n,$$

où

$$\left\{ \begin{array}{l} b_i^n = -\frac{\Delta t}{4\Delta x} u_{i-1}^n - \frac{1}{2}s \\ d_i^n = 1 + s \\ a_i^n = \frac{\Delta t}{4\Delta x} u_{i+1}^n - \frac{1}{2}s \\ K_i^n = \frac{1}{2}s u_{i-1}^n + (1-s)u_i^n + \frac{1}{2}s u_{i+1}^n, \text{ ou} \\ s = \frac{\Delta t}{Re(\Delta x)^2} \end{array} \right. \quad (85)$$

A ce système on applique l'algorithme de Thomas avec les conditions initiales et aux limites:

$$\left\{ \begin{array}{l} u_0^n = 2, \\ u_{N+1}^n = -2 \end{array} \right. \quad (86)$$

$$\left\{ \begin{array}{l} u_i^0 = 2 \frac{shx_i}{chx_i+1}, i = 0 \dots N/2 \\ u_i^0 = -2 \frac{shx_i}{chx_i+1}, i = N/2 + 1, \dots N + 1 \end{array} \right.$$

17.2 Résolution numérique de l'équation de Burgers non visqueuse à une dimension.

L'équation de Burgers non visqueuse a la forme:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = 0 \quad (87)$$

où u est la vitesse d'un fluide.

Imposons les conditions initiales:

$$u(x, 0) = \left\{ \begin{array}{ll} u_l = 1 & \text{si } 0 < x < 2 \\ u_r = 0 & \text{si } 2 < x < 4 \end{array} \right. \quad (88)$$

Pour la propagation d'une onde de choc l'équation admet la solution exacte. C'est un choc qui se propage sans changer la forme avec la célérité $v_s = \frac{1}{2}(u_l + u_r) = \frac{1}{2}$.

17.2.1 Méthode de Lax.

$$u_i^{n+1} = \frac{1}{2}(u_{i+1}^n + u_{i-1}^n) - \frac{\Delta t}{4\Delta x}((u_{i+1}^n)^2 - (u_{i-1}^n)^2)$$

17.2.2 Méthode de Lax-Wendroff.

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{2\Delta x}(F_{i+1}^n - F_{i-1}^n) + \frac{(\Delta t)^2}{4(\Delta x)^2}((u_{i+1}^n + u_i^n)(F_{i+1}^n - F_i^n) - (u_i^n + u_{i-1}^n)(F_i^n - F_{i-1}^n)),$$

où $F_i^n = (u_i^n)^2/2$.

17.2.3 Méthode de Mac-Cormack.

La méthode de Mac-Cormack est explicite et n'exige que les dérivées d'ordre 1 par rapport au temps. La méthode consiste à calculer la vitesse en utilisant la schéma aux différences finies "progressives" en temps et en espace et l'introduction de la valeur moyenne de variation de la vitesse entre les moments t et $t + \Delta t$.

$$u_i^{n+1} = u_i^n + \left(\frac{\partial u}{\partial t}\right)_{moy} \Delta t = u_i^n + \frac{1}{2} \left(\left(\frac{\partial u}{\partial t}\right)_i^n + \left(\frac{\partial \tilde{u}}{\partial t}\right)_i^{n+1} \right) \Delta t \quad (89)$$

Pour calculer $(\frac{\partial u}{\partial t})_i^{n+1}$ on discrétise l'équation () en forme conservative:

$$\frac{\partial u}{\partial t} + \frac{\partial F}{\partial x} = 0, \quad \text{où } F = \frac{u^2}{2} \implies \quad (90)$$

$$\left(\frac{\partial u}{\partial t}\right)_i^n = -\frac{F_{i+1}^n - F_i^n}{\Delta x} \quad (91)$$

Pour calculer $(\frac{\partial \tilde{u}}{\partial t})_i^n$ on utilise le schéma à deux étapes: Prediction et Correction:

Prediction d'une vitesse intermédiaire :

$$\tilde{u}_i^{n+1} = u_i^n + \left(\frac{\partial u}{\partial t}\right)_i^n \Delta t$$

Correction: Dans l'étape de correction on utilise la schéma aux différences finies "rétrogradées" et centrés en espace:

$$\left(\frac{\partial \tilde{u}}{\partial t}\right)_i^{n+1} = -\frac{\tilde{F}_i^{n+1} - \tilde{F}_{i-1}^{n+1}}{\Delta x}$$

On impose soit les conditions aux limites :

$$u_0^n = 1, \quad u_N^n = 0,$$

soit les conditions aux limites périodiques.

17.3 Résolution numérique de l'équation de Korteweg de Vries.

17.3.1 Discrétisation

On applique la méthode aux différences finies de Zabusky pour résoudre l'équation de Korteweg de Vries nonlinéaire qui décrit la propagation des solitons.

$$\frac{\partial u}{\partial t} + 6u \frac{\partial u}{\partial x} + \frac{\partial^3 u}{\partial x^3} = 0$$

1) On utilise Leap-Frog(Saute mouton) pour approximer la dérivée temporelle, la difference centrale pour approximer la dérivée spatiale et l'approximation d'ordre 2 pour $\frac{\partial^3 u}{\partial x^3}$

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

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

$$\frac{\partial^3 u}{\partial x^3} \approx \frac{u_{i+2}^n - 2u_{i+1}^n + 2u_{i-1}^n - u_{i-2}^n}{2(\Delta x)^3}$$

2) Au lieu de u dans $6u\frac{\partial u}{\partial x}$ on écrit la moyenne temporelle

$$u_i^n \rightarrow \frac{1}{3}(u_{i+1}^n + u_i^n + u_{i-1}^n)$$

3) On obtient l'équation discretisée

$$\frac{u_i^{n+1} - u_i^{n-1}}{2\Delta t} + 2(u_{i+1}^n + u_i^n + u_{i-1}^n) \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} + \frac{u_{i+2}^n - 2u_{i+1}^n + 2u_{i-1}^n - u_{i-2}^n}{2(\Delta x)^3} = 0$$

17.3.2 Résolution

- On a obtenu le schéma explicite:

$$\left\{ \begin{array}{l} u_i^{n+1} = u_i^{n-1} - \frac{2\Delta t}{\Delta x}(u_{i-1}^n + u_{i+1}^n + u_{i-1}^n)(u_{i+1}^n - u_{i-1}^n) - \frac{\Delta t}{(\Delta x)^3}(u_{i+2}^n - 2u_{i+1}^n + 2u_{i-1}^n - u_{i-2}^n) \\ u_i^0 = g(x) = \frac{v}{2ch^2(\sqrt{v}/2\Delta x i)} \end{array} \right.$$

- Introduisons une notation pour simplifier l'écriture:

$$u_i^{n+1} = u_i^{n-1} + F(u_{i+2}^n, u_{i+1}^n, u_{i-1}^n, u_{i-2}^n)$$

$$F = -\frac{2\Delta t}{\Delta x}(u_{i-1}^n + u_{i+1}^n - u_{i-1}^n)(u_{i+1}^n - u_{i-1}^n) - \frac{\Delta t}{(\Delta x)^3}(u_{i+2}^n - 2u_{i+1}^n + 2u_{i-1}^n - u_{i-2}^n)$$

- On commence par une boucle temporelle:

$$\text{For } (n = 1; \quad n \leq M; \quad n++) \{ \quad u_i^{n+1} = u_i^{n-1} + F(u_{i+2}^n, u_{i+1}^n, u_{i-1}^n, u_{i-2}^n) \quad \}$$

Il y a un problème: on ne connaît pas u_i^1

Pour calculer la valeur de u dans ce point fictif on peut approximer une seule fois (pour $n=0$) $\frac{\partial u}{\partial t}$ la dérivée avancée temporelle:

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

Pour l'équation de KdV on a

$$u_i^{n+1} = u_i^n - \frac{2\Delta t}{\Delta x}(u_{i-1}^n + u_{i+1}^n - u_{i-1}^n)(u_{i+1}^n - u_{i-1}^n) - \frac{\Delta t}{(\Delta x)^3}(u_{i+2}^n - 2u_{i+1}^n + 2u_{i-1}^n - u_{i-2}^n)$$

Prenons $n = 0$

$$u_i^1 = u_i^0 + \frac{2\Delta t}{\Delta x}(u_{i-1}^0 + u_{i+1}^0 - u_{i-1}^0)(u_{i+1}^0 - u_{i-1}^0) + \frac{\Delta t}{(\Delta x)^3}(u_{i+2}^0 - 2u_{i+1}^0 + 2u_{i-1}^0 - u_{i-2}^0)$$

- L'équation KdV est une équation nonlinéaire hyperbolique qui décrit la propagation des solitons. Donc pour obtenir les solution analytiques sur $] - \infty, +\infty[$ on n'a pas besoin des conditions aux limites. Pour une résolution numérique on doit les imposer. Ces conditions aux limites artificielles s'appellent conditions aux limites numériques.

- On peut imposer

$$\begin{cases} u_0^n = 0 \\ u_{N+1}^n = 0 \end{cases}$$

- L'autre possibilité est d'imposer les conditions aux limites periodiques avec deux point fictifs:

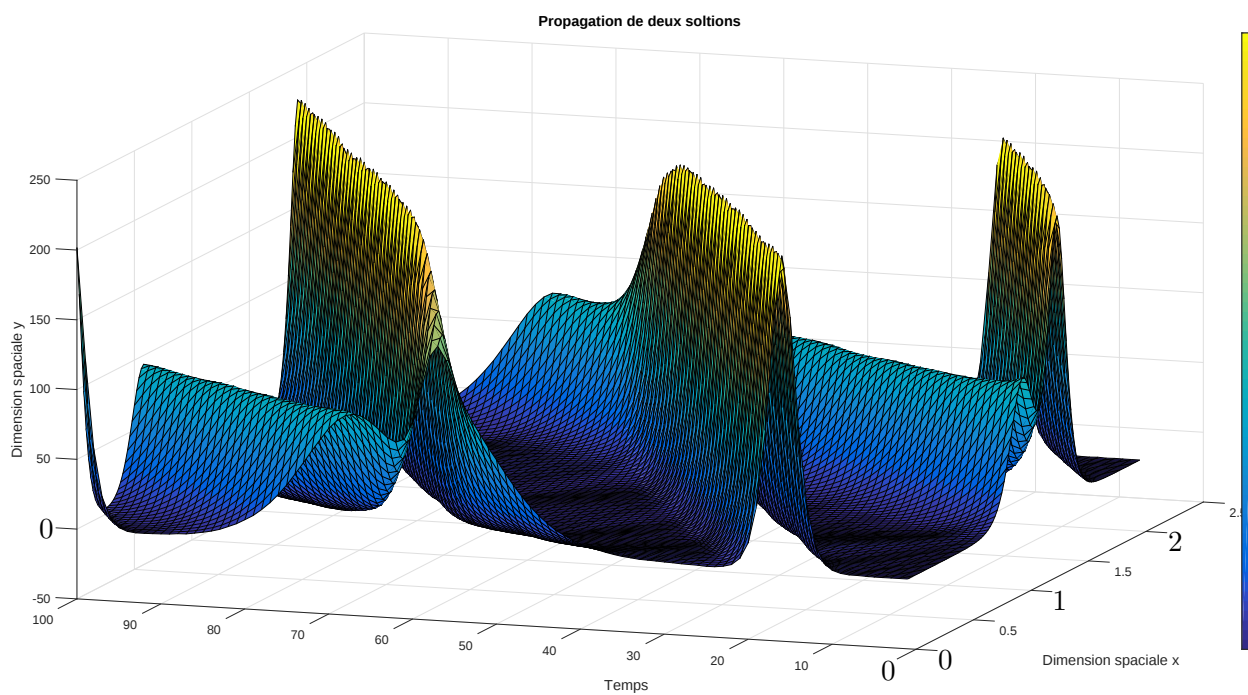
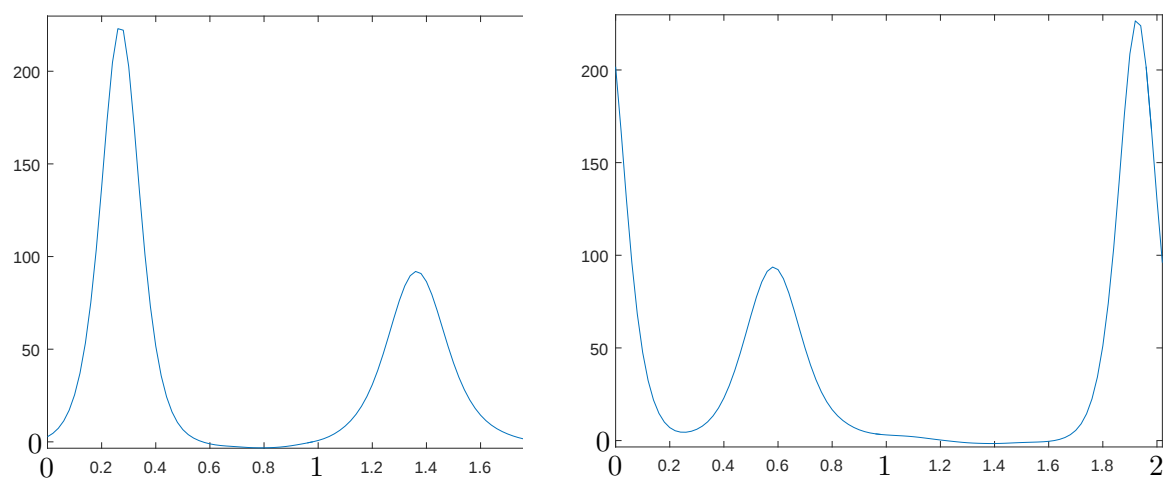
$$\begin{cases} u_0^n = u_{N-2}^n \\ u_1^n = u_{N-1}^n \\ u_N^n = u_2^n \\ u_{N+1}^n = u_3^n \end{cases}$$

On peut utiliser les données numériques pour simuler deux solitons avec les amplitudes différentes et par conséquent propagants avec les vitesses différentes:

$$\begin{cases} v_1 = 200, v_2 = 200, L = 2 \\ N = 100, M = 10000 \\ \Delta x = 2/100 = 1/50 \\ \Delta t = 10^{-6} \end{cases}$$

Le schéma est stable si

$$\Delta t \leq \frac{1}{4}(\Delta x)^3$$



Sur les figures on voit bien que le soliton avec l'amplitude plus élevée se propage avec la vitesse plus grande et double le première.

Bibliographie :

1. H. M. Antia. Numerical Methodes for Scietists and Engineers. Birkhauser.
2. M. Rappaz, M. Bellet, M. Deville. Numerical Modeling in Material Science and Engineering. Springer
3. J.W. Thomas. Numerical Partial Differential Equations: Finite Difference Methods. Springer, 1995
4. W.F. Ames. Numerical Methods for Partial Differential Equations, Nelson and Sons LTD. London, 1969
5. G.D. Smith, Numerical solution of PDE : Fintite difference methods, Clarendon Press, Oxford, 1978
6. J. Anderson. Computational Fluid dynamics. McGraw-Hill, Inc.
7. Tveito and Winther. Introduction to Partial Differential Equations: A Computational Approach. Springer
8. Taylor. Partial Differential Equations: Basic Theory. Springer
9. Gustafson and Wilcox. Analytical and Computational Methods of Advanced Engineering Mathematics. Springer
10. Grégoire Allaire. Cours sur EDP de l'Ecole Polytechnique.
11. Grégoire Allaire. Analyse numérique et optimisation. Les Editions Ellipses.