UFR DE MATHÉMATIQUES ET INFORMATIQUE

DESS Mathématiques et Statistique pour la Biologie

Notes de cours :

ANALYSE NUMÉRIQUE

Georges Koepfler 2002 - gk@math-info.univ-paris5.fr

Table des matières

Avertissement : ces notes sont un support et complément du cours magistral, des travaux dirigés et pratiques. Leur contenu n'est pas équivalent au cours enseigné, en particulier les examens et contrôles se réfèrent au cours enseigné uniquement.

Bibliographie.

Un livre très utile pour suivre ce cours est :

- P.G. Ciarlet, *Introduction à l'analyse matricielle et à l'optimisation*, Masson 1990.

Pour trouver des algorithmes en `C` et des références utiles, on pourra consulter

- W.H. Press, B.P. Flannery, S.A. Teukolsky et W.T. Vetterling, *Numerical Recipes in C*, Cambridge University Press 1988.

Quelques livres qui traitent de l'analyse numérique linéaire, et de ses applications, sont :

- J. Demmel, *Applied Numerical Linear Analysis*, SIAM 1997 ;
- P. Lascaux et J. Théodor, *Analyse numérique matricielle appliquée à l'art de l'ingénieur*, 2 tomes, Masson 1988.

Les livres suivants traitent des méthodes d'optimisation numérique :

- A. Auslender, *Optimisation. Méthodes Numériques*, Masson 1976 ;
- S. Boyd et L. Vandenberghe, *Convex Optimization*, notes de cours (livre à paraître en 2003) ;
- R. Fletcher, *Practical Methods of Optimization*, J. Wiley & Sons 1987 ;
- J.B. Hiriart-Urruty et C. Lemaréchal, *Convex Analysis and Minimization Algorithms*, Vol. I et II, Springer 1996.
- J. Nocedal et S.J. Wright, *Numerical Optimization*, Springer Series in Operations Research 1999.

Les livres suivants traitent des schémas aux différences finies pour les équations aux dérivées partielles :

- R. LeVeque, *Numerical Methods for Conservation Laws*, Birkhäuser 1992 ;
- J.C. Strikwerda, *Finite Difference Schemes and Partial Differential Equations*, Wadsworth & Brooks 1989 .

Pour l'étude de la méthode des éléments finis on pourra consulter :

- P.A. Raviart et J.M. Thomas, *Introduction à l'analyse numérique des équations aux dérivées partielles*, Masson 1992 ;
- B. Lucquin est O. Pironneau, *Introduction au calcul scientifique*, Masson 1996 .

Un receuil très complet de modèles mathématiques en biologie est le livre :

- J.D. Murray, *Mathematical Biology*, Springer 1993.

Chapitre 1

Introduction

Dans ce cours on va s'intéresser à la résolution numérique des problèmes suivants :

- Résolution de systèmes d'équations linéaires : $Ax = b$ où A est une matrice réelle carrée. On présentera des méthodes directes de factorisation de A et des méthodes itératives de résolution du système. Ces méthodes seront utilisées dans la suite du cours.
- Optimisation continue : on considère une fonction coût réelle que l'on veut minimiser. On va considérer la minimisation sans contraintes et avec contraintes (inégalités et égalités) et proposer divers algorithmes.
- Schémas aux différences finies : on veut déterminer une solution numérique d'une équation différentielle ordinaire ou équation aux dérivées partielles. On présente la technique des différences finies appliquée à divers types de problèmes.

Rappelons quelques points essentiels dont on doit tenir compte dans une résolution numérique d'un problème :

1. *Instabilité du problème* : Certains problèmes mathématiques sont instables : de petites perturbations des paramètres, dont dépend le problème, vont engendrer de grandes variations des solutions.
   Un exemple est la résolution de l'équation de la chaleur inversée $u_t = -\Delta u$; un autre la détermination des racines d'un polynôme.
   Comme les paramètres des modèles mathématiques viennent souvent d'expériences on utilisera autant que possible des modèles stables.
   Soit $s = \mathcal{A}[e]$ un algorithme ou problème qui, en fonction du paramètre e, produit le résultat s ; on s'intéresse à la stabilité de $\mathcal{A}$ et l'on veut contrôler l'erreur relative commise par une petite perturbation de e :

   $$\frac{|\tilde{s} - s|}{|s|} = \frac{|\mathcal{A}(e + \delta e) - \mathcal{A}(e)|}{|\mathcal{A}(e)|} \leq c(\mathcal{A}) \frac{|\delta e|}{|e|} .$$

   On appelle $c(\mathcal{A})$ le *conditionnement du problème* $\mathcal{A}$ (en choisisant la meilleure borne possible dans l'inégalité pour obtenir $c(\mathcal{A})$).

2. *Erreurs d'approximation* : Ce type d'erreur apparait lorsque l'on remplace un problème «continue» par un problème «discret». Quand le pas de discrétisation tend vers zéro la formulation discrète doit tendre vers la formulation continue du problème.
   Les formules de quadrature pour calculer des intégrales et les schémas aux différences

finies pour les équations aux dérivées partielles sont des exemples de formulations discrètes.

3. *Erreurs d'arrondi* : Les calculs numériques sur ordinateur se font avec une précision finie : la mémoire disponible étant finie on ne peut pas représenter les nombres réels qui ont un dévelopement décimal infini. On peut représenter $1/3$, mais pas son écriture décimale, on ne pourra représenter π que par un nombre fini de décimales. Un représentation fréquente est celle en *virgule flottante* des nombres réels :
$f = (-1)^s \cdot 0, d_{-1}d_{-2}\dots d_{-r} \cdot b^j$, $m \le j \le M$ et $0 \le d_{-i} \le b-1$;
où s est le signe, $d_{-1}d_{-2}\cdots d_{-r}$ la mantisse, b la base et r est le nombre de chiffres significatifs. La *précision machine* pour un flottant de mantisse r est b^{1-r} , c'est la distance entre 1 et le nombre flottant immédiatement plus grand $1 + b^{1-r}$. Comme on ne dispose que d'un nombre fini de réels le resultat des opérations arithmétiques de base $(+, -, \times, /)$ n'est pas nécessairement représentable : on est obligé d'arrondir.

   L'arithmétque IEEE, utilisée sur les machines sous Unix et Linux, définit des nombres flottants en base 2 : en simple précision de 32 bits (type `float` en C) et double précision de 64 bits (type `double` en C). En simple précision on utilise un bit pour s, 1 octet pour l'exposant (signé) j , *i.e.* $j \in \{-127, \dots, +128\}$, et 23 bits pour la mantisse. Si l'on suppose que la représentation est normalisée, *i.e.* $d_{-1} \neq 0$, on gagne un bit significatif et un flottant en simple précision s'écrit $f = (-1)^s\, 1, d_{-1}d_{-2}\dots d_{-r}\, 2^j$. dans ce cas la précision machine est de 2^{-23}, c.-à-d. approximativement 10^{-7}, et les flottants normalisés positifs vont de 10^{-38} à 10^{+38} .

   En plus des résultats, les modules arithmétiques envoient des messages qui indiquent la nature du résultat de l'opération : si ce n'est pas un nombre (`NaN` $=$ *NotANumber* $= \infty - \infty = \sqrt{-1} = 0/0 \dots$) ou un nombre trop grand, resp. petit, et qui ne peut être représenté.

4. *Complexité des calculs* : Pour la plupart des applications on veut obtenir un résultat en un temps raisonnable, il est donc important de pouvoir estimer le temps que l'algorithme va utiliser pour résoudre le problème. Pour cela, on compte le nombre d'opérations flottantes (angl. *flops*) en fonction de la taille du problème. Souvent il suffit d'avoir un ordre de grandeur : si N est la taille du problème (p.ex. nombre d'inconnues) on peut avoir des algorithmes en $O(N)$, $O(N \ln N)$, $O(N^2)$, $O(N!) \dots$

Pour illustrer les problèmes cités en haut on va présenter quatre exemples où ces notions jouent un rôle important :

Exemple 1 : Évaluation d'un polynôme proche d'une racine multiple.

Soit $p(x) = \sum_{i=0}^{d} a_i\, x^i$ un polynôme à coefficients réels avec $a_d \in \mathbb{R}^*$.

Si l'on connaît les racines du polynôme on peut le factoriser $p(x) = a_d \prod_{i=1}^{d} (x - r_i)$.

On peut aussi associer à p sa *matrice compagne*

$$\begin{pmatrix} -\frac{a_{d-1}}{a_d} & -\frac{a_{d-2}}{a_d} & & -\frac{a_2}{a_d} & -\frac{a_1}{a_d} & -\frac{a_0}{a_d} \\ 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & & 0 & 0 & 0 \\ \vdots & & \ddots & & \vdots & \vdots \\ 0 & 0 & & 1 & 0 & 0 \\ 0 & 0 & \dots & 0 & 1 & 0 \end{pmatrix},$$

dont les valeurs propres sont les racines de p.

Si l'on veut calculer $p(x)$ en utilisant la première représentation, on doit effectuer de l'ordre de $2d$ opérations élémentaires ($+$ et $\times$) et $d-1$ appels à la fonction puissance ; si l'on connaît les racines, il suffit de $2d$ opérations élémentaires. Mais comme on connaît rarement une factorisation du polynôme on utilise l'*algorithme de* HORNER pour calculer $p(x)$:

ALGORITHME 1.1 (HORNER)
Évaluation d'un polynôme en x à partir de ses coefficients $a_0, a_1, \dots, a_d$.

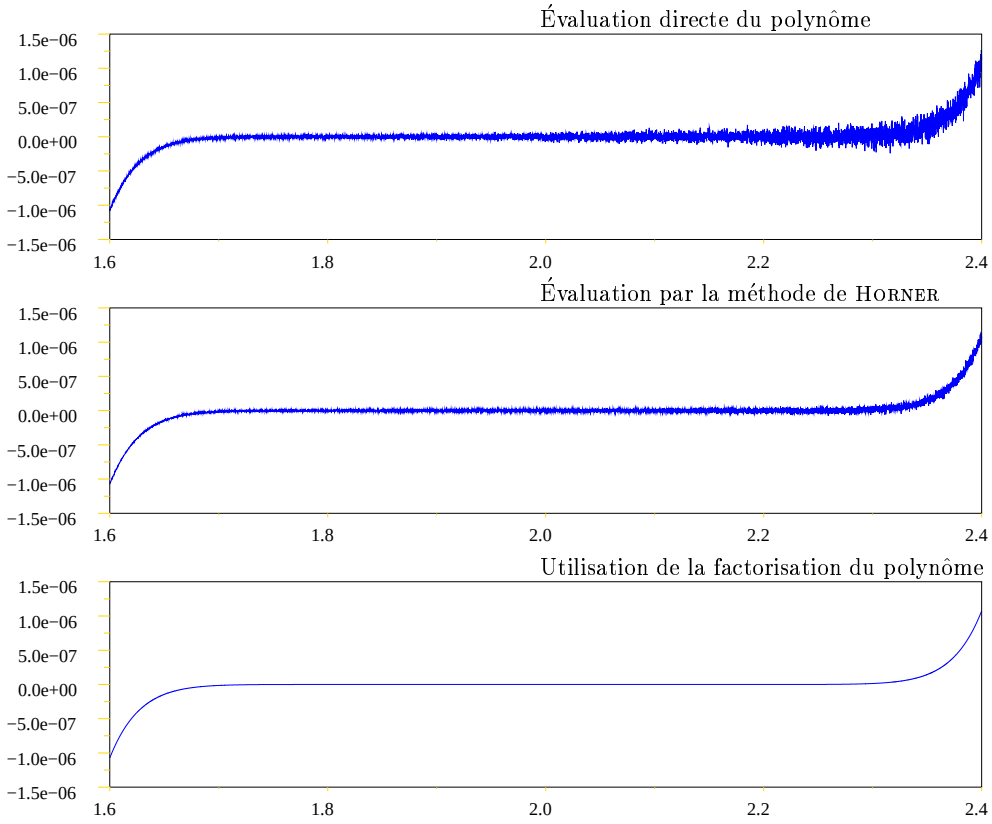
$y = a_d$
pour *i=d-1* ***à*** *0*
  $y = x \cdot y + a_i$

Cet algorithme permet d'évaluer $p(x)$ à partir des coefficients a_i en en $2d$ opérations élémentaires. On se propose de calculer et représenter le polynôme suivant :

$$\begin{aligned} p(x) &= x^{15} - 30x^{14} + 420x^{13} - 3640x^{12} + 21840x^{11} - 96096x^{10} \\ &\quad +320320x^9 - 823680x^8 + 1647360x^7 - 2562560x^6 + 3075072x^5 \\ &\quad -2795520x^4 + 1863680x^3 - 860160x^2 + 245760x - 32768 \\ &= (x-2)^{15} . \end{aligned}$$

Les résultats sont représentés à la figure suivante, on a calculé p sur l'intervalle $[1.6, 2.4]$ avec un pas de 10^{-4} .

Les oscillations qui apparaissent dans le graphe de p rendent difficiles la détection de la racine de p par une méthode comme celle de la dichotomie par exemple.
Pour calculer de façon rapide et précise les valeurs d'un polynôme il vaut donc mieux utiliser sa forme factorisée, or pour cela il faut trouver tous les zéros de l'équation polynomiale $p(x) = 0$, ce qui est instable comme l'on verra, ou il faut trouver les valeurs propres de la matrice compagne, ce qui est un autre problème, non moins difficile.



Exemple 2 : Détermination des racines d'un polynôme perturbé.

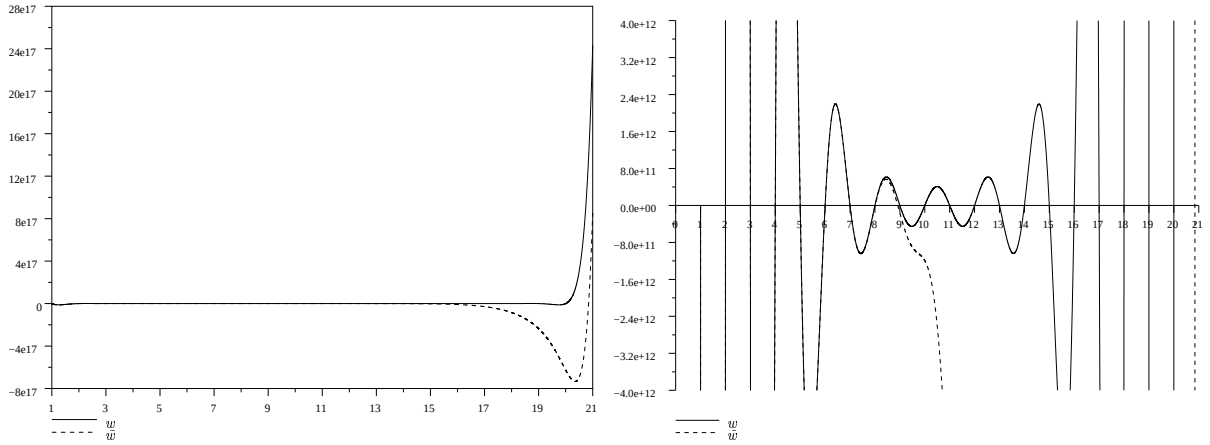
Pour illustrer l'instabilité de la résolution d'une équation polynomiale par rapport aux coefficients du polynôme on va utiliser l'exemple de proposé par Wilkinson (1963) :

$$
\begin{aligned}
w(x) &= \prod_{i=1}^{20}(x-i) \\
&= x^{20} - 210\,x^{19} + 20615\,x^{18} - 1256850\,x^{17} + 53327946\,x^{16} - 1.672\cdot 10^{9}\,x^{15} \\
&\quad +4.017\cdot 10^{10}\,x^{14} - 7.561\cdot 10^{11}\,x^{13} + 1.131\cdot 10^{13}\,x^{12} - 1.356\cdot 10^{14}\,x^{11} \\
&\quad +1.308\cdot 10^{15}\,x^{10} - 1.014\cdot 10^{16}\,x^{9} + 6.303\cdot 10^{16}\,x^{8} - 3.113\cdot 10^{17}\,x^{7} \\
&\quad +1.207\cdot 10^{18}\,x^{6} - 3.600\cdot 10^{18}\,x^{5} + 8.038\cdot 10^{18}\,x^{4} - 1.287\cdot 10^{19}\,x^{3} \\
&\quad +1.380\cdot 10^{19}\,x^{2} - 8.753\cdot 10^{18}\,x + 20!\cdot 10^{18}
\end{aligned}
$$

On considère ensuite le polynôme perturbé $\tilde{w}(x) = w(x) - 2^{-23}\,x^{19}$ et, en faisant des calculs numériques en haute précision, on obtient les racines suivantes pour $\tilde{w}$:

1, 00000 0000	10, 09526 6145 + 0, 64350 0904 i
2, 00000 0000	10, 09526 6145 − 0, 64350 0904 i
3, 00000 0000	11, 79363 3881 + 1, 65232 9728 i
4, 00000 0000	11, 79363 3881 − 1, 65232 9728 i
4, 99999 9928	13, 99235 8137 + 2, 51883 0070 i
6, 00000 6944	13, 99235 8137 − 2, 51883 0070 i
6, 99969 7234	16, 73073 7466 + 2, 81262 4894 i
8, 00726 7603	16, 73073 7466 − 2, 81262 4894 i
8, 91725 0249	19, 50243 9400 + 1, 94033 0347 i
20, 84690 8101	19, 50243 9400 − 1, 94033 0347 i

On obtient 5 racines complexes conjuguées de partie imaginaire non négligeable. Cet exemple célèbre montre comment une petite perturbation d'un coefficient change de façon profonde l'ensemble des solutions d'une équation polynomiale. Les figures suivantes donnent les graphes de w et $\tilde{w}$.



Exemple 3 : Calcul de la trajectoire de la fusée ARIANE 5

Le 4 juin 1996, le premier vol de la fusée ARIANE 5 s'est terminé par l'explosion de l'engin à peine 50s après le décollage. Dans le rapport ESA/CNES [1], établi suite à cet incident, on peut lire que les causes de l'échec sont dues à un signal d'*overflow* mal interprété par l'ordinateur de bord.

En effet, les deux systèmes de référence inertiels ont arrêté de fonctionner à cause d'une erreur lors d'une conversion d'un nombre flottants en 64 bit vers un entier signé en 16 bits. Le nombre à convertir ayant une valeur trop grande pour être représenté en 16 bits, l'opération s'est terminé par un signal d'*overflow*. Le premier système de référence inertiel a arrêté de fonctionner et le second à pris le relais, et la même erreur c'est produit. L'ordinateur de bord a cette fois pris en compte le signal d'*overflow*, mais en l'interprétant comme étant une donnée. Il y a eu un changement de trajectoire qui a mené la fusée vers une forte déviation, ceci a causé la désintégration du lanceur.

Le rapport constate que le programme en cause n'était plus utilisé après le décollage, de plus le code avait été transféré sans changement du système ARIANE 4 vers ARIANE 5, sans tenir compte du fait que sur le nouveau lanceur les paramètres prenaient d'autres valeurs.

Conclusion de ce feu d'artifice fort coûteux : il est important de connaître des fourchettes pour les valeurs d'entrée d'un programme et de protéger les logiciels contre les résultats erronés.

Exemple 4 : Multiplication de matrices

Soient A, B et C des matrices rectangulaires, de dimensions respectives (n, m), (m, p) et (p, q). Combien d'opérations sont nécessaires pour calculer ABC ? Comme $ABC = (AB)C = A(BC)$, on a deux possibilités d'évaluation.

Le calcul de $(AB)C$ nécessite de l'ordre de $2np(m + q)$ opérations, tandis que le calcul de $A(BC)$ se fait en $2mq(n + p)$ opérations. Si l'on prend par exemple $n = p = 10$ et

[1] *ARIANE 5 : Flight 501 Failure*, rapport sous la direction de J.L. Lions, juillet 1996.

$m = q = 100$ on trouve $4 \cdot 10^4$ et $4 \cdot 10^5$. La multiplication des matrices rectangulaires est bien distributive, mais le nombre d'opérations nécessaires peut varier de façon importante.

Soit $n = m = p$, alors A et B sont des matrices carrées et le coût de leur multiplication est $O(n^3)$, or il existe un algorithme qui permet de réduire ce coût à $O(n^{\log_2 7})$. La méthode est basé sur une réécriture de le multiplication de deux matrices $(2,2)$, si les matrices sont carrées d'ordre n, avec $n = 2^r$, on applique l'algorithme de manière récursive.

ALGORITHME 1.2 (STRASSEN)
Calcul du produit des matrices carrées d'ordre $n = 2^r$: $C = AB$
$C = STRASSEN(A, B, n)$
si $n = 1$ ***alors***
$C = A \cdot B$ // *cas scalaire* //
sinon
$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$ *et* $B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$ // *où les* A_{ij} *et* B_{ij} *sont* $(n/2, n/2)$ //
$Q_1 = STRASSEN(A_{11} + A_{22}, B_{11} + B_{22}, n/2)$
$Q_2 = STRASSEN(A_{21} + A_{22}, B_{11}, n/2)$
$Q_3 = STRASSEN(A_{11}, B_{12} - B_{22}, n/2)$
$Q_4 = STRASSEN(A_{22}, -B_{11} + B_{21}, n/2)$
$Q_5 = STRASSEN(A_{11} + A_{12}, B_{22}, n/2)$
$Q_6 = STRASSEN(-A_{11} + A_{21}, B_{11} + B_{12}, n/2)$
$Q_7 = STRASSEN(A_{12} - A_{22}, B_{21} + B_{22}, n/2)$
$C_{11} = Q_1 + Q_4 - Q_5 + Q_7$
$C_{12} = Q_3 + Q_5$
$C_{21} = Q_2 + Q_4$
$C_{22} = Q_1 - Q_2 + Q_3 + Q_6$
$C = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix}$

En pratique il faut n assez grand pour obtenir un gain satisfaisant avec la méthode de STRASSEN ($\log_2 7 = 2,8073$). Mais des méthodes simples de multiplication par blocs sont par exemple utilisées dans les librairies BLAS (angl. *Basic Linear Algebra Subroutines*). Nous allons en décrire le principe en quelques mots.

Il ne suffit pas d'utiliser la vitesse de calcul et la mémoire centrale des processeurs pour obtenir des programmes rapides : supposons que l'on veut multiplier deux grandes matrices dont la taille nécessite une sauvegarde dans une mémoire qui est d'accès lent. Il est clair que la plus grande partie du temps est utilisé pour transférer des données. Les librairies tels que BLAS implémentent des multiplications de matrices par blocs en tenant compte de l'architecture du processeur. Les blocs transférés de la mémoire lente sont de la taille des mémoires d'accès rapide, on fait moins de transferts lents et on passe proportionnellement plus de temps sur les calculs.

Les librairies LAPACK, CLAPACK et ScaLAPACK (`http://www.netlib.org`) utilisent cette approche.
On peut noter aussi que des logiciels tels que `Scilab` [2] ou `Matlab`©, utilisent des opérations par blocs pour accroître leur performances (et il est fortement recommandé d'en faire usage en TP !).

[2] © INRIA, logiciel disponible à `http://www-rocq.inria.fr/scilab/`

Chapitre 2

Algèbre linéaire

Dans la première partie de ce chapitre nous allons proposer des méthodes de résolution directes de l'équation $Ax = b$, où A est une matrice carrée d'ordre n inversible. Dans la seconde partie on s'intéresse au cas où A est rectangulaire, on cherchera x minimisant $\|Ax - b\|_2$. La dernière section présente des méthodes itératives pour le cas d'une grande matrice A carrée d'ordre n inversible.

2.1 Résolution de systèmes linéaires par des méthodes directes

On s'intéresse à l'équation $Ax = b$. En un premier temps nous allons étudier la stabilité de ce problème. Pour cela on considère le problème perturbé
$(A + \delta A)\tilde{x} = b + \delta b$, et on pose $\delta x = \tilde{x} - x$.
Par soustraction on trouve $\delta x = A^{-1}(\delta b - \delta A\tilde{x})$ et

$$\frac{\|\delta x\|}{\|\tilde{x}\|} \leq \|A^{-1}\| \, \|A\| \left(\frac{\|\delta A\|}{\|A\|} + \frac{\|\delta b\|}{\|A\|\|\tilde{x}\|} \right) .$$

Le nombre $c(A) = \|A^{-1}\| \, \|A\|$ est le *conditionnement* de la matrice A pour le problème d'inversion (et pour la norme $\|.\|$). Si de plus $\|A^{-1}\| \, \|\delta A\| < 1$

$$\frac{\|\delta x\|}{\|x\|} \leq \frac{c(A)}{1 - c(A)\frac{\|\delta A\|}{\|A\|}} \left(\frac{\|\delta A\|}{\|A\|} + \frac{\|\delta b\|}{\|b\|} \right) .$$

Le théorème suivant donne une caractérisation intéressante du nombre $c(A)$.

THÉORÈME 2.1.1
Soit A une matrice inversible, alors

$$\min \left\{ \frac{\|\delta A\|_2}{\|A\|_2} \Big/ \, A + \delta A \text{ singulière} \right\} = \frac{1}{\|A^{-1}\|_2 \|A\|_2} = \frac{1}{c_2(A)} .$$

Donc la distance relative de A à la matrice singulière la plus proche est l'inverse du conditionnement de A en $\|.\|_2$.

2.1.1 Factorisation LU de matrices

L'algorithme de résolution est basé sur le

THÉORÈME 2.1.2 (ÉLIMINATION DE GAUSS)
Soit A une matrice inversible, alors il existe des matrices de permutations P_1 et P_2, une matrice L triangulaire inférieure avec des 1 sur la diagonale et une matrice U triangulaire supérieure inversible telles que

$$P_1 A P_2 = L U .$$

En fait une seule matrice de permutation est nécessaire, de plus
(i) on peut choisir $P_2 = I$ et P_1 tel que a_{11} soit l'élément de valeur absolue maximale dans la première colonne, c'est l'algorithme d'élimination de GAUSS *avec pivot partiel.*
(ii) on peut choisir P_1 et P_2 de façon à ce que a_{11} soit l'élément de valeur absolue maximale dans toute la matrice, c'est l'algorithme d'élimination de GAUSS *avec pivot total.*

Remarque : Si $\sigma : \{1, \ldots, n\} \to \{1, \ldots, n\}$ est une permutation, alors la matrice associée P_σ est définie par $(P_\sigma)_{ij} = \delta_{i\sigma(j)}$, où δ_{ij} est le symbole de KRONECKER. $P_\sigma A$ réordonne les lignes de A, AP_σ réordonne les colonnes de A et $P_\sigma A P_{\sigma'}$ réordonne tout.

ALGORITHME 2.1
Factorisation LU avec pivot.

pour $i = 1$ **à** $n - 1$
  Appliquer les permutations à A, L et U telles que $a_{ii} \neq 0$.
  pour $j = i + 1$ **à** n
    $l_{ji} = a_{ji}/a_{ii}$
  pour $j = i$ **à** n
    $u_{ij} = a_{ij}$
  pour $j = i + 1$ **à** n
    pour $k = i + 1$ **à** n
      $a_{jk} = a_{jk} - l_{ji} \cdot u_{ik}$

L'algorithme de factorisation LU est utilisé pour résoudre le système linéaire $Ax = b$.

ALGORITHME 2.2
Résoudre $Ax = b$ en utilisant l'élimination de GAUSS *avec pivot.*

Factoriser A en $A = P_1 L U P_2$.
Par permutation : $LUP_2 x = P_1^t b$.
Résoudre un système triangulaire inférieur : $UP_2 x = L^{-1}(P_1^t b)$.
Résoudre un système triangulaire supérieur : $P_2 x = U^{-1}(L^{-1}(P_1^t b))$.
Par permutation : $x = P_2^t(U^{-1}(L^{-1}(P_1^t b)))$.

Coût : La résolution du système $Ax = b$ revient donc à la résolution de deux systèmes triangulaires qui demandent peu d'opérations, $O(n^2)$. La décomposition LU est d'ordre $O(\frac{2}{3}n^3)$ et le coût total de l'élimination de GAUSS pour résoudre un système linéaire est donc $O(\frac{2}{3}n^3)$.
On n'a pas tenu compte des mouvements de données nécessaires pour le pivot par exemple, des implémentations effectives de l'algorithme de factorisation LU optimisent aussi cette partie de la méthode (*cf.* implémentations BLAS).

Pour pouvoir garantir la précision des résultats on aimerait avoir des bornes sur l'erreur relative, or pour cela il faut déterminer $\|\delta A\|/\|A\|$, $\|\delta b\|/\|b\|$ et $c(A) = \|A\|\|A^{-1}\|$.
Pour obtenir $\|\delta A\|$ on utilise le théorème suivant

THÉORÈME 2.1.3
Soit $\tilde{x}$ la solution calculée et $r = A\tilde{x} - b$ le résidu. Il existe une matrice δA tel que $\|\delta A\| = \frac{\|r\|}{\|\tilde{x}\|}$ et $(A + \delta A)\tilde{x} = b$. Il n'y pas de matrice de norme plus petite vérifiant $(A + \delta A)\tilde{x} = b$.

Pour déterminer $c(A)$ on doit calculer $\|A^{-1}\|$, c.-à-d. A^{-1} de façon explicite, ce qui est trop coûteux. On préfère utiliser un bon estimateur de $\|A^{-1}\|$. L'algorithme de HAGER que nous allons présenter donne une borne inférieure de la norme $\|.\|_1$ d'une matrice B, il est basé sur une maximisation de la fonction $f(x) = \|Bx\|_1 = \sum_{i=1}^{n} \left| \sum_{j=1}^{n} b_{ij} x_j \right|$.

ALGORITHME 2.3 (HAGER)
Estimation de $\|B\|_1$.
Choisir $x^{(0)}$ tel que $\|x^{(0)}\|_1 = 1$, par ex. $(x^{(0)})_i = 1/n$, $k = 0$
boucle
$y = Bx^{(k)}$
$s = \text{signe}(y)$
$g = B^t s$
si $\|g\|_\infty \leq g.x^{(k)}$ ***alors***
retourner $\|y\|_1$
sinon
$k = k + 1$
$x^{(k)} = e_j$ *pour* j *tel que* $|g_j| = \|g\|_\infty$

Si $\|y\|_1$ est retourné par l'algorithme, alors c'est un maximum local de f ;
sinon on montre qu'en début de boucle le nouveau choix de x vérifie $f(x_{nouv.}) > f(x_{anc.})$.
L'algorithme de HAGER est un exemple d'algorithme qui remonte dans la direction du gradient (cf. optimisation convexe).
On applique l'algorithme à la matrice $(A^{-1})^t$ pour estimer $\|A^{-1}\|_\infty$. Noter qu'il suffit d'appliquer A^{-1} et $(A^{-1})^t$ et d'utiliser la factorisation LU de A.

2.1.2 Matrices réelles définies positives

Dans beaucoup d'applications la matrice A est symétrique définie positive, dans ce cas on économise de l'espace mémoire en ne stockant que $n(n+1)/2$ termes. De plus il existe un algorithme de factorisation en $O(\frac{1}{3}n^3)$. Le théorème suivant regroupe quelques résultats utiles.

THÉORÈME 2.1.4

1. *Si $A = (a_{ij})_{1\leq i,j\leq n}$ est une matrice symétrique définie positive, alors toute sous-matrice $M_m = (a_{ij})_{1\leq i,j\leq m}$ est symétrique définie positive.*
2. *Soit B une matrice inversible, alors A est symétrique définie positive si et seulement si B^tAB est symétrique définie positive.*
3. *Une matrice A est symétrique définie positive si et seulement si $A^t = A$ et toutes ses valeurs propres sont strictement positives.*
4. *Si A est symétrique définie positive, alors $\max_{1\leq i,j\leq n} |a_{ij}| = \max_{1\leq i\leq n} a_{ii} > 0$.*
5. *Une matrice A est symétrique définie positive si et seulement si il existe une matrice L triangulaire inférieure inversible telle que $A = L\,L^t$.*
   C'est la factorisation de CHOLESKY d'une matrice.
6. *Si A est une matrice symétrique définie positive de valeurs propres $0 < \lambda_1 \leq \cdots \leq \lambda_n$, alors pour tout $x \in \mathbb{R}^n \setminus \{0\}$ on a*

$$\frac{(x^tx)^2}{(x^tAx)(x^tA^{-1}x)} \geq \frac{4\lambda_1\lambda_n}{(\lambda_1+\lambda_n)^2} = 1 - \left(\frac{c_2(A)-1}{c_2(A)+1}\right)^2 ,$$

   avec $c_2(A) = \lambda_n/\lambda_1$. C'est l'inégalité de KANTOROVICH.

ALGORITHME 2.4 (CHOLESKY)
Factorisation de CHOLESKY *d'une matrice symétrique définie positive.*

pour $j = 1$ ***à*** n

$$l_{jj} = \left(a_{jj} - \sum_{k=1}^{j-1} l_{jk}^2\right)^{\frac{1}{2}}$$

pour $i = j+1$ ***à*** n

$$l_{ij} = \frac{1}{l_{jj}}\left(a_{ij} - \sum_{k=1}^{j-1} l_{ik}l_{jk}\right)$$

Pour résoudre $Ax = LL^tx = b$ on doit résoudre deux systèmes triangulaires.

2.1.3 Autres types de matrices

Dans les applications on rencontre souvent d'autres types de matrices : des matrices bande, tridiagonales, tridiagonales par blocs, creuses ...
Avec des méthodes adaptées on peut souvent améliorer les performances des algorithmes de factorisation, que ce soit au niveau de l'espace mémoire, du nombre d'opérations ou du transfert de données.
La plupart des librairies scientifiques (*e.g.* LAPACK, CLAPACK) ou logiciels (*e.g.* `Scilab`, `Matlab`) proposent des algorithmes et représentations pour les types de matrices les plus fréquents (*e.g.* matrices creuses).

2.2 Résolution de systèmes linéaires au sens des moindres carrées

Soit A une matrice de taille (m,n), un vecteur $x \in \mathbb{R}^n$ est solution du système $Ax = b$ au sens des moindres carrées si x minimise $\|Ax - b\|_2$.

Si $m = n$ et A inversible il existe une solution unique $x = A^{-1}b$; si $m > n$ on a un problème *surdéterminé*, en général il n'existe pas de solution de l'équation $Ax = b$; si $m < n$ on a un problème *sous-déterminé*, on peut avoir une infinité de solutions vérifiant le système linéaire.

Nous allons énoncer quelque résultats d'algèbre linéaire qui permettent de résoudre ce problème, les algorithmes de factorisation présentés font partie des librairies mathématiques standard.

2.2.1 Matrices de Householder

DÉFINITION 2.2.1
On appelle matrice de HOUSEHOLDER *toute matrice de la forme*

$$H = I_m - 2v\,v^t$$

où $v \in \mathbb{R}^m$ *et* $\|v\|_2 = 1$.

On montre que H est une matrice symétrique et que $H\,H^t = I$; une matrice de HOUSEHOLDER correspond à une symétrie par rapport à l'hyperplan perpendiculaire à v .

Soit $x \in \mathbb{R}^m$ tel que $\alpha^2 = x_k^2 + \ldots + x_m^2 > 0$, et soit $v \in \mathbb{R}^m$ tel que

$$v_i = \begin{cases} 0 & \text{pour } 1 \le i \le k-1 \\ x_k + \text{signe}(x_k)|\alpha| & \text{pour } i = k \\ x_i & \text{pour } k+1 \le i \le m \end{cases} ,$$

alors si $H = I - 2v\,v^t$, on a $Hx = x - v$ et $(Hx)_i = 0$ pour $k+1 \le i \le m$.

Soit A une matrice de taille (m, n) , la *méthode de* HOUSEHOLDER de résolution du système linéaire $Ax = b$ équivaut à construire des matrices de HOUSEHOLDER H_1 ,..., H_n telles que la matrice $H_n H_{n-1} \cdots H_1 A$ soit triangulaire supérieure.
Il suffit ensuite de résoudre un système triangulaire supérieur par la méthode de la remontée.
Si $m = n$ il suffit d'appliquer $n - 1$ transformation pour obtenir une matrice triangulaire supérieure, si $m > n$ on annule $m - n$ lignes de la matrice A .

Les matrices H étant orthogonales on a $\|Ax - b\|_2 = \|HAx - Hb\|_2$,
le résidu du système triangulaire obtenu par la méthode de HOUSEHOLDER est identique en $\|.\|_2$ au résidu du problème initial.

2.2.2 Factorisation QR

THÉORÈME 2.2.1 (FACTORISATION QR)
Soit A une matrice de taille (m, n) avec $m \ge n$ et telle que $\text{rang}(A) = n$. *Alors il existe une unique matrice Q de taille (m, n) telle que $Q^tQ = I_n$; une unique matrice R de taille (n, n) triangulaire supérieure dont les éléments diagonaux sont positifs, $r_{ii} > 0$, telles que $A = QR$.*

Grâce à la factorisation QR de A on a :

$$Ax - b = QRx - b = Q(Rx - Q^tb) - (I_m - QQ^t)b .$$

Les vecteurs $Q(Rx - Q^t b)$ et $(I_m - QQ^t)b$ étant orthogonaux on obtient :

$$\|Ax - b\|_2^2 = \left\|Q(Rx - Q^t b)\right\|_2^2 + \left\|(I_m - QQ^t)b\right\|_2^2 ,$$

cette somme de carrées est minimale pour $x = R^{-1}Q^t b$.

On peut utiliser les matrices de HOUSEHOLDER pour obtenir une factorisation QR.
En effet $H_n \cdots H_1 A = A_n$ est une matrice (m,n) dont les $m-n$ dernières lignes sont nulles. Grâce aux propriétés des H_i on a $A = H_1 \cdots H_n A_n = QR$ où
$Q = (H_1 \cdots H_n)(1 \leq i \leq m; 1 \leq j \leq n)$, $Q^t Q = I_n$,
et $R = A_n(1 \leq i \leq n; 1 \leq j \leq n)$ triangulaire supérieure.

2.2.3 Décomposition en valeurs singulières

La décomposition en valeur singulières d'une matrice est souvent appelé décomposition SVD (angl. *Singular Values Decomposition*) .

THÉORÈME 2.2.2 (DÉCOMPOSITION SVD)
Soit A une matrice de taille (m,n) avec $m \geq n$.
a) On peut écrire $A = U\Sigma V^t$, où U est une matrice de taille (m,n) telle que $U^t U = I_n$; Σ est une matrice diagonale avec $\sigma_1 \geq \ldots \geq \sigma_n \geq 0$; V est de taille (n,n) et vérifie $V^t V = I_n$.
Les σ_i sont les valeurs singulières de A .
b) Les valeurs propres de la matrice $A^t A$ sont σ_i^2 , les vecteurs propres sont les vecteurs colonne $V_1, \ldots, V_n$.
c) Les valeur propres de la matrice AA^t sont σ_i^2 et $m-n$ zéros. Les vecteurs propres sont les $U_1, \ldots, U_n$ et $m-n$ vecteurs propres associés à 0 .
d) Si $m = n$ et A inversible, alors $\|A\|_2 = \sigma_1$, $\|A^{-1}\|_2 = 1/\sigma_n$ et $c_2(A) = \dfrac{\sigma_1}{\sigma_n}$.

Si A est une matrice de taille (m,n) avec $m \geq n$ et $\text{rang}(A) = n$, on montre que la solution du problème des moindres carrées est $x = V\Sigma^{-1}U^t b$.

DÉFINITION 2.2.2
Soit A une matrice de taille (m,n) avec $m \geq n$ et $\text{rang}(A) = n$.
On appelle matrice pseudo-inverse de MOORE-PENROSE *la matrice*

$$A^+ = R^{-1}Q^t = V\Sigma^{-1}U^t .$$

On peut donc exprimer x minimisant $\|Ax - b\|_2$ grâce à A^+ .

On peut généraliser la définition de A^+ à des matrices de rang $r < n$ comme suit :
soit $A = U\Sigma V^t$, on écrit

$$A = (U_1\, U_2) \begin{pmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{pmatrix} (V_1\, V_2)^t = U_1\, \Sigma_1\, V_1^t ,$$

où Σ_1 est de taille (r,r) inversible et les matrices U_1 et V_1 ont r colonnes. On pose $A^+ = V_1\Sigma_1^{-1}U_1^t = V \begin{pmatrix} \Sigma_1^{-1} & 0 \\ 0 & 0 \end{pmatrix} U^t$.
Dans ce cas on a encore $x = A^+ b$ qui minimise $\|Ax - b\|_2$.

2.3 Résolution de systèmes linéaires par des méthodes itératives

Les méthodes directes présentées jusqu'ici permettent de calculer la solution d'un système linéaire en un nombre fini de pas (solution exacte en ne tenant pas compte des erreurs d'arrondi). Par contre pour de grands systèmes ces méthodes demandent souvent trop d'espace mémoire et trop de calculs. On utilise alors des méthodes itératives où l'on arrête l'itération dès que la précision désirée est atteinte.

Avant de brièvement rappeler les méthodes les plus courantes, nous allons donner quelques résultats généraux.

Soit A carrée d'ordre n inversible , on appelle *décomposition* de A tout couple de matrices (M, N) où M est inversible, et tel que $A = M - N$.

Le système $Ax = b$ devient alors $x = M^{-1}Nx + M^{-1}b = Rx + r$ et la méthode itérative s'écrit

$$x^{(k+1)} = R \cdot x^{(k)} + r \ .$$

PROPOSITION 2.3.1
Soit (M, N) la décomposition associée à A et supposons que pour la norme matricielle subordonnée on a $\|R\| < 1$.
Alors pour tout vecteur $x^{(0)} \in \mathbb{R}^n$, la suite $x^{(k+1)} = R \cdot x^{(k)} + r$ est convergente.

PROPOSITION 2.3.2
La suite $(x^{(k)})$ définie par $x^{(k+1)} = R \cdot x^{(k)} + r$ converge vers la solution de $Ax = b$, pour tout $x^{(0)} \in \mathbb{R}^n$, si et seulement si $\rho(R) < 1$.

On appelle *vitesse de convergence* de la méthode le nombre $-\log_{10}(\rho(R))$.

Les méthodes que nous allons présenter sont basées sur des choix qui permettent d'inverser facilement M et d'obtenir une matrice R pour laquelle $\rho(R)$ est petit.

2.3.1 Méthode de Jacobi

On décompose A en $A = D - K$ où D est la diagonale de A et $-K$ contient les éléments hors diagonale. On note $R_J = D^{-1}K$.

ALGORITHME 2.5 (JACOBI)
Après permutations éventuelles pour rendre D inversible,
une itération de la méthode de JACOBI *s'écrit :*

pour $i = 1$ ***à*** n

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1, j \neq i}^{n} a_{ij} x_j^{(k)} \right)$$

PROPOSITION 2.3.3
Si la matrice A est à diagonale strictement dominante,
c.-à-d. pour tout $1 \leq i \leq n$: $\displaystyle\sum_{j=1, j \neq i}^{n} |a_{ij}| < |a_{ii}|$, alors la méthode de JACOBI *converge.*

2.3.2 Méthode de Gauss-Seidel

On écrit $A = (D - L) - U$ où D est la diagonale de A et $-L$, resp. $-U$, la matrice triangulaire strictement inférieure, resp. supérieure, de A. On note $R_{GS} = (D - L)^{-1}U$.

ALGORITHME 2.6 (GAUSS-SEIDEL)
Après permutations éventuelles, une itération de la méthode de GAUSS-SEIDEL *s'écrit :*

pour $i = 1$ ***à*** n

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right)$$

PROPOSITION 2.3.4
Si la matrice A est à diagonale strictement dominante, alors la méthode de GAUSS-SEIDEL *converge.*
Dans ce cas on a de plus $\|R_{GS}\|_\infty \leq \|R_J\|_\infty < 1$.

2.3.3 Méthode de relaxation

Le décomposition de A s'écrit $A = D - L - U$ et $M = \frac{1}{\omega}D - L$, $N = (\frac{1}{\omega} - 1)D + U$.
On note $R_{rel(\omega)} = (\frac{1}{\omega}D - L)^{-1}((\frac{1}{\omega} - 1)D + U)$.

ALGORITHME 2.7
Une itération de la méthode de relaxation de paramètre ω s'écrit :

pour $i = 1$ ***à*** n

$$x_i^{(k+1)} = (1 - \omega)x_i^{(k)} + \frac{\omega}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right)$$

THÉORÈME 2.3.1
On a $\rho(R_{rel(\omega)}) > |\omega - 1|$, une condition nécessaire de convergence de la méthode de relaxation est donc que $0 < \omega < 2$.

THÉORÈME 2.3.2
Soit A une matrice symétrique définie positive, alors $\rho(R_{rel(\omega)}) < 1$ pour $0 < \omega < 2$. Une condition suffisante de convergence de la méthode de relaxation est alors que $0 < \omega < 2$.
Pour $\omega = 1$ la méthode de GAUSS-SEIDEL *converge aussi.*

Chapitre 3

Optimisation continue

3.1 Définitions et rappels

3.1.1 Problème général d'optimisation continue

Soient f, g et h définies sur $\mathcal{D} \subset \mathbb{R}^n$ avec $\mathcal{D} = \mathbf{dom}\, f \cap \mathbf{dom}\, g \cap \mathbf{dom}\, h$
et $f : \mathcal{D} \to \mathbb{R}$, $g : \mathcal{D} \to \mathbb{R}^m$, $h : \mathcal{D} \to \mathbb{R}^p$.
Un *problème d'optimisation non linéaire* sur $\mathbb{R}^n$ se présente sous la *forme standard* :

$$\begin{cases} \text{Minimiser} & f(x) \\ \text{sous la condition} & x \in \mathfrak{C} = \{x \in \mathcal{D} \mid g(x) \leq 0\,, h(x) = 0\}. \end{cases} \tag{3.2}$$

La fonction (ou fonctionnelle) f est appelée *critère*, *coût*, *fonction-objectif* ou *fonction économique*; l'ensemble $\mathcal{D} \subset \mathbb{R}^n$ est l'ensemble des *paramètres* ou *variables d'état*; l'ensemble $\mathfrak{C} \subset \mathcal{D}$ est l'ensemble des *contraintes*, lorsque x se trouve dans $\mathfrak{C}$ on dit que x est *admissible* ou *réalisable*.

Lorsque $\mathfrak{C} = \mathcal{D}$ on a un problème *sans contraintes*, pour $m = 0 < p$ on a des *contraintes-égalités* et pour $p = 0 < m$ on a des *contraintes-inégalités*.

La *valeur optimale* (ou *minimale*) du problème (3.2) est définie par $\overline{f} = \inf_{x \in \mathfrak{C}} f(x)$.

On a $\overline{f} \in \mathbb{R} \cup \{\pm\infty\}$.

On dira que $\overline{x}$ est un *point optimal* (ou *minimal*) si $\overline{x}$ est admissible et $f(\overline{x}) = \overline{f}$.
L'ensemble des points optimaux est $\{x \mid x \in \mathfrak{C}\,, f(x) = \overline{f}\}$.

Un point $\overline{x} \in \mathfrak{C}$ est un *minimum local* (ou *relatif*) (resp. *minimum local strict*)
s'il existe un voisinage V de $\overline{x}$ tel que $f(\overline{x}) \leq f(x)$ pour tout $x \in \mathfrak{C} \cap V$
(resp. $f(\overline{x}) < f(x)$ pour tout $x \in \mathfrak{C} \cap V$, $x \neq \overline{x}$).

Pour $\varepsilon > 0$ donné, une solution à ε-près, ou ε-sous-optimale, de (3.2) est un élément $\overline{x}_\varepsilon \in \mathfrak{C}$ tel que $f(\overline{x}_\varepsilon) \leq \overline{f} + \varepsilon$.
La suite $(x_k) \in \mathfrak{C}$ est une *suite minimisante* si $\lim_{k \to +\infty} f(x_k) = \overline{f}$.

On se restreint ici aux problèmes où les paramètres x appartiennent à un espace vectoriel de dimension finie et sont continues, *i.e.* $x \in \mathbb{R}^n$. On ne considère pas les problèmes d'*optimisation discrète* ou *combinatoire* où $x \in \mathbb{Z}^n$, ni les problèmes en dimension infinie où x appartient à un espace fonctionel.

Suivant la forme des fonctions f, g et h on obtient une classification des problèmes d'optimisation :

- *Programmation linéaire.* Dans ce cas la fonction coût est linéaire $f(x) = c.x$, avec $c \in \mathbb{R}^n$ donnée et g est affine. On a $g_i(x) = a_i.x - b_i$ pour $1 \leq i \leq m$ et $h = 0$.
  L'ensemble des contraintes est donné par
  $\mathfrak{C} = \{x \in \mathbb{R}^n \mid a_i.x \leq b_i, 1 \leq i \leq m\}$, c'est un polyèdre convexe fermé.

- *Optimisation quadratique.* La fonction-objectif est quadratique $f(x) = \frac{1}{2}\, x^t Q x + q.x$ avec Q symétrique définie positive et les contraintes sont encore affines
  $\mathfrak{C} = \{x \in \mathbb{R}^n \mid a_i.x \leq b_i, 1 \leq i \leq m\}$.

- *Optimisation convexe.* La fonction-objectif f et les contraintes-inégalités g_i sont convexes, les h_i sont affines, de la forme $h_i(x) = a_i.x - b_i$ pour $1 \leq i \leq p$. Dans ce cas l'ensemble contrainte $\mathfrak{C}$ est un convexe.

- *Optimisation différentiable* ou *lisse.* Les fonctions f, g_i $(1 \leq i \leq m)$ et h_j $(1 \leq j \leq p)$ sont différentiables (p.ex. de classe $\mathcal{C}^1$, $\mathcal{C}^2$).

- *Optimisation non-différentiable.* C'est le cas si certaines fonctions parmi f, g_i et h_j ne sont pas régulières.

Les hypothèses sur f, g et h vont déterminer de façon essentielle l'existence et l'unicité d'une solution de (3.2). Ce n'est qu'en ayant des conditions nécessaires et suffisantes d'existence de points optimaux que l'on peut se poser la question d'appliquer un algorithme numérique pour déterminer ces solutions.

3.1.2 Exemples

Voir cours et TD TP...

3.1.3 Ensembles et fonctions convexes

En optimisation on obtient des résultats forts d'existence et unicité de minima si on utilise la convexité, d'où l'importance des définitions et résultats de cette section.

DÉFINITION 3.1.1
Un ensemble $C \subset \mathbb{R}^n$ est un ensemble convexe si pour tout $(x, y) \in C^2$ le segment $[x, y] = \{t\, x + (1 - t)\, y\,;\, 0 \leq t \leq 1\}$ est inclus dans C.

Exemples :

1. L'ensemble vide, l'espace entier $\mathbb{R}^n$ et tout singleton $\{x\}$, $x \in \mathbb{R}^n$, sont des ensembles convexes.
2. Les sous-espaces vectoriels et affines de $\mathbb{R}^n$ sont des ensembles convexes : l'*hyperplan affine* $\{x \mid a.x = b\}$, $a \in \mathbb{R}^n$, $b \in \mathbb{R}$ est convexe et partage $\mathbb{R}^n$ en deux *demi-espaces* convexes $\{x \mid a.x < b\}$ et $\{x \mid a.x > b\}$ (fermés pour l'inégalité large).
3. Les boules $B(x, r)_{\|.\|} \subset \mathbb{R}^n$ sont des ensembles convexes.
4. Le *simplexe unité* $S = \{x \in \mathbb{R}^n \;\Big|\; \sum_{i=1}^{n} x_i \leq 1\,,\; x_i \geq 0 \text{ pour } i = 1, \ldots, n\}$ est convexe.

5. Un ensemble C est un *cône* si pour tout $x \in C$ on : $\{t\,x \,|\, t \geq 0\} \subset C$. Un ensemble C est un cône convexe si pour tout $(x,y) \in C^2$, $(s,t) \in (\mathbb{R}_+)^2 \quad s\,x + t\,y \in C$.
L'*hyperoctant positif* $(\mathbb{R}_+)^n$ est un exemple simple de cône convexe.
L'ensemble des matrices carrées symétriques semidéfinies positives est aussi un cône convexe.

Opérations qui conservent la convexité des ensembles.

PROPOSITION 3.1.1
(a) Soit $\{C_i\}_{i\in I}$ une famille quelconque d'ensembles convexes,
alors $C = \bigcap_{i\in I} C_i$ est un ensemble convexe.
(b) Pour $i = 1, ..., k$, soient $C_i \subset \mathbb{R}^{n_i}$ des ensembles convexes, alors $C = C_1 \times \cdots \times C_k$ est un ensemble convexe de $\mathbb{R}^{n_1+\cdots+n_k}$.
(c) Soit $C \subset \mathbb{R}^n$ un ensemble convexe et $f : \mathbb{R}^n \to \mathbb{R}^m$ une application affine, alors $f(C) = \{f(x) \,|\, x \in C\}$ est convexe.
Inversement, si $g : \mathbb{R}^m \to \mathbb{R}^n$ est une application affine, alors $g^{-1}(C) = \{x \,|\, g(x) \in C\}$ est convexe.

DÉFINITION 3.1.2
Soit $C \subset \mathbb{R}^n$ un ensemble convexe non vide et $f : C \to \mathbb{R}$.
– *On dit que f est une <u>fonction convexe</u> si pour tout $(x,y) \in C^2$ et pour tout $t \in]0,1[$*

$$f(t\,x + (1-t)\,y) \leq t\,f(x) + (1-t)\,f(y)\ .$$

f est <u>concave</u> si $-f$ est convexe.

– *La fonction f est <u>strictement convexe</u> si pour tout $(x,y) \in C^2$, $x \neq y$, et pour tout $t \in]0,1[$*

$$f(t\,x + (1-t)\,y) < t\,f(x) + (1-t)\,f(y)\ .$$

– *On dit que f est <u>fortement convexe</u> ou <u>uniformément convexe</u> de module $\nu > 0$ si*

$$f(t\,x + (1-t)\,y) \leq t\,f(x) + (1-t)\,f(y) - \frac{\nu}{2}\,t(1-t)\,\|x-y\|_2^2$$

pour tout $(x,y) \in C^2$ et pour tout $t \in]0,1[$.

– *Une fonction $f : \mathbf{dom}\, f \to \mathbb{R}$ est <u>quasi-convexe</u> si tout ensemble de niveau inférieur $L_f(\alpha) = \{x \in \mathbf{dom}\, f \mid f(x) \leq \alpha\}$ est un ensemble convexe.*

PROPOSITION 3.1.2
(a) Une fonction f est convexe si et seulement si sa restriction à tout segment inclus dans $\mathbf{dom}\, f$ est une fonction convexe.
(b) Une fonction f est convexe si et seulement si son épigraphe
$\mathbf{epi} f = \{(x,\alpha) \in \mathbb{R}^n \times \mathbb{R} \,|\, f(x) \leq \alpha\}$ est un ensemble convexe.
(c) Une fonction f est fortement convexe de module $\nu > 0$ si et seulement si la fonction $x \longmapsto f(x) - \frac{\nu}{2}\,\|x\|_2^2$ est convexe.

(d) Une fonction f est quasi-convexe si et seulement si pour tout $(x, y) \in \mathbf{dom}\, f$ et pour tout $t \in]0, 1[$

$$f(t\,x + (1-t)\,y) \leq \max(f(x), f(y))\,.$$

On dit que f est strictement quasi-convexe si on a inégalité stricte ci-dessus.

Note : Rappelons qu'une fonction convexe f est localement Lipschitz.

La proposition suivante présente des caractérisations de convexité pour des fonctions régulières.

PROPOSITION 3.1.3
Soit C un ouvert convexe non vide de $\mathbb{R}^n$.
Si f est de classe $\mathcal{C}^1$ sur C, alors
(a) f est convexe sur C si et seulement si

$$f(y) \geq f(x) + \nabla f(x).(y-x) \quad \textit{pour tout } (x, y) \in C^2\,.$$

Si f est de classe $\mathcal{C}^2$ sur C, alors
(b) f est est convexe sur C si et seulement si $\nabla^2 f(x)$ est semidéfinie positive pour tout $x \in C$;
(c) si $\nabla^2 f(x)$ est définie positive sur C, f est strictement convexe sur C ;
(d) f est fortement convexe, de module $\nu > 0$, si et seulement si

$$\textit{pour tout } x \in C \textit{ et pour tout } w \in \mathbb{R}^n \;:\quad w^t \nabla^2 f(x) w \geq \nu\, \|w\|_2^2\,.$$

i.e. *ν minore les valeurs propres de $\nabla^2 f(x)$ pour tout $x \in C$.*

Grâce à ces caractérisations on obtient les exemples suivants.

Exemples :

1. La *fonction indicatrice* I_C d'un ensemble convexe C est convexe :
$$I_C(x) = \begin{cases} 0 & x \in C \\ +\infty & x \notin C\,. \end{cases}$$
2. Les fonctions $x \mapsto e^{ax}$, $a \in \mathbb{R}$ et $x \mapsto |x|^p$, $p \geq 1$ sont convexes sur $\mathbb{R}$.
3. Sur $\mathbb{R}_+^*$ la fonction $x \mapsto x^a$ est convexe pour $a \geq 1$ ou $a \leq 0$ et concave si $0 \leq a \leq 1$. La fonction $x \mapsto \ln x$ est concave sur $\mathbb{R}_+^*$.
4. Sur $\mathbb{R}^n$ toute norme, $x \mapsto \|x\|$, et $x \mapsto \max\{x_1, \ldots, x_n\}$ sont des fonctions convexes.
5. La moyenne géométrique $f(x) = \left(\prod_{i=1}^n x_i\right)^{1/n}$ est concave sur $\mathbf{dom}\, f = (\mathbb{R}_+^*)^n$.
6. La fonction $f \;:\; x \mapsto 1 - e^{-\|x\|_2^2}$ est une fonction strictement quasi-convexe sur $\mathbb{R}^n$ qui n'est pas convexe.

Des opérations qui conservent la convexité sont données par la

PROPOSITION 3.1.4
(a) Soient $f_1, \ldots, f_m$ des fonctions convexes, $\alpha_1, \ldots, \alpha_m$ des réels positifs, alors la fonction $f = \sum_{i=1}^m \alpha_i f_i$ est convexe sur $\mathbf{dom}\, f = \bigcap_{i=1}^m \mathbf{dom}\, f_i \neq \emptyset$.

(b) Soit $\{f_y\}_{y\in I}$ une famille quelconque de fonctions convexes, alors la fonction $f = \sup_{y\in I} f_y$ est convexe sur $\mathbf{dom}\, f = \{x \mid \sup_{y\in I} f_y(x) < +\infty\}$.

Exemple : Soit $f : \mathbb{R}^n \to \mathbb{R}$, la *fonction conjuguée* de f est définie par

$$f^*(x) = \sup_{y\in \mathbf{dom}\, f} (x.y - f(y)) ,$$

où $\mathbf{dom}\, f^*$ est l'ensemble des x pour lesquels $x.y - f(y)$ est borné sur $\mathbf{dom}\, f$.
Comme $x \mapsto x.y - f(y)$ est une fonction convexe (affine) en x, la fonctionf^* est convexe.
- Soit $f(y) = e^y$, alors la fonction $y \mapsto xy - e^y$ n'est pas bornée pour $x < 0$; pour $x > 0$ on a un maximum en $y = \ln x$, donc $f^*(x) = x \ln x - x$; pour $x = 0$ on trouve $f^*(0) = 0$. Finalement $\mathbf{dom}\, f^* = \mathbb{R}_+$ et $f^*(x) = x \ln x - x$.
- Soit $f(y) = y \ln y$ définie sur $\mathbb{R}_+$, on montre que $\mathbf{dom}\, f^* = \mathbb{R}$ et $f^*(x) = e^{x-1}$.
- Si $f = I_S$ la fonction indicatrice d'un ensemble $S \subset \mathbb{R}^n$, alors $f^*(x) = I_S^*(x) = \sup_{y\in S}(x.y)$.

Les propriétés suivantes sont souvent utilisés pour obtenir des résultats d'existence de minima

Définition 3.1.3
- *On dit que la fonction $f : \mathbb{R}^n \to \mathbb{R}$ est <u>coercive</u> si $\lim_{\|x\|\to+\infty} f(x) = +\infty$.*

- *Soit C un ouvert convexe de $\mathbb{R}^n$, on dit que la fonction $F : C \to \mathbb{R}$ est <u>monotone</u>, respectivement <u>fortement monotone</u> de module $\nu > 0$, si pour tout $(x, y) \in C^2$ $(F(x) - F(y)).(x - y) \geq 0$, respectivement si pour tout $(x, y) \in C^2$ $(F(x) - F(y)).(x - y) \geq \nu\|x - y\|_2^2$.*

- *Soit C un ouvert convexe de $\mathbb{R}^n$, la fonction $f : C \to \mathbb{R}$ est <u>elliptique si elle est de classe $\mathcal{C}^1$ sur C et si ∇f est fortement monotone de module $\nu > 0$ sur C</u> .*

Proposition 3.1.5
Soit f une fonction de classe $\mathcal{C}^1$ sur l'ouvert convexe C, alors f est elliptique de module $\nu > 0$ si et seulement si f est fortement convexe de module ν .
Dans ce cas f est strictement convexe et coercive, en particulier les ensembles $L_f(\alpha) = \{x \in C \mid f(x) \leq \alpha\}$ sont bornés.

3.1.4 Caractérisation de points optimaux

Dans cette section on va présenter quelques résultats qui permettent, grâce aux hypothèses sur f (*e.g.* régularité, convexité, coercivité, ...) d'assurer l'existence et/ou l'unicité d'un minimum (local, global, strict, ...) du problème de minimisation $\inf_{\mathcal{D}} f$.

Théorème 3.1.1
Soit U une partie non vide fermée de $\mathbb{R}^n$ et $f : \mathbb{R}^n \to \mathbb{R}$ continue. Si U est non borné on suppose que f est coercive.
Alors il existe au moins un $\overline{x} \in U$ avec $f(\overline{x}) = \inf_{x\in U} f(x)$.

THÉORÈME 3.1.2
Soit $\mathcal{D}$ un ouvert de $\mathbb{R}^n$ et $f : \mathcal{D} \to \mathbb{R}$ et $\overline{x} \in \mathcal{D}$.

1. *Si $f \in \mathcal{C}^1(\mathcal{D})$ et si $\overline{x}$ est un minimum local de f, alors nécessairement $\nabla f(\overline{x}) = 0$.*
2. *Si $f \in \mathcal{C}^2(\mathcal{D})$ et si $\overline{x}$ est un minimum local de f, alors nécessairement $\nabla^2 f(\overline{x})$ est semidéfinie positive.*
3. *Soit $f \in \mathcal{C}^2(\mathcal{D})$, si $\nabla f(\overline{x}) = 0$ et $\nabla^2 f(\overline{x})$ est définie positive, alors $\overline{x}$ est un minimum local strict de f.*

Le résultat suivant montre que la convexité permet de passer de propriétés locales à des propriétés globales.

THÉORÈME 3.1.3
Soit C un convexe non vide de $\mathbb{R}^n$ et f convexe sur C, alors
– *chaque minimum local de f est un minimum global sur C ;*
– *si f est différentiable, alors $\overline{x} \in C$ est un minimum global si et seulement si pour tout $y \in C$: $\nabla f(\overline{x}).(y - \overline{x}) \geq 0$;*
si de plus C est un ouvert on a : $\nabla f(\overline{x}) = 0 \Leftrightarrow \overline{x}$ est un minimum global.

THÉORÈME 3.1.4
Soit $\mathcal{D}$ un ouvert convexe non vide de $\mathbb{R}^n$ et $f \in \mathcal{C}^2(\mathcal{D})$, si :
(i) soit il existe $\alpha \in \mathbb{R}$ tel que l'ensemble de niveau inférieur $L_f(\alpha) = \{x \in \mathcal{D} \mid f(x) \leq \alpha\}$ est fermé, soit $\mathcal{D} = \mathbb{R}^n$;
(ii) il existe $\nu > 0$ tel que $w^t \nabla^2 f(x) w \geq \nu \|w\|_2^2$ pour tout $x \in \mathcal{D}$ et tout $w \in \mathbb{R}^n$,
alors f admet un unique minimum global (strict) sur $\mathcal{D}$.

Exemple : On veut minimiser sur $\mathbb{R}^n$ la fonction quadratique $f(x) = x^t P x + qx + r$, où P est symétrique semidéfinie positive, c'est un problème d'optimisation quadratique sans contrainte. L'hypothèse sur P entraine que f est convexe et la CNS d'optimalité pour $\overline{x} \in \mathbb{R}^n$ s'écrit

$$\nabla f(\overline{x}) = 2P\overline{x} + q = 0\,.$$

Si P est inversible, *i.e.* défini positive, f est strictement convexe et on a une solution unique $\overline{x} = -P^{-1}q$. Si $q \notin \mathbf{Im}(P)$, il n'y a pas de solution, f n'admet pas de borne inférieure ; si $q \in \mathbf{Im}(P)$, mais P non inversible, l'ensemble des points optimaux est un sous-espace affine de $\mathbb{R}^n$: $\overline{x} \in \mathbf{ker}(P) - P^+ q$, où P^+ est la matrice pseudo-inverse de MOORE-PENROSE.

THÉORÈME 3.1.5
Soit $\mathcal{D}$ un ouvert convexe non vide de $\mathbb{R}^n$ et $f : \mathcal{D} \to \mathbb{R}$ continue et quasi-convexe, alors tout minimum local strict est un minimum global.
S'il existe $\alpha \in \mathbb{R}$ tel que l'ensemble de niveau inférieur $L_f(\alpha) = \{x \in \mathcal{D} \mid f(x) \leq \alpha\}$ est compact, alors il existe $\overline{x} \in L_f(\alpha)$ minimum local de f.
Si f est de plus strictement quasi-convexe, alors il existe un unique minimum $\overline{x}$.

3.2 Minimisation sans contrainte

3.2.1 Méthodes de descente. Vitesse de convergence

Dans ce qui suit on suppose en général que f est au moins de classe $\mathcal{C}^1$ sur l'ouvert $\mathcal{D}$ de $\mathbb{R}^n$. Pour un point $x^{(0)} \in \mathcal{D}$ donné on veut construire une suite $(x^{(\mathrm{k})})_k$ vérifiant

$$\begin{aligned} &(i) \quad x^{(\mathrm{k})} \in L_f(f(x^{(0)})) \ \text{ pour tout } k \geq 1 \\ &(ii) \quad \nabla f(x^{(\mathrm{k})}) \to 0 \ \text{ pour } k \to +\infty \\ &(iii) \quad \left\| x^{(\mathrm{k}+1)} - x^{(\mathrm{k})} \right\| \to 0 \ \text{ pour } k \to +\infty \end{aligned}$$

Si l'ensemble $L_f(f(x^{(0)}))$ est compact on a existence d'une sous-suite qui converge vers un *point stationnaire* $\overline{x}$, *i.e.* $\nabla f(\overline{x}) = 0$.

ALGORITHME 3.1 (ALGORITHME DE DESCENTE)
Un algorithme de descente général se présente sous la forme suivante :

choisir $x^{(0)}$, *poser* $k = 0$
tant que $\left\|\nabla f(x^{(k)})\right\| >$ *seuil de tolérance*
déterminer une direction $d^{(k)}$ *telle que* $\quad \exists \sigma > 0 \ : \ f(x^{(k)} + \sigma\, d^{(k)}) < f(x^{(k)})$
déterminer un pas convenable $\sigma_k > 0$
poser $x^{(k+1)} = x^{(k)} + \sigma\, d^{(k)}$ *et faire* $k = k + 1$

Remarques :

1) On dit que $d \in \mathbb{R}^n$ est une <u>*direction de descente*</u> de f en x si $\nabla f(x).d < 0$, c.-à-d. si la dérivée directionnelle de f dans la direction d est négative en x.
Dans ce cas il existe $\tilde{\sigma} > 0$ tel que pour tout $\sigma \in]0, \tilde{\sigma}[: \ f(x + \sigma d) < f(x)$.
On définit la *direction normalisée de plus grande descente* de f en x comme étant la solution de

$$\min\{\nabla f(x).d \mid \|d\| = 1\} .$$

Ce minimum existe, n'est pas nécessairement unique et dépend du choix de $\|.\|$.
Pour les algorithmes de descente il suffit en général d'avoir une *direction de plus grande descente, i.e.* $d \in \mathbb{R}^n \setminus \{0\}$ telle que $d \,/\, \|d\|$ soit une direction normalisée de plus grande descente.

2) Après avoir fixé d il faut déterminer un pas σ «convenable» : on souhaite avoir une descente significative de $f(x^{(\mathrm{k})})$ vers $f(x^{(\mathrm{k}+1)})$, ne pas rester trop près de $x^{(\mathrm{k})}$, ne pas mettre trop de temps à trouver σ, etc.
Si la fonction coût le permet, ou pour démontrer des résultats de convergence et vitesse de convergence, on peut supposer trouver un minimum exact de f sur $\{x + \sigma d \,|\, \sigma > 0\}$.

Vitesse de convergence :

On considère une méthode itérative convergente $x^{(k)} \to \overline{x}$.
On dit que la méthode est *d'ordre* $r \geq 1$, s'il existe une constante $C \in \mathbb{R}_+^*$ telle que, pour k suffisamment grand

$$\frac{\left\|x^{(k+1)} - \overline{x}\right\|}{\left\|x^{(k)} - \overline{x}\right\|} \leq C \ .$$

Si $r = 1$ il faut $C \in]0,1[$ pour avoir convergence et on a alors une *convergence linéaire.*

Si $r = 2$ on a une *convergence quadratique.*

La quantité $-\log_{10}\left\|x^{(k)} - \overline{x}\right\|$ mesure le nombre de décimales exactes dans l'approximation $x^{(k)}$. Si pour une méthode d'ordre un on a asymptotiquement

$$-\log_{10}\left\|x^{(k+1)} - \overline{x}\right\| \sim -\log_{10}\left\|x^{(k)} - \overline{x}\right\| - \log_{10}(C)$$

alors on gagne à chaque itération $\log_{10}(1/C)$ décimales.
Si pour une méthode d'ordre $r > 1$ on a asymptotiquement

$$-\log_{10}\left\|x^{(k+1)} - \overline{x}\right\| \sim -r\ \log_{10}\left\|x^{(k)} - \overline{x}\right\| - \log_{10}(C)$$

alors $x^{(k+1)}$ a r fois plus de décimales exactes que $x^{(k)}$.

Si $\lim\limits_{k\to+\infty} \dfrac{\left\|x^{(k+1)} - \overline{x}\right\|}{\left\|x^{(k)} - \overline{x}\right\|} = 0$ on dit que l'on a *convergence superlinéaire.*

3.2.2 Minimisation en une dimension

Dans cette section on suppose que l'on sait trouver à chaque itération une direction de descente $d^{(k)}$. On va s'intéresser au problème de la détermination d'un pas σ_k convenable. Parmi les nombreuses possibilités proposées dans la littérature nous allons présenter les *conditions de* Wolfe (faibles) sur $\sigma > 0$:

$$f(x^{(k)} + \sigma d^{(k)}) \leq f(x^{(k)}) + c_1\, \sigma\, \nabla f(x^{(k)}).d^{(k)} \tag{W1}$$

$$\nabla f(x^{(k)} + \sigma d^{(k)}).d^{(k)} \geq c_2\, \nabla f(x^{(k)}).d^{(k)} \tag{W2}$$

avec $0 < c_1 < c_2 < 1$.

Pour expliquer ces conditions posons $\phi(\sigma) = f(x^{(k)} + \sigma d^{(k)})$ et $l(\sigma) = \phi(0) + (c_2\phi'(0))\sigma$. On remarque que $\phi'(0) < 0$ car $d^{(k)}$ est une direction de descente.

La condition (W1) impose que la réduction de f est proportionnelle à σ et $\phi'(0)$. On ne retiendra que les valeurs de σ pour lesquelles le graphe de ϕ est en dessous de la droite l, comme $0 < c_1 < 1$ ceci est possible au moins pour σ petit.

La condition (W2) implique que $\phi'(\sigma) \geq c_2\phi'(0) \geq \phi'(0)$: si $\phi'(\sigma)$ est «très» négatif (*i.e.* $< c_2\phi'(0)$), on va chercher plus loin, sinon on peut s'arrêter. De de cette façon le pas σ ne sera pas trop petit.

On obtient des contraintes plus fortes si l'on remplace (W2) par

$$\left|\nabla f(x^{(k)} + \sigma d^{(k)}).d^{(k)}\right| \leq c_2\ \left|\nabla f(x^{(k)}).d^{(k)}\right| \tag{W3}$$

Les (W1) et (W3) sont les *conditions de* Wolfe *fortes.* La contrainte (W3) entraîne que $c_2\phi'(0) \leq \phi'(\sigma) \leq -c_2\phi'(0)$ c.-à-d. $\phi'(\sigma)$ n'est pas «trop» positif.

L'existence de valeurs de σ vérifiant (W1-2) et (W1-3) est donnée par la

PROPOSITION 3.2.1
Soit $f \in \mathcal{C}^1(\mathbb{R}^n)$ et d une direction de descente de f en x, on suppose que f est minorée sur $\{x + \sigma d \,|\, \sigma \geq 0\}$. Alors, si $0 < c_1 < c_2 < 1$, il existe des intervalles dans $\mathbb{R}_+$ qui vérifient les conditions de WOLFE *faibles et fortes.*

THÉORÈME 3.2.1 (ZOUTENDIJK)
Soit f de classe $\mathcal{C}^1$ sur l'ouvert $\mathcal{D} \subset \mathbb{R}^n$ et $x^{(0)} \in \mathcal{D}$ tel que l'ensemble de niveau inférieur $L_f(f(x^{(0)})) = \{x \in \mathcal{D} \mid f(x) \leq f x^{(0)})\}$ est fermé, de plus on suppose que f est minoré sur $L_f(f(x^{(0)}))$ et que ∇f est Lipschitz sur $\mathcal{D}$.
Considérons la suite $(x^{(k)})_k$, définie pour tout $k \geq 0$ par $x^{(k+1)} = x^{(k)} + \sigma_k d^{(k)}$,
où $d^{(k)}$ est une direction de descente et σ_k vérifie les conditions de WOLFE*, alors*

$$\sum_{k=0}^{+\infty} \cos^2 \theta_k \, \left\| \nabla f(x^{(k)}) \right\|_2^2 < +\infty \, ,$$

où $\cos\theta_k = -\dfrac{\nabla f(x^{(k)}).d^{(k)}}{\|\nabla f(x^{(k)})\|_2 \, \|d^{(k)}\|_2}$.

COROLLAIRE 3.2.1
Si f et la méthode de descente $(x^{(k)})_k$ vérifient les hypothèses du théorème 3.2.1 et si de plus il existe $\delta > 0$ tel que pour k suffisamment grand on a $\cos\theta_k \geq \delta$, alors $\lim_k \left\| \nabla f(x^{(k)}) \right\|_2 = 0$.

Ce corollaire est un résultat de *convergence globale* : pour des fonctions coût vérifiant les hypothèses du théorème 3.2.1 et si la direction de descente ne devient pas perpendiculaire au gradient, alors la méthode itérative converge vers un point stationnaire $\overline{x}$ de f, ceci à partir d'un point initial $x^{(0)}$ qui n'est pas nécessairement proche de $\overline{x}$.

Il existe des algorithmes qui déterminent un σ_k vérifiant les conditions de WOLFE, nous allons proposer un algorithme simplifié, le *backtracking* ou recherche rétrograde.

ALGORITHME 3.2 (ALGORITHME DE BACKTRACKING)
Choisir σ_{init}, ρ, $c \in]0,1[$
$\sigma = \sigma_{init}$
tant que $f(x^{(k)} + \sigma\, d^{(k)}) > f(x^{(k)}) + c\,\sigma\, \nabla f(x^{(k)}).d^{(k)}$
  $\sigma = \rho\sigma$
$\sigma_k = \sigma$

À partir d'une «grande» valeur σ_{init}, on détermine σ_k en diminuant à chaque itération la valeur de σ jusqu'à ce que σ vérifie (W1) et comme l'on part loin de 0 $x^{(k+1)}$ ne sera pas trop proche de $x^{(k)}$.

3.2.3 Méthode de descente du gradient

Pour cet algorithme on choisit $d^{(\mathrm{k})} = -\nabla f(x^{(\mathrm{k})})$ comme direction de descente, l'algorithme complet s'écrit :

ALGORITHME 3.3 (DESCENTE DE GRADIENT)

choisir $x^{(0)} \in \mathbf{dom}\, f$ *et poser* $k = 0$

tant que $\left\|\nabla f(x^{(k)})\right\| > \varepsilon$

$d^{(k)} = -\nabla f(x^{(k)})$

déterminer $\sigma_k > 0$

$x^{(k+1)} = x^{(k)} + \sigma\, d^{(k)}$

$k = k + 1$

Grâce au corollaire 3.2.1 la méthode de descente du gradient est globalement convergente. Pour étudier la vitesse de convergence on va étudier un cas particulier.

Application : On considère le problème d'optimisation quadratique $\min_{x\in\mathbb{R}^n} f(x)$,

où f est une fonction fortement convexe quadratique $f(x) = \frac{1}{2}\, x^t Q x - b.x$, avec $b \in \mathbb{R}^n$ et Q une matrice symétrique définie positive de valeurs propres $0 < \lambda_1 \leq \cdots \leq \lambda_n$.

On a $\nabla f(x) = Qx - b$ et $\nabla^2 f(x) = Q$. La fonction f admet un minimum global unique $\overline{x}$ qui est solution de $Qx = b$.

Dans ce cas on peut déterminer le pas de descente optimal σ_k et l'itération s'écrit :

$$x^{(\mathrm{k}+1)} = x^{(\mathrm{k})} - \left[\frac{\left\|\nabla f(x^{(\mathrm{k})})\right\|_2^2}{\nabla f(x^{(\mathrm{k})})^t\, Q\, \nabla f(x^{(\mathrm{k})})}\right] \nabla f(x^{(\mathrm{k})}) \ .$$

Posons $\|x\|_Q^2 = x^t Q x$, on montre alors que

$$\left\|x^{(\mathrm{k}+1)} - \overline{x}\right\|_Q^2 = \left(1 - \frac{\left\|\nabla f(x^{(\mathrm{k})})\right\|_2^4}{(\nabla f(x^{(\mathrm{k})})^t\, Q\, \nabla f(x^{(\mathrm{k})}))(\nabla f(x^{(\mathrm{k})})^t\, Q^{-1}\, \nabla f(x^{(\mathrm{k})}))}\right) \left\|x^{(\mathrm{k})} - \overline{x}\right\|_Q^2 \ ,$$

et, en utilisant l'inégalité de KANTOROVICH :

$$\left\|x^{(\mathrm{k}+1)} - \overline{x}\right\|_Q^2 \leq \left(\frac{c_2(Q) - 1}{c_2(Q) + 1}\right)^2 \left\|x^{(\mathrm{k})} - \overline{x}\right\|_Q^2 \ ,$$

où $c_2(Q) = \lambda_n/\lambda_1$.

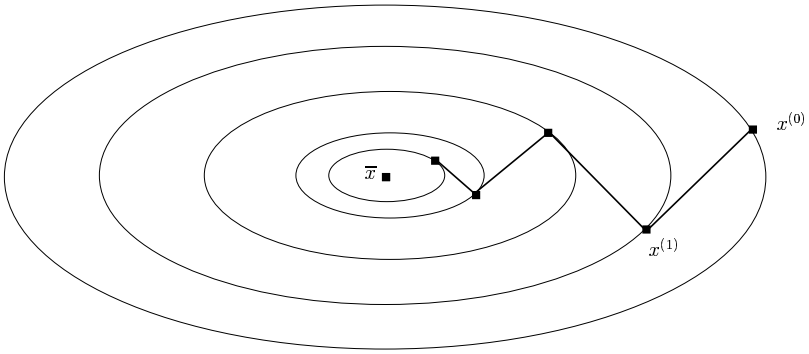
Ainsi, dans le cas d'une fonction coût quadratique, la méthode du gradient est d'ordre un et la vitesse de convergence dépend de $c_2(Q)$: si $c_2(Q) = 1$, c.-à-d. $Q = \lambda I_n$ on a convergence en une itération ; si Q est mal conditionnée la convergence devient très lente.

Le conditionnement de Q indique la déformation de l'ellipsoïde $x^t Q x = \alpha > 0$:
pour $c_2(Q) = 1$ on trouve des sphères, sinon les lignes de niveau sont des ellipsoïdes

allongés le plus dans la direction associé à λ_n.
Comme σ_k est le minimum exact de $\phi(\sigma) = f(x^{(\mathrm{k})} - \sigma \nabla f(x^{(\mathrm{k})}))$ on a :

$$d^{(\mathrm{k}+1)}.d^{(\mathrm{k})} = \nabla f(x^{(\mathrm{k}+1)}).\nabla f(x^{(\mathrm{k})}) = \phi'(\sigma_k) = 0\,,$$

le chemin de $x^{(0)}$ vers $\overline{x}$ est donc en zigzag, ceci est illustré en dimension deux dans la figure suivante.



3.2.4 Méthode de la plus forte descente

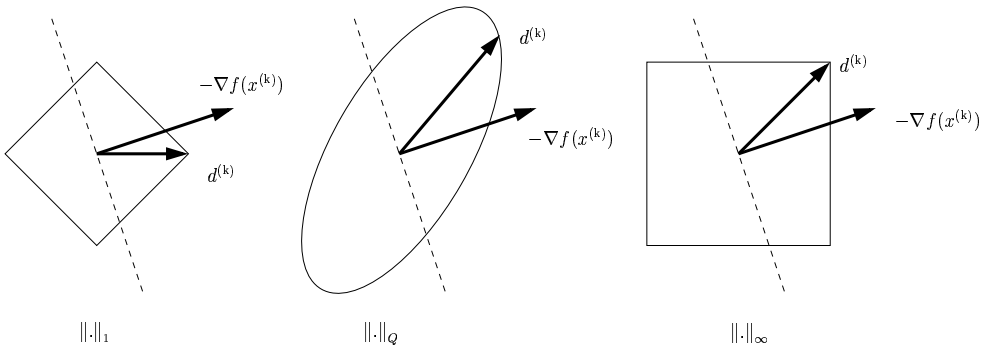
Pour une norme $\|.\|$ donnée on choisit

$$d^{(\mathrm{k})} = \arg\min\{\nabla f(x^{(\mathrm{k})}).d \mid \|d\| = 1\}\,,$$

c.-à-d. que l'on cherche une direction dans la sphère unité de $\|.\|$ qui s'étend le plus dans le demi-plan de $-\nabla f(x^{(\mathrm{k})})$:

- si $\|.\| = \|.\|_2$, on obtient la méthode de descente du gradient ;
- si $\|.\| = \|.\|_1$, on a $d^{(\mathrm{k})} = signe\left(\left(\nabla f(x^{(\mathrm{k})})\right)_i\right) e_i$,
  où $i \in \{1, \ldots, n\}$ est tel que $\left\|\nabla f(x^{(\mathrm{k})})\right\|_\infty = \left|\left(\nabla f(x^{(\mathrm{k})})\right)_i\right|$.

Sur la figure suivante on a représenté en dimension deux la sphère unité pour les normes $\|.\|_1$, $\|.\|_Q$ (Q symétrique définie positive) et $\|.\|_\infty$.



3.2.5 Méthode de relaxation ou des directions alternées

Dans cette méthode on prend successivement les vecteurs de la base de $\mathbb{R}^n$ comme direction de recherche : à chaque itération on minimise par rapport à une seule variable, *i.e.* à l'étape k, on minimise la fonction $\sigma \longmapsto f(x^{(k)} + \sigma\, e_k)$, pour $\sigma \in \mathbb{R}$.

Cette méthode est voisine de celle de la plus forte descente en $\|.\|_1$, mais $\pm e_k$ n'est pas nécessairement une direction de descente en $x^{(k)}$. Cet algorithme est très simple mais on ne peut pas garantir la convergence vers un point stationnaire et même si on on a convergence celle-ci peut être très lente.

Note : Si $f(x) = \frac{1}{2}\, x^t A x - b.x$, avec A définie positive, alors cette méthode correspond à la méthode itérative de Gauss-Seidel pour résoudre $Ax = b$.

En effet, à l'itération $k \,(\mathrm{mod}\, n)$ on ne change que la k^{e} coordonnée de $x^{(k)}$, il faut n itérations avant d'avoir mis à jour toutes les coordonnées de $x^{(k)}$.

3.2.6 Méthode de Newton

On suppose que f est de classe $\mathcal{C}^2$ et on utilise un modèle quadratique de f :

$$f(x^{(k)} + d) \sim \tilde{f}_k(d) = f(x^{(k)}) + \nabla f(x^{(k)}).d + \frac{1}{2}\, d^t \nabla^2 f(x^{(k)})\, d\,.$$

Si $\nabla^2 f(x^{(k)})$ est définie positive, la direction $d^{(k)}$ sera le minimum de $\tilde{f}_k$:

$$d^{(k)} = [\nabla^2 f(x^{(k)})]^{-1}\, \nabla f(x^{(k)})\,.$$

Dans ce cas $\nabla f(x^{(k)}).d^{(k)} = -{d^{(k)}}^t \nabla^2 f(x^{(k)})\, d^{(k)} \leq 0$, on obtient bien une direction de descente.

Théorème 3.2.2
Soit f de classe $\mathcal{C}^2$, $\overline{x} \in \mathbb{R}^n$ tel que $\nabla f(\overline{x}) = 0$ et $\nabla^2 f(\overline{x})$ définie positive et $\nabla^2 f$ une fonction Lipschitz au voisinage de $\overline{x}$.
On considère la suite $(x^{(k)})$ définie par $x^{(0)}$ et

$$x^{(k+1)} = x^{(k)} - [\nabla^2 f(x^{(k)})]^{-1}\, \nabla f(x^{(k)})\,.$$

Alors si $x^{(0)}$ est suffisamment proche de $\overline{x}$
(i) la suite $(x^{(k)})$ converge vers $\overline{x}$;
(ii) la méthode de Newton est d'ordre 2 ;
(iii) la suite $\left(\left\|\nabla f(x^{(k)})\right\|\right)$ converge vers 0 de façon quadratique.

Remarques :

1. Si en un point stationnaire la matrice hessienne est définie positive et si $x^{(0)}$ est proche de $\overline{x}$, on a une convergence très rapide. Dans le cas contraire l'algorithme peut diverger.
2. Par contre le coût ce cette méthode est grand : à chaque itération on doit construire et garder en mémoire la matrice hessienne et résoudre $\nabla^2 f(x^{(k)})\, d = \nabla f(x^{(k)})$, en utilisant l'algorithme de Cholesky cela fait $O(n^2)$ opérations.
3. Les *methodes quasi*-Newton remplacent la matrice hessienne par une approximation B_k qui vérifie la relation de la sécante

$$B_k(x^{(k)} - x^{(k-1)}) = \nabla f(x^{(k)}) - \nabla f(x^{(k-1)})\,.$$

3.2.7 Méthode du gradient conjugué

Dans cette méthode on utilise encore un modèle quadratique de f, nous allons donc d'abord présenter l'algorithme appliqué à une fonction fortement convexe quadratique.

Cas linéaire

Soit $f(x) = \frac{1}{2}\, x^t A x - b.x$, avec A symétrique définie positive.
On sait alors qu'il existe un minimum unique sur $\mathbb{R}^n$ donné par $\overline{x} = A^{-1}b$.
Dans la suite on note $r(x) = Ax - b = \nabla f(x)$ le résidu.

DÉFINITION 3.2.1
Les vecteurs non nuls $\{d_1, \ldots, d_p\}$ sont dits conjugués par rapport à la matrice A si

$$\textit{pour tout } i \neq j \in \{1, \ldots, p\} \; : \; d_i^t A\, d_j = 0 \; .$$

LEMME 3.2.1
Un ensemble de vecteurs conjugués par rapport à A est un ensemble de vecteurs linéairement indépendants.

Posons $\phi(\sigma) = f(x + \sigma\, d)$, alors $\phi'(\sigma) = 0$ pour

$$\sigma = -\frac{r(x).d}{d^t A\, d} \; . \tag{3.3}$$

THÉORÈME 3.2.3
Soit $x^{(0)} \in \mathbb{R}^n$ et $\{d_0, \ldots, d_{n-1}\}$ un ensemble de vecteurs conjugués par rapport à A.
On considère la suite définie par

$$x^{(k+1)} = x^{(k)} + \sigma_k\, d_k \; , \qquad \text{où } \sigma_k = -\frac{r(x^{(k)}).d_k}{d_k^t A\, d_k} \; .$$

Alors $r(x^{(k)}).d_i = 0 \quad$ *pour* $i = 0, \ldots, k-1$,
$x^{(k)}$ *minimise* f *sur* $x^{(0)} + \mathrm{Vect}\{d_0, \ldots, d_{k-1}\}$
et la suite $(x^{(k)})$ *converge en au plus n itérations.*

Pour pouvoir utiliser la méthode itérative du théorème précédent il faut construire un ensemble de vecteurs conjugués par rapport à A.
Or, afin de réduire le nombre d'opérations, on se propose de déduire $d^{(\mathrm{k})}$ à partir de $d^{(\mathrm{k\text{-}1})}$ uniquement. Dans l'algorithme du gradient conjugué linéaire on prend

$$d^{(\mathrm{k})} = -\nabla f(x^{(\mathrm{k})}) + \beta_k\, d^{(\mathrm{k\text{-}1})}$$

avec β_k tel que ${d^{(\mathrm{k})}}^t A\, d^{(\mathrm{k\text{-}1})} = 0$, on en déduit

$$\beta_k = \frac{r(x^{(\mathrm{k})})^t A\, d^{(\mathrm{k\text{-}1})}}{{d^{(\mathrm{k\text{-}1})}}^t A\, d^{(\mathrm{k\text{-}1})}} \; . \tag{3.4}$$

Comme $d^{(\mathrm{k})}.\nabla f(x^{(\mathrm{k})}) = -\left\| r(x^{(\mathrm{k})}) \right\|_2^2$, on obtient bien une direction de descente.

ALGORITHME 3.4 (GRADIENT CONJUGUÉ LINÉAIRE)

choisir $x^{(0)}$ *et poser* $k = 0$

$r^{(0)} = Ax^{(0)} - b$, $d^{(0)} = -r^{(0)}$, $k = 0$

tant que $\left\| r(x^{(k)}) \right\| > \varepsilon$

$$\sigma_k = \frac{r^{(k)}.r^{(k)}}{d^{(k)^t} A\, d^{(k)}}$$

$$x^{(k+1)} = x^{(k)} + \sigma_k\, d^{(k)}$$

$$r^{(k+1)} = r^{(k)} + \sigma_k\, A\, d^{(k)}$$

$$\beta_{k+1} = \frac{r^{(k+1)}.r^{(k+1)}}{r^{(k)}.r^{(k)}}$$

$$d^{(k+1)} = -r^{(k+1)} + \beta_{k+1}\, d^{(k)}$$

$$k = k + 1$$

Noter que les formules pour σ_k et β_{k+1} ont changé, ceci est possible grâce au théorème suivant qui affirme en particulier que les $d^{(k)}$ sont conjugués.

THÉORÈME 3.2.4
Soit $x^{(k)} \neq \overline{x}$ *, obtenu après la* k^e *itération de l'algorithme du gradient conjugué linéaire avec* σ_k *, resp.* β_{k+1} *, défini par* (3.3) *, resp.* (3.4) *. Alors on a*

$$(i) \qquad r^{(k)}.r^{(i)} = 0 \qquad pour\ i = 0, \ldots, k-1 \ ;$$

$$(ii) \ \mathrm{Vect}\{r^{(0)}, \ldots, r^{(k)}\} = \mathrm{Vect}\{r^{(0)}, Ar^{(0)}, \ldots, A^k r^{(0)}\} \ ;$$

$$(iii) \ \mathrm{Vect}\{d^{(0)}, \ldots, d^{(k)}\} = \mathrm{Vect}\{r^{(0)}, Ar^{(0)}, \ldots, A^k r^{(0)}\} \ ;$$

$$(iv) \qquad d^{(k)^t} A\, d^{(i)} = 0 \qquad pour\ i = 0, \ldots, k-1 \ .$$

Le sous-espace vectoriel $\mathrm{Vect}\{r^{(0)}, Ar^{(0)}, \ldots, A^k r^{(0)}\}$ est l'*espace de* KRYLOV $K_{k+1}(A, r^{(0)})$.

Remarques :

1. Dans la démonstration du théorème il est essentiel de choisir $d^{(0)} = -\nabla f(x^{(0)})$, c'est à partir de $d^{(0)}$ que l'on construit des gradients orthogonaux et des directions conjugées.
2. L'algorithme du GC linéaire est surtout utile pour résoudre des grands systèmes creux, en effet il suffit de savoir appliquer la matrice A à un vecteur.
3. La convergence peut être assez rapide : si A admet seulement r valeurs propres distinctes la convergence a lieu en au plus r itérations.
   De façon générale la vitesse de convergence dépend de la distance entre les valeur

propres, *i.e.* le conditionnement de A. Suivant la nature du problème on applique souvent des préconditionneurs, c.-à-d. on transforme le problème grâce à un changement de variables $x_{nouv} = Cx_{anc}$ avec C telle que le conditionnement de la matrice $C^{-t}AC^{-1}$ soit beaucoup plus petit que celui de A.

Cas non linéaire

Pour pouvoir appliquer l'algorithme du GC linéaire au cas d'une fonction f quelconque il faut :

remplacer $r(x^{(k)})$ par $\nabla f(x^{(k)})$;
déterminer σ_k par une minimisation en dimension un.

L'algorithme s'écrit alors :

ALGORITHME 3.5 (GRADIENT CONJUGUÉ [FLETCHER-REEVES 1964])

choisir $x^{(0)}$

calculer $f^{(0)} = f(x^{(0)})$ *et* $\nabla f^{(0)} = \nabla f(x^{(0)})$

$d^{(0)} = -\nabla f^{(0)}$, $k = 0$

tant que $\left\|\nabla f^{(k)}\right\| > \varepsilon$

déterminer σ_k

$x^{(k+1)} = x^{(k)} + \sigma_k\, d^{(k)}$

calculer $\nabla f^{(k+1)} = \nabla f(x^{(k)})$

$$\beta_{k+1}^{FR} = \frac{\nabla f^{(k+1)}.\nabla f^{(k+1)}}{\nabla f^{(k)}.\nabla f^{(k)}}$$

$$d^{(k+1)} = -\nabla f^{(k+1)} + \beta_{k+1}^{FR}\, d^{(k)}$$

$k = k + 1$

Si f est une fonction fortement convexe quadratique et si σ_k est un minimum exact, alors cet algorithme est équivalent au GC linéaire. Dans le cas général $d^{(k)}$ peut ne pas être une direction de descente :

$$d^{(k)}.\nabla f(x^{(k)}) = -\left\|\nabla f(x^{(k)})\right\|_2^2 + \beta_{k+1}^{FR}\, d^{(k-1)}.\nabla f(x^{(k)})\ .$$

Si σ_{k-1} n'est pas un point stationnaire de $\sigma \mapsto f(x^{(k-1)} + \sigma\, d^{(k-1)})$ on peut avoir $\nabla f(x^{(k)}).d^{(k)} > 0$. On montre que si σ_{k-1} vérifie les conditions de WOLFE fortes avec $0 < c_1 < c_2 < 1/2$, alors tous les $d^{(k)}$ est une direction de descente.

Dans l'algorithme du GC on peut donner diverses expressions de β_{k+1} qui sont équivalentes dans le cas d'une fonction fortement convexe quadratique, c.-à-d. que l'on obtient chaque

fois l'algorithme du GC linéaire. L'expression suivante donne lieu à l'algorithme du GC de Polak-Ribière (1969) :

$$\beta_{k+1}^{PR} = \frac{\nabla f^{(\mathrm{k}+1)}.(\nabla f^{(\mathrm{k}+1)} - \nabla f^{(\mathrm{k})})}{\nabla f^{(\mathrm{k})}.\nabla f^{(\mathrm{k})}} \ .$$

Cet algorithme est en général plus performant que celui de Fletcher-Reeves et est utilisé le plus souvent en pratique. On peut noter qu'il existe un contre exemple qui montre que le GC de Polak-Ribière est divergent même si σ est un minimum exact.

Chapitre 4

Équations aux dérivées partielles

4.1 Introduction

Nous allons en un premier temps présenter des exemples d'applications des équations aux dérivées partiellesavant de présenter une classification qui est importante pour le choix des méthodes numériques. On ne va pas présenter des résultats d'existence, d'unicité et de stabilité d'un problème continue, on ne s'intéresse qu'aux méthodes de résolution : on suppose donc l'existence de solutions.

4.1.1 Exemples d'équations aux dérivées partielles

Physique

Sans rentrer dans les détails et en négligeant les constantes dimensionnelles, citons quelques équations aux dérivées partielles «célèbres» de la physique :

- $\Delta u = 0$ équation de Laplace
- $\Delta u - f = 0$ équation de Poisson
- $u_t - \Delta u = 0$ équation de la chaleur, diffusion homogène
- $u_{tt} - \Delta u = 0$ équation des ondes
- $u_{tt} - u_{xx} + u_t + u = 0$ équation des télégraphistes
- $u_t + cuu_x + u_{xxx} = 0$ équation de Korteweg-de Vries pour des vagues sur de l'eau peu profonde
- $i\Psi_t + \Delta\Psi = V\Psi$ équation de Schrödinger

et des systèmes d'équations aux dérivées partielles :

- $\begin{cases} E_t = \mathrm{rot}(H) \\ H_t = -\mathrm{rot}(E) \\ \mathrm{div}(E) = \mathrm{div}(H) = 0 \end{cases}$ équations de Maxwell pour le champ électrique E et le champ magnétique H

- $\begin{cases} U_t + (U.\nabla)U - \Delta U = -\nabla p \\ \mathrm{div}(U) = 0 \end{cases}$ équations de Navier-Stokes d'un fluide incompressible et visqueux

Biologie. Dynamique des populations

Dans les modèles en biologie ou en dynamique des populations on s'intéresse à la concentration d'une substance chimique ou à la densité d'une population (particules, cellules) et son évolution au cours du temps. L'inconnue u est en général fonction de $(x,t) \in \mathbb{R}^d \times \mathbb{R}_+$.

- L'équation de Fisher (1937), pour $u(x,t) : \mathbb{R} \times \mathbb{R}_+ \to \mathbb{R}$, est un modèle de dispersion en une dimension spatiale d'un gène favorable dans une population

$$u_t = k\Delta u + ru\left(1 - \frac{u}{C}\right) ,$$

  le terme de *croissance logistique* dépend de la constante de reproduction linéaire r, de la capacité de l'environnement C, et du coefficient de dispersion de la population, k .

- Les équations de FitzHugh-Nagumo

$$\begin{cases} u_t &= u_{xx} + f(u) - v \\ v_t &= \delta v_{tt} + \alpha u - \beta v \end{cases} \qquad \text{pour } \delta, \alpha, \beta \text{ dans } \mathbb{R}_+$$

  utilisées pour modéliser la transmission d'impulsion nerveuses le long d'axones ou des réactions chimiques cycliques du type Belousov-Zhabotinsky.

- Les deux équations précédentes sont des équations de type *réaction-diffusion* .

  Les équations de réaction-diffusion ont été proposées par A. Turing (1952) pour la modélisation de phénomènes de *morphogenèse* (*i.e.* développement des formes/structures). Un modèle d'interaction d'espèces ou de substances chimiques est donné par le système d'équations aux dérivées partielles

$$\frac{\partial u_i}{\partial t} = \operatorname{div}\Big(D_i \, \nabla u_i\Big) + Q_i \, , \qquad \text{pour } 1 \leq i \leq n \, ,$$

  où $u_i(x,t)$ représente la densité, resp. la concentration, de la substance i . Les matrices de diffusion D_i et les termes de réaction Q_i peuvent dépendre de (x,t) et des concentrations u_i , de façon non linéaire.
  Le terme de réaction Q_i modélise l'interaction des substances (inhibition, catalyse) et le terme $\operatorname{div}(D_i \, \nabla u)$ représente la diffusion de la substance à travers le système.
  En fonction du choix de D_i et Q_i, les concentrations u_i peuvent donner lieu à des motifs locaux : on modélise ainsi la pigmentation des coquillages, le pelage des animaux (zèbre, guépard, ...), cf. Meinhardt (page 69), et des réactions chimiques cycliques, cf. Belousov-Zhabotinsky.

  En *synthèse d'images* des chercheurs ont utilisé ce modèle pour créer des textures naturelles.

Traitement des images

Une image numérique scalaire est une matrice $I(c,l)_{0\leq c<NC,0\leq l<NL}$ où à chaque ligne l et colonne c on fait correspondre un niveau de gris $I(c,l) \in \{0,\ldots,255\}$, (c,l) est un *pixel* (angl. picture element)

Suite à des problèmes de capteurs, de transmission ou autres, l'image peut être endommagée, c'est-à-dire que les valeurs en certains pixels ont été modifiés ou détruits. Le but des techniques de *débruitage* que nous allons exposer est d'éliminer ce *bruit* tout en préservant le maximum d'information contenu dans l'image originale.

On va modéliser une image par une fonction u définie sur un rectangle ouvert de $\mathbb{R}^2$ et à valeurs dans $\mathbb{R}$, $u(x,y) \in \mathbb{R}$ si l'on se restreint à des images en niveaux de gris.

En traitement du signal on utilise les filtres linéaires passe-bas pour enlever le bruit (*i.e.* les hautes fréquences) d'un signal. Le filtre gaussien est un filtre linéaire qui est souvent utilisé car il a une bonne localisation en espace et en fréquence.

Soit u_o l'image originale bruitée et, pour $x \in \mathbb{R}^2$, posons $G_\sigma(x) = \dfrac{1}{2\pi\sigma^2} e^{-\frac{|x|^2}{2\sigma^2}}$.

On calcule le résultat du filtrage de u_o par le filtre gaussien :

$$(G_\sigma \star u_o)(x) = \int_{\mathbb{R}^2} G_\sigma(x-y) u_o(y)\, dy\,.$$

La valeur de $\sigma > 0$ indique la taille spatiale des structures éliminées par le filtrage : plus σ est grand, plus on lisse et plus on perd de détails.

Or on montre que la convolution avec la Gaussienne G_σ consiste à faire évoluer l'image originale suivant l'équation de la chaleur

$$\begin{cases} u_t &= \Delta u \\ u(x,0) &= u_o(x,y)\,. \end{cases}$$

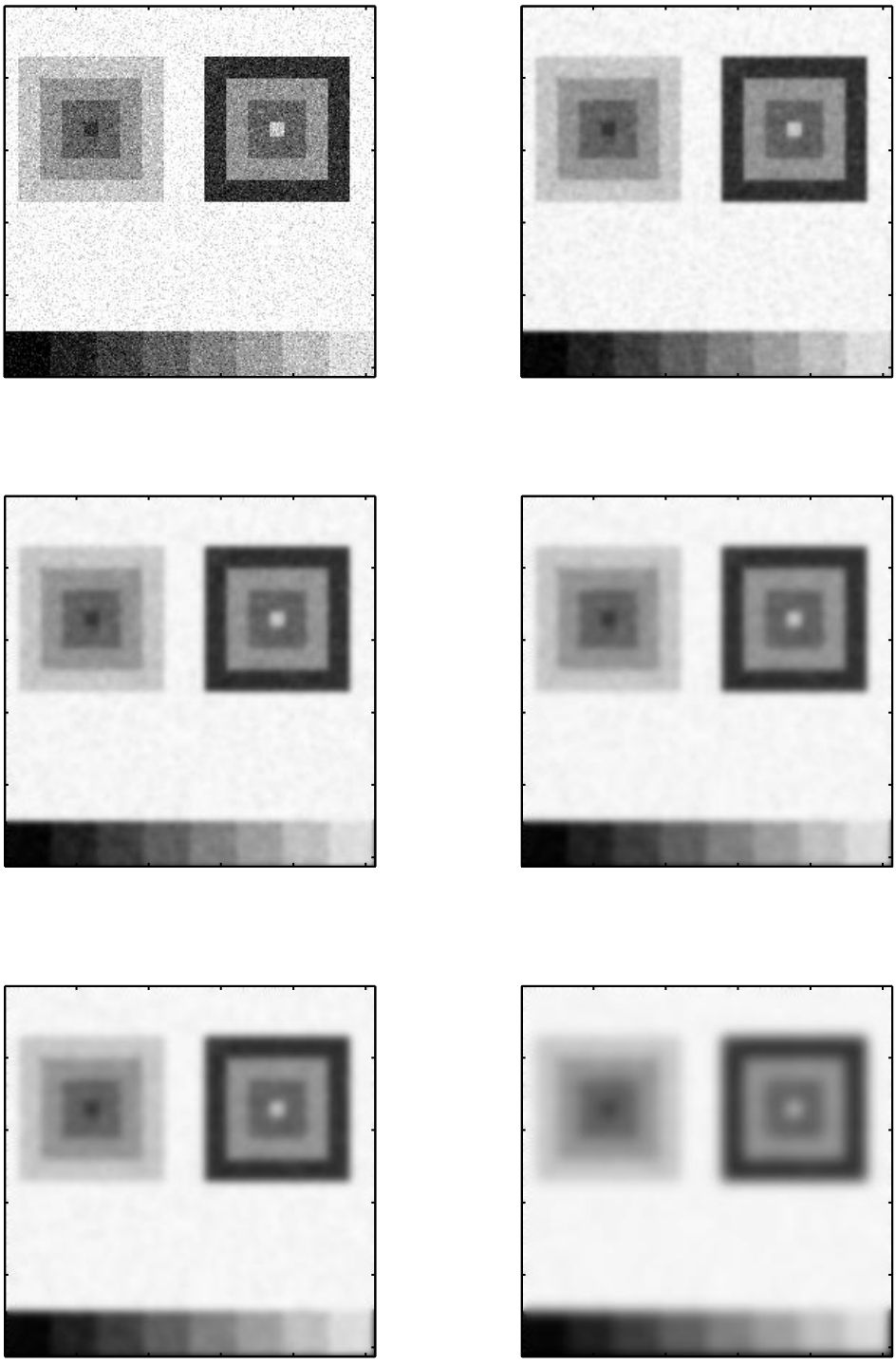
Les propriétés de régularisation de la convolution par G_σ s'interprètent donc en termes de *diffusion isotrope* et le temps t est lié à la l'échelle spatiale σ par $t = \sigma^2/2$.

La diffusion isotrope s'obtient en prenant une moyenne glissante pondérée qui a tendance à éliminer les pixels bruités mais qui rend flou les contours des objets de l'image.

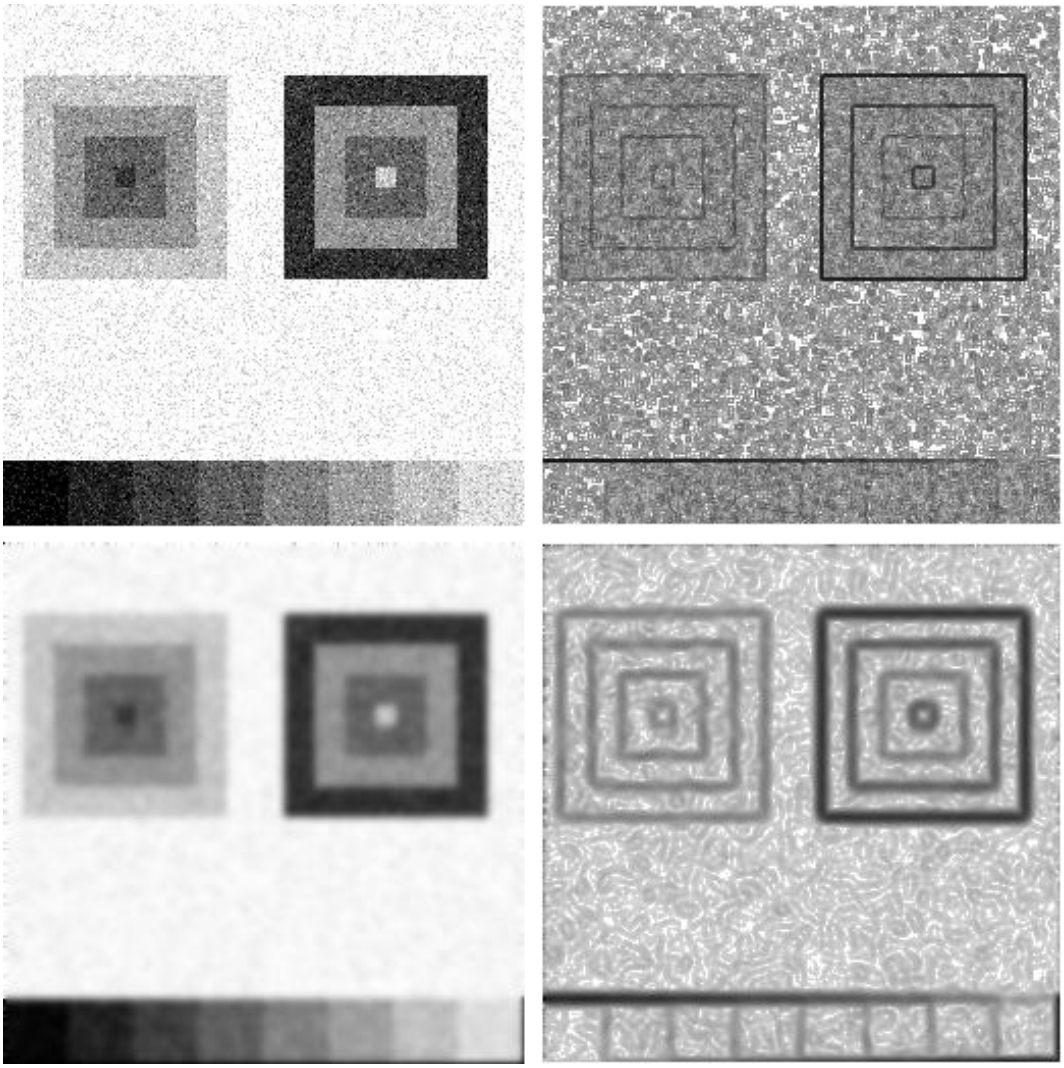
Sur la figure de la page suivante on a représenté quelques étapes de la diffusion isotrope d'une image bruité.

Pour éviter la destruction de contours on préfère appliquer une *diffusion anisotrope*, c'est-à-dire que l'on va lisser l'image partout, sauf au-delà des bords d'objets. Cette méthode nécessite donc deux étapes. On doit d'abord détecter les contours et autres structures fines de l'image, ensuite on va lisser l'image en ne considérant que des voisinages de pixels qui ne traversent pas les bords.

On modélise ce comportement par exemple par les équations aux dérivées partielles non linéaires (4.1) et (4.2), présentées plus loin.

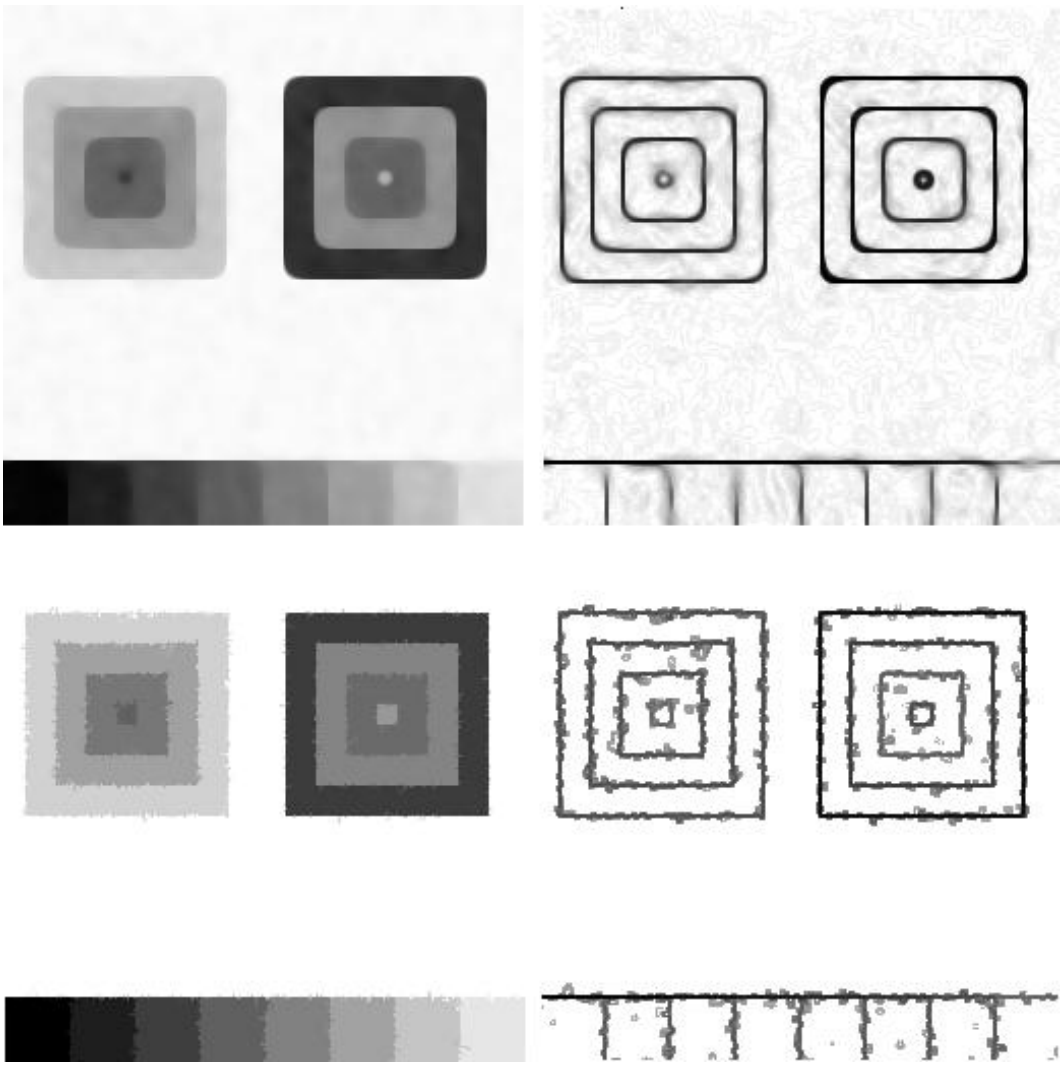


Itérations du Laplacien sur une image bruitée
De haut en bas et de gauche à droite : $t = i\,\Delta t$ pour $i = 0, \ldots, 5$.



En haut à gauche une image bruité u et en haut à droite une détection de bord par gradient discret $|\nabla u|$. Ici un gradient fort est représenté en noir et un gradient nul est blanc (*i.e.* inversion vidéo).

En bas à gauche on a appliqué l'équation de la chaleur à u : on calcule la solution de l'équation aux dérivées partielles à l'instant t, ceci revient à faire la convolution de u avec $G_{\sqrt{2t}}$.
En appliquant le gradient à cette image lissée on obtient l'image de droite $|\nabla G_{\sqrt{2t}} * u|$. On remarquera l'épaisseur des bords obtenus.



En haut à gauche on a représenté un lissage obtenu par une technique non linéaire modélisée par l'équation aux dérivées partielles, dite de la *mean curvature motion*

$$\frac{\partial u}{\partial t} = |\nabla u| \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right) \qquad \text{avec } u(x,0) = u_o(x)\,. \tag{4.1}$$

En appliquant le gradient on remarque que les bords sont bien localisés : il reste peu de bruit mais les coins des contours sont arrondis.

En bas à gauche un résultat pour l'équation aux dérivées partielles

$$\frac{\partial u}{\partial t} = \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right) \qquad \text{avec } u(x,0) = u_o(x)\,. \tag{4.2}$$

L'image du gradient montre que le bruit dans les zones homogènes à complètement disparu mais les bords sont irréguliers.

Interprétation des équations aux dérivées partielles précédentes :

Dans le repère canonique (e_1, e_2) de $\mathbb{R}^2$ on note

$$\nabla^2 u(x,y) = \begin{pmatrix} u_{xx}(x,y) & u_{xy}(x,y) \\ u_{xy}(x,y) & u_{yy}(x,y) \end{pmatrix}$$

la matrice associée à la forme bilinéaire symétrique $\mathrm{d}^2 u_{(x,y)}$, c'est la matrice *Hessienne*.

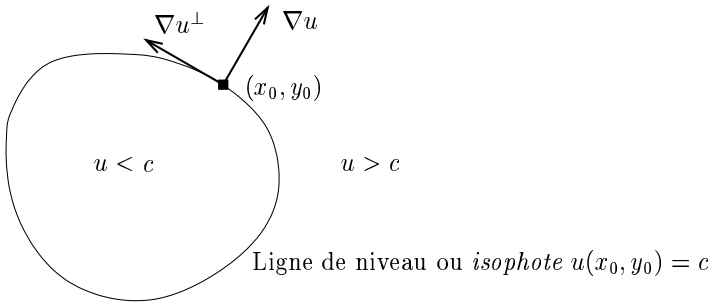
On rappelle que $\Delta u(x,y) = trace(\nabla^2 u(x,y))$ et $\nabla u(x,y) = \begin{pmatrix} u_x(x,y) \\ u_y(x,y) \end{pmatrix}$.

Pour η et ξ deux vecteurs unitaires quelconques de $\mathbb{R}^2$, on note

$$u_\eta \stackrel{\Delta}{=} \frac{\mathrm{d}u}{\mathrm{d}\eta} = \nabla u.\eta \qquad \text{et} \qquad u_{\eta\xi} \stackrel{\Delta}{=} \mathrm{d}^2 u(\eta,\xi) = \eta^t \, \nabla^2 u \, \xi \; .$$

En particulier pour $\eta = e_1$ et $\xi = e_2$ on retrouve les dérivées partielles d'ordre un et deux habituelles.

Supposons que $u \in \mathcal{C}^\infty(\mathbb{R}^2)$ et soit $c = u(x_0, y_0)$, pour $(x_0, y_0) \in \mathbb{R}^2$, tel que $u^{-1}(c)$ ne contient pas de points critiques (par le théorème de Sard ceci est vrai pour presque tout $c \in \mathbb{R}$). En particulier $|\nabla u(x_0, y_0)| \neq 0$, le théorème d'inversion locale entraîne alors que par le point (x_0, y_0) il passe une ligne de niveau unique $I_c = \{(x,y) \,/\, u(x,y) = c\}$.
L'ensemble d'intensité lumineuse constante I_c est appelé *isophote*. Sur la figure suivante on a représenté une isophote fermé.



En tout point (x,y) de I_c on définit un repère local orthonormé

$$(\eta, \xi)_{|(x,y)} = \left(\frac{\nabla u}{|\nabla u|}, \frac{\nabla u^\perp}{|\nabla u|} \right)_{|(x,y)} ,$$

où $\nabla u^\perp = \begin{pmatrix} -u_y \\ u_x \end{pmatrix}$ est le vecteur perpendiculaire direct à ∇u en (x,y).
Dans le repère local (η, ξ) on a $u_\xi = \nabla u.\xi = 0$ car u est constante sur I_c, tandis que η indique la direction de plus grande croissance de u, $u_\eta = |\nabla u|$.
Un calcul montre qu'au point $(x,y) \in I_c$:

$$\mathrm{div}\left(\frac{\nabla u}{|\nabla u|} \right) = \frac{1}{|\nabla u|} \left(\Delta u - \mathrm{d}^2 u \left(\frac{\nabla u}{|\nabla u|}, \frac{\nabla u}{|\nabla u|} \right) \right)$$

$$= \frac{u_y^2 u_{xx} + u_x^2 u_{yy} - 2 u_x u_y u_{xy}}{|\nabla u|^3} = \frac{1}{|\nabla u|} \mathrm{d}^2 u \left(\frac{\nabla u^\perp}{|\nabla u|}, \frac{\nabla u^\perp}{|\nabla u|} \right) .$$

En particulier on remarque que $\Delta u = \mathrm{d}^2u(\eta,\eta) + \mathrm{d}^2u(\xi,\xi)$, on retrouve l'invariance du Laplacien et l'équation de la chaleur s'écrit

$$u_t = u_{\eta\eta} + u_{\xi\xi} \quad .$$

Dans les coordonnées locales (η,ξ) l'équation (4.1) s'écrit

$$u_t = u_{\xi\xi} \quad ;$$

tandis que (4.2) devient

$$u_t = \frac{u_{\xi\xi}}{u_\eta} \quad .$$

On a donc une diffusion dans la direction perpendiculaire au gradient, on ne lisse pas les contours dans une image !

Un modèle qui permet de prendre en compte en chaque point les deux directions η et ξ a été proposé par MALIK et PERONA :

$$\frac{\partial u}{\partial t} = \mathrm{div}\Big(g(|\nabla u|)\,\nabla u\Big) \qquad \text{avec } u(x,0) = u_o(x)\,, \tag{4.3}$$

où le coefficient de diffusion $g(r)$, $r \geq 0$, est une fonction régulière positive décroissante avec $g(0) = 1$. Dans le repère local (η,ξ) l'équation (4.3) devient :

$$u_t = (g(u_\eta) + u_\eta g'(u_\eta))\,u_{\eta\eta} + g(u_\eta)\,u_{\xi\xi} = G'(u_\eta)\,u_{\eta\eta} + g(u_\eta)\,u_{\xi\xi} \quad ,$$

où on note $G(r) = rg(r)$ l'intensité du flux.
Au points où $|\nabla u| = u_\eta = 0$ on obtient l'équation de la chaleur isotrope, pour les autres points on obtient une diffusion anisotrope pondérée par les valeurs de g et G'.
Si $G'(u_\eta) = 0$ on a uniquement une diffusion dans la direction perpendiculaire au gradient ;
si $G'(u_\eta) > 0$ on ajoute une diffusion dans la direction du gradient ;
pour $G'(u_\eta) < 0$ on a une diffusion inversée instable dans la direction η.

Un exemple classique de coefficient de diffusion est $g(r) = \dfrac{1}{\lambda^2 + r^2}$, $\lambda \in \mathbb{R}_+^*$,
dans ce cas $G'(r) = \dfrac{\lambda^2 - r^2}{(\lambda^2 + r^2)^2}$ et λ est le seuil qui décide du signe de G'.

Note : Les équations aux dérivées partielles (4.1),(4.2) et (4.3) sont non linéaires et leur étude ne fera pas l'objet de ce cours.

4.1.2 Classification des e.d.p. linéaires d'ordre 2

Dans ce qui suit on suppose $u \in \mathcal{C}^2(\Omega)$, Ω ouvert de $\mathbb{R}^d$, nous allons d'abord rappeler quelques définitions dans le cas particulier des équations aux dérivées partielles d'ordre 2.
Une équation aux dérivées partielles est *quasilinéaire* si elle est linéaire en toutes les dérivées partielles d'ordre le plus élevé, pour une équation d'ordre 2 on a :

$$\sum_{1\le i,j\le d} a_{ij}(x,u,\nabla u)\,\frac{\partial^2 u}{\partial x_i \partial x_j}(x) = B(x,u,\nabla u)\,.$$

Note : comme u est de classe $\mathcal{C}^2$ on peut supposer $a_{ij} = a_{ji}$, pour tout $1 \le i,j \le d$.

Une équation aux dérivées partielles quasilinéaire est *semilinéaire* si les coefficients des dérivées partielles d'ordre le plus élevé ne dépendent que de x, une équation semilinéaire d'ordre 2 est de la forme

$$\sum_{1\le i,j\le d} a_{ij}(x)\,\frac{\partial^2 u}{\partial x_i \partial x_j}(x) = B(x,u,\nabla u)\,.$$

Une équation aux dérivées partielles est *linéaire* si elle est linéaire par rapport à u et ses dérivées partielles, une équation aux dérivées partielles linéaire d'ordre 2 s'écrit en tout point x de Ω :

$$\sum_{1\le i,j\le d} a_{ij}(x)\,\frac{\partial^2 u}{\partial x_i \partial x_j}(x) + \sum_{i=1}^{d} b_i(x)\,\frac{\partial u}{\partial x_i}(x) + c(x)\,u(x) = d(x)\,, \tag{4.4}$$

Nous allons maintenant introduire une classification plus spécifique des équations aux dérivées partielles linéaires d'ordre 2.
On définit le *symbole* de l'équation aux dérivées partielles (4.4), en $x \in \Omega$ et pour tout $\xi = (\xi_1,\ldots,\xi_d) \in \mathbb{R}^d$, par

$$\Lambda_x(\xi) = \sum_{1\le i,j\le d} a_{ij}(x)\,\xi_i\xi_j\,.$$

Pour tout $x \in \Omega$, Λ_x est une forme quadratique de matrice associée $A(x) = \Big(a_{ij}(x)\Big)_{1\le i,j\le d}$.

Or, grâce à un changement de variables adapté, une forme quadratique réelle peut être réduite à sa forme canonique :

$$\sum_{i=1}^{m} \eta_i^2 - \sum_{i=m+1}^{r} \eta_i^2\,,$$

où $1 \le m \le r \le d$. Donc, en un point $x_* \in \Omega$ donné, il existe un changement de variables, $x = P_{(x_*)}\,\tilde{x}$, tel que l'équation aux dérivées partielles (4.4) s'écrit sous la forme

$$\sum_{i=1}^{m} \frac{\partial^2 u}{\partial \tilde{x}_i^2}(\tilde{x}_*) - \sum_{i=m+1}^{r} \frac{\partial^2 u}{\partial \tilde{x}_i^2}(\tilde{x}_*) + \sum_{i=1}^{d} \tilde{b}_i(\tilde{x}_*)\,\frac{\partial u}{\partial \tilde{x}_i}(\tilde{x}_*) + \tilde{c}(\tilde{x}_*)\,u(\tilde{x}_*) = \tilde{d}(\tilde{x}_*)\,.$$

DÉFINITION 4.1.1
On dit que l'équation aux dérivées partielles (4.4) *est :*

- *elliptique au point* x_* *si* $r = d$ *et* $m = 0$ *ou* $m = d$, i.e. $A(x_*)$ *est définie positive ou négative et a* d *valeurs propres réelles non nulles de même signe ;*

- *hyperbolique au point* x_* *si* $r = d$ *et* $m = 1$ *ou* $m = d - 1$, i.e. $A(x_*)$ *est indéfinie et a* d *valeurs propres réelles non nulles dont une de signe contraire aux* $d - 1$ *autres ;*

- *parabolique au point* x_* *si* $r = d - 1$ *et* $m = 0$ *ou* $m = d - 1$, i.e. $A(x_*)$ *a 0 comme valeur propre et* $d - 1$ *valeurs propres réelles non nulles de même signe.*

Exemples :

1. L'équation de LAPLACE dans $\mathbb{R}^d$ est elliptique en tout point :
$$\forall x \in \mathbb{R}^d, \xi \in \mathbb{R}^d : \Lambda_x(\xi) = \sum_{i=1}^{d} \xi_i^2 .$$
2. L'équation de la chaleur est parabolique en tout point :
$$\forall (x,t) \in \mathbb{R}^d \times \mathbb{R}_+^*, \xi \in \mathbb{R}^{d+1} : \Lambda_{(x,t)}(\xi) = \sum_{i=1}^{d} \xi_i^2 .$$
3. L'équation des ondes est hyperbolique en tout point :
$$\forall (x,t) \in \mathbb{R}^d \times \mathbb{R}_+^*, \xi \in \mathbb{R}^{d+1} : \Lambda_{(x,t)}(\xi) = \sum_{i=1}^{d} \xi_i^2 - \xi_{d+1}^2 .$$
4. Pour l'équation de TRICOMI dans $\mathbb{R}^2$: $x_2\, u_{x_1 x_1} + u_{x_2 x_2} = 0$, on a
$$\forall x \in \mathbb{R}^2, \xi \in \mathbb{R}^2 : \Lambda_x(\xi) = x_2 \xi_1^2 + \xi_2^2 .$$
Elle est donc elliptique si $x_2 > 0$, hyperbolique si $x_2 < 0$ et parabolique si $x_2 = 0$.
5. L'équation $u_{x_1 x_2} = 0$ est hyperbolique dans $\mathbb{R}^2$.

Remarques :

1. Pour tout point $x_* \in \mathbb{R}^d$ on a une transformation qui réduit (4.4) sous forme canonique or cette transformation dépend de x_*. Pour $d \geq 3$ il est en général impossible de trouver un difféomorphisme Φ qui permet d'écrire l'équation aux dérivées partielles sous forme réduite sur une région donnée (arbitrairement petite).
Dans le cas $d = 2$ et sous quelques hypothèses sur les coefficients de (4.4) on peut trouver Φ qui permet de mettre (4.4) sous forme réduite.
2. La classification ci-dessus a été donnée pour des équations aux dérivées partielles linéaires d'ordre deux. On retrouve ces mêmes dénominations pour des équations aux dérivées partielles plus générales, mais qui partagent les propriétés essentielles avec les équations aux dérivées partielles linéaires présentées dans ce chapitre.

3. On élargit la classe des équations aux dérivées partielles hyperboliques aux équations qui admettent des solutions de type ondes, *i.e.* on a une vitesse de propagation finie. Considérons par exemple le système d'équations aux dérivées partielles d'ordre un en t

$$U_t(x,t) + A\,U_x(x,t) = F(x,t) \quad \text{dans } \mathbb{R} \times \mathbb{R}_+^* \ ,$$

où $U = (\ u_1 \quad \cdots \quad u_N\ )^t$ est défini sur $\mathbb{R}\times\mathbb{R}_+$ et A est une matrice constante d'ordre $N \times N$. On a un système hyberbolique à coefficients constants si A est diagonisable sur $\mathbb{R}$.

Cette définition se généralise au cas où $A(x,t)$ et pour $U : \mathbb{R}^d \times \mathbb{R}_+ \to \mathbb{R}^N$.

L'équation de transport est une équation aux dérivées partielles hyperbolique d'ordre un en t.

4.2 Étude d'un problème hyperbolique modèle

Nous allons étudier la méthode des différences finies pour résoudre numériquement des équations aux dérivées partielles d'évolution linéaires d'ordre 1 en t.
Le représentant le plus simple qui nous servira d'exemple type pour illustrer les définitions et techniques proposées, est l'équation hyperbolique d'ordre un en t : l'équation de transport.

L'inconnue est une fonction u définie pour $(x,t) \in \mathbb{R} \times \mathbb{R}_+$. Le domaine de définition de u est discrétisé par la *grille* $G_{h,k} = \{(x_m, t_n) \,/\, x_m = mh, m \in \mathbb{Z}, t_n = nk, n \in \mathbb{N}\}$
avec h et k des réels strictement positifs que l'on choisira petits.
On dit que h est le *pas de discrétisation spatiale* et k le *pas de temps*.

La fonction u qui est définie pour les variables continues (x,t), prend la valeur $u_m^n = u(mh,nk)$ au point $(x_m,t_n) = (mh,nk)$ de la grille $G_{h,k}$.

Pour bien distinguer la *solution continue* u du résultat du schéma numérique, on va noter v la *solution numérique* qui est définie uniquement sur $G_{h,k}$: $(v_m^n)_{m\in\mathbb{Z},n\in\mathbb{N}}$.

Donc v_m^n est la solution approchée, au point (x_m,t_n), de l'équation aux dérivées partielles.

Pour obtenir un problème discret on remplace les dérivées partielles par des *différences finies*, ainsi :

- $\dfrac{\partial u}{\partial x}(x_m,t_n) \simeq \dfrac{u_{m+1}^n - u_m^n}{h}$, différence finie progressive ;
- $\dfrac{\partial u}{\partial x}(x_m,t_n) \simeq \dfrac{u_m^n - u_{m-1}^n}{h}$, différence finie rétrograde ;
- $\dfrac{\partial u}{\partial x}(x_m,t_n) \simeq \dfrac{u_{m+1}^n - u_{m-1}^n}{2h}$, différence finie centrée ;
- $\dfrac{\partial^2 u}{\partial x^2}(x_m,t_n) \simeq \dfrac{u_{m+1}^n - 2u_m^n + u_{m-1}^n}{h^2}$, différence finie centrée d'ordre deux .

Ces approximations sont obtenus grâce à la formule de Taylor. On fait de même pour les dérivées partielles par rapport à t :

- $\dfrac{\partial u}{\partial t}(x_m,t_n) \simeq \dfrac{u_m^{n+1} - u_m^n}{k}$, différence finie progressive en t .

Exemple :

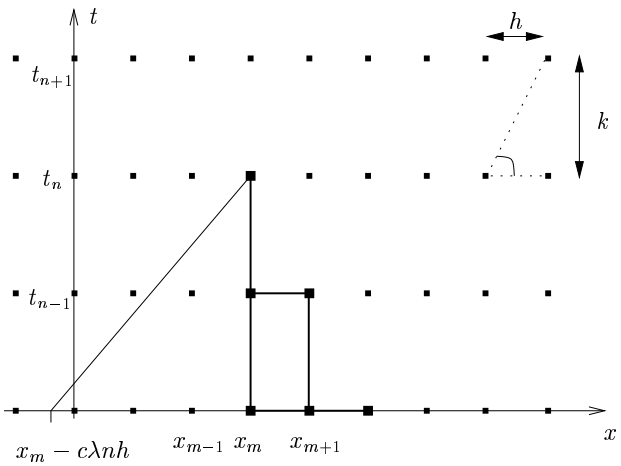
Considérons l'équation de transport $u_t + cu_x = 0$ dans $\mathbb{R} \times \mathbb{R}_+^*$, avec la condition initiale $u(x,0) = \Phi(x)$ pour $x \in \mathbb{R}$ et on suppose que $c > 0$.
On va discrétiser cette équation aux dérivées partielles en utilisant un schéma progressif en espace et en temps :

$$\frac{v_m^{n+1} - v_m^n}{k} + c\,\frac{v_{m+1}^n - v_m^n}{h} = 0\,. \tag{4.5}$$

En posant $\lambda = k/h$ on obtient : $v_m^{n+1} = (1+\lambda c)v_m^n - \lambda c v_{m+1}^n = (1 + \lambda c - \lambda c \mathcal{T}_{-h})\, v_m^n$,
où $\mathcal{T}_\alpha$ est l'opérateur de translation spatiale défini par $\mathcal{T}_\alpha u(x,t) = u(x-\alpha,t)$ pour tout $\alpha \in \mathbb{R}$, *i.e.* $\mathcal{T}_{-h} v_m^n = v_{m+1}^n$. On a par récurrence :

$$\begin{aligned} v_m^n &= (1 + \lambda c - \lambda c\, \mathcal{T}_{-h})^n\, v_m^0 \\ &= \sum_{p=0}^{n} C_n^p\, (1+\lambda c)^{n-p}\, (-\lambda c\, \mathcal{T}_{-h})^p\, \Phi(x_m) \\ &= \sum_{p=0}^{n} C_n^p\, (1+\lambda c)^{n-p}\, (-\lambda c)^p\, \Phi(x_m + ph)\,. \end{aligned}$$

Ainsi le domaine de dépendance de v au point $(x_m, t_n) = (ih, nk)$ est formé par les points x_m, $x_m + h$, $x_m + 2h$, ... , $x_m + nh$. Or, on sait que $u(x,t) = \Phi(x - ct)$ et que le domaine de dépendance de $u_m^n = u(x_m, t_n)$ est restreint au point $x_m - c\lambda nh$. Le schéma discret ne tient donc pas compte des propriétés de la solution de l'équation aux dérivées partielles et la solution discrète v_m^n est en général différente de u_m^n : le schéma (4.5) ne peut pas être convergent.



Supposons de plus que Φ est obtenu de façon expérimentale et que l'on mesure par exemple $\widetilde{\Phi}(x_m + ph) = \Phi(x_m + ph) + (-1)^p \varepsilon$, *i.e.* on commet une erreur de ε sur $|\Phi|$.
Dans ce cas, l'erreur sur v_m^n est de l'ordre de $(1 + 2\lambda c)^n \varepsilon$ et, pour λ fixé, l'erreur sur v_m^n croît de façon exponentielle avec le nombre d'itérations n. On dit que le schéma (4.5) est *instable.*

On constate sur cet exemple que, bien qu'il semble facile de remplacer les dérivées partielles par des différences finies, il faut toujours garantir que la solution donnée par un schéma aux différences finies converge, en un sens à définir, vers une solution de l'équation aux dérivées partielles.

Remarques :

1. Dans notre discussion du schéma discret (4.5) on n'a pas tenu compte d'éventuelles conditions au bord, pour un schéma discret celles-ci peuvent être de deux types.
   Supposons que l'on considère l'équation de transport sur $\mathbb{R}_+ \times \mathbb{R}_+^*$ uniquement.
   Pour pouvoir résoudre l'équation aux dérivées partielles on se donne une condition en $x = 0$ de la forme $u(0,t) = \Psi(t)$ pour $t \geq 0$, ceci entraîne pour le schéma discret que $v_0^n = \Psi(t_n)$ $(n \geq 0)$.
   De plus, étant donnée la mémoire limité d'un ordinateur, on est obligé de se restreindre à une grille spatiale finie $\{x_0, x_1, \ldots, x_M\}$. En x_M on introduit alors des contraintes de la forme $v_M^{n+1} = v_{M-1}^{n+1}$ ou $v_M^{n+1} = v_{M-1}^n$, ...
   Ces *conditions au bord numériques* sont nécessaires pour pouvoir déterminer v_m^{n+1} pour $0 \leq m \leq M$. Dans ce qui suit nous n'insisterons pas sur les problèmes posés par ces contraintes.
2. Les schémas aux différences finies permettent de traiter facilement des problèmes en une, deux ou trois dimensions spatiales pour des domaines à géométrie simple et avec des propriétés (p.ex. densité, vitesse, ...) qui varient lentement dans le domaine de définition. Dès que la géométrie devient complexe ou que les propriétés changement rapidement, on préférera la méthode des éléments finis.
   En effet, la méthode des éléments finis permet de traiter des géométries complexes (*e.g.* bords internes) et la triangulation peut être localement raffinée pour traiter des variations rapides de la solution. Par contre, il est en général plus compliqué d'écrire un code pour les éléments finis que pour les schémas aux différences finies.

4.3 Convergence, consistance et stabilité

4.3.1 Définitions

Dans toute la suite L est un opérateur différentiel linéaire qui est d'ordre un en t et on suppose que le problème

$$\begin{cases} Lu(x,t) = f(x,t) & \text{dans } \Omega \times \mathbb{R}_+^* \\ u(x,0) = \Phi(x) & \text{pour } x \in \Omega \text{ ouvert de } \mathbb{R} \end{cases} \tag{4.6}$$

est bien posé.

En remplaçant les dérivées partielles par des différences finies on obtient un opérateur discret $L_{h,k}$. Ainsi l'équation aux dérivées partielles homogène discrétisée s'écrit sous la forme $L_{h,k}v = 0$. Pour tenir compte du second membre on utilise un opérateur discret $I_{h,k}$ que l'on applique à f.

Un schéma aux différences finies général, associé à l'équation aux dérivées partielles (4.6), s'écrit donc comme suit :

$$L_{h,k}v = I_{h,k}f \ .$$

Dans la suite on choisira $I_{h,k}$ toujours de façon à avoir $I_{h,k}f = f_m^n$, de plus la condition initiale sera simplement donnée par $v_m^0 = \Phi(x_m)$, pour $x_m \in \Omega$.

Une première classification des schémas discrets est donné par la définition suivante.

DÉFINITION 4.3.1

- *Un schéma aux différences finies est explicite si on peut écrire v_m^{n+1} comme combinaison linéaire finie des v_i^j pour $j \leq n$.*
  Le schéma est dit implicite si d'autres valeurs de v sont nécessaires (p.ex. $v_{m\pm1}^{n+1}$).

- *Un schéma aux différences finies est à un pas de temps s'il utilise les valeurs de v à deux instants seulement, par exemple en t_n et t_{n+1}.*
  Le schéma est à pas multiples si la valeur de v à plus de deux instants intervient.

Un schéma à un pas construit $(v_m^n)_m$, pour tout $n \geq 1$, à partir des conditions initiales $v_m^0 = \Phi(x_m)$. Pour initialiser un schéma à J pas, les v_m^0 ne sont pas suffisants, il faut fournir $(v_m^j)_m$ pour J instants.

Dans la section précédente on a montré qu'il faut s'assurer que la solution discrète obtenue par un schéma aux différences finies donne une représentation correcte de la solution de l'équation aux dérivées partielles, d'où la

DÉFINITION 4.3.2 (CONVERGENCE)
Soit $u(x,t)$ la solution de (4.6) et v une solution du schéma discret $L_{h,k}v = f_m^n$ telle que v_m^0 converge vers $\Phi(x)$ quand x_m converge vers x.
On dit que le schéma aux différences finies $L_{h,k}$ est un schéma convergent si v_m^n converge vers $u(x,t)$ quand (x_m, t_n) converge vers (x,t) pour (h,k) tendant vers $(0,0)$.

La convergence exprime que la solution d'un schéma aux différences finies converge vers la solution de l'équation aux dérivées partielles. Avant de lancer des calculs, il faut s'assurer que le schéma utilisé est convergent, or ceci est en général difficile à démontrer. On va proposer dans la suite des critères plus facile à vérifier.

Remarques :

1. La définition ci-dessus et celles qui suivent, ne concernent que les schémas à un pas de temps. Pour des méthodes à pas multiples il faudra tenir compte de l'étape d'initialisation.
2. La définition précise de la convergence de v_m^n vers u, cohérente avec la suite, entraîne l'utilisation d'un opérateur d'interpolation : pour n fixé, on associe à $(v_m^n)_m \in l^2(h\mathbb{Z})$ un élément de $L^2(\mathbb{R})$. On impose ensuite une convergence dans $L^2(\mathbb{R})$ de l'interpolée vers la solution u.

DÉFINITION 4.3.3 (CONSISTANCE)
On dit que le schéma $L_{h,k}v = f_m^n$ est consistant avec l'équation aux dérivées partielles $Lu = f$ si pour toute fonction φ de classe $\mathcal{C}^\infty$ on a, en tout point (x_m, t_n) :

$$\lim_{(h,k)\to(0,0)} (L\varphi - L_{h,k}\varphi) = 0 .$$

La consistance entraîne en particulier qu'une solution régulière de l'équation aux dérivées partielles est une solution du schéma aux différences finies quand les pas de discrétisation tendent vers 0. C'est une condition nécessaire de convergence mais ce n'est pas une condition suffisante.

En effet, considérons de nouveau le schéma (4.5) et soit φ une fonction de classe $\mathcal{C}^\infty$.

On a $L\varphi = \varphi_t + c\varphi_x$ et $L_{h,k}\varphi = \dfrac{\varphi_m^{n+1} - \varphi_m^n}{k} + c\,\dfrac{\varphi_{m+1}^n - \varphi_m^n}{h}$.

Grâce à la formule de TAYLOR :

$$\varphi_{m+1}^n = \varphi(x_m + h, t_n) = \varphi_m^n + h\varphi_x(x_m,t_n) + \frac{h^2}{2}\varphi_{xx}(x_m,t_n) + O(h^3) ,$$

$$\varphi_m^{n+1} = \varphi(x_m, t_n + k) = \varphi_m^n + k\varphi_t(x_m,t_n) + \frac{k^2}{2}\varphi_{tt}(x_m,t_n) + O(k^3) .$$

On en déduit que

$$L_{h,k}\varphi(x_m,t_n) = \varphi_t(x_m,t_n) + c\,\varphi_x(x_m,t_n) + \frac{1}{2}\left(k\varphi_{tt}(x_m,t_n) + c\,h\varphi_{xx}(x_m,t_n)\right) + O(h^2) + O(k^2)$$

et donc

$$(L\varphi - L_{h,k}\varphi)(x_m,t_n) = -\frac{1}{2}\left(k\varphi_{tt}(x_m,t_n) + c\,h\varphi_{xx}(x_m,t_n)\right) + O(h^2) + O(k^2) \underset{(h,k)\to(0,0)}{\longrightarrow} 0\,.$$

Le schéma (4.5) est donc consistant bien que non convergent.

On avait constaté que ce schéma est en un certain sens instable. Nous allons donner une définition précise de stabilité pour une équation aux dérivées partielles homogène d'ordre un en t dans la

DÉFINITION 4.3.4 (STABILITÉ)
Le schéma aux différences finies $L_{h,k}v = 0$, associé à l'équation aux dérivées partielles $Lu = 0$, est stable s'il existe $\Lambda \subset (\mathbb{R}_+^*)^2$, avec $(0,0) \in \overline{\Lambda}$, tel que pour tout $T > 0$ il existe une constante C_T tel que

$$h \sum_{m=-\infty}^{+\infty} |v_m^n|^2 \;\le\; C_T h \sum_{m=-\infty}^{+\infty} |v_m^0|^2\,,$$

pour tous $0 \le t_n \le T$ et $(h,k) \in \Lambda$.
On dit que Λ est la région de stabilité du schéma.

Un exemple fréquent pour Λ est le segment $\{(h,\lambda h)\,/\,0 < h < C\}$, avec C et λ des constantes positives; le cône $\{(h,k)\,/\,0 < C_1 h \le k \le C_2 h < C_3\}$, avec C_1, C_2 et C_3 des constantes positives est un autre exemple.

Remarque : En utilisant le principe de DUHAMEL on montre qu'un schéma $L_{h,k}v = f_m^n$ est stable si le schéma homogène $L_{h,k}v = 0$ est stable. De plus on peut étendre la définition au cas d'équations d'ordre 2 en t et au cas de schémas à pas multiples.

La définition de stabilité utilise une norme sur l'espace $l^2(h\mathbb{Z})$ (voir section 4.4) :

$$(v_m^n)_{m\in\mathbb{Z}} \text{ est un élément de } l^2(h\mathbb{Z}) \text{ si } \|v^n\|_{l^2(h\mathbb{Z})} = \left(h \sum_{m=-\infty}^{+\infty} |v_m^n|^2\right)^{1/2} \text{ est fini.}$$

Le critère de stabilité s'écrit alors, pour h et k suffisamment petits dans Λ :

$$(\forall T > 0)(\exists C_T)(\forall t_n \in [0,T]) \;:\qquad \|v^n\|_{l^2(h\mathbb{Z})} \le C_T\,\|v^0\|_{l^2(h\mathbb{Z})}\,. \tag{S}$$

La stabilité garantit qu'à chaque instant $t_n \in [0,T]$, la norme de la solution discrète est bornée, à un facteur constant près, par la norme des données initiales.

Cette *stabilité numérique* n'est pas à confondre avec la stabilité du problème bien posé (4.6). Cette dernière concerne le comportement à l'infini de la solution en fonction des conditions initiales, par contre la stabilité numérique d'un schéma concerne le comportement de v sur l'intervalle $[0,T]$ quand (h,k) tend vers $(0,0)$.

Remarques :

1. Il existe d'autres façons de définir la stabilité numérique, on a choisi la définition particulière ci-dessus parce qu'elle s'applique à un grand nombre de schémas linéaires et qu'elle est adaptée à l'analyse de stabilité de VON NEUMANN.

2. Pour des équations aux dérivées partielles et schémas discrets non linéaires on utilisera d'autres espaces, *e.g.* L^1 , et d'autres propriétés, *e.g.* la diminution de la variation totale ou la monotonie d'un schéma, afin d'éviter des oscillations de v là où la solution de l'équation aux dérivées partielles varie rapidement, *e.g.* discontinuités.

L'importance des notions de consistance et stabilité est due au

THÉORÈME 4.3.1 (ÉQUIVALENCE DE LAX-RICHTMYER)
Un schéma linéaire, consistant avec le problème (4.6) *, est convergent si et seulement si il est stable.*

La dernière définition que l'on va introduire permet de distinguer les schémas convergents, en effet deux schémas convergents n'ont pas nécessairement les mêmes performances pour l'approximation de la solution continue.

DÉFINITION 4.3.5
Un schéma $L_{h,k}v = f_m^n$ *, consistant avec le problème* (4.6) *, est d'ordre* p *en espace et* q *en temps si, pour toute fonction* φ *de classe* $\mathcal{C}^\infty$ *:*

$$E_{h,k}\varphi = L_{h,k}\varphi - L\varphi = O(h^p) + O(k^q) \ .$$

On dit que le schéma est d'ordre (p, q) *et* $E_{h,k}$ *est l'erreur de troncature du schéma.*

Remarques :

1. L'erreur de troncature permet d'évaluer l'erreur introduite lorsque l'on remplace l'opérateur continu L par l'opérateur discret $L_{h,k}$.
2. La consistance d'un schéma nécessite uniquement que $L_{h,k}\varphi - L\varphi = o(1)$.

Exemple :

Sans démontrer le théorème de LAX-RICHTMYER nous allons illustrer, grâce à un exemple, comment la stabilité et la consistance interviennent dans la convergence d'un schéma discret.

Pour $c < 0$, considérons l'équation d'advection $u_t + cu_x = 0$ dans $\mathbb{R} \times \mathbb{R}_+^*$,
avec $u(x,0) = \Phi(x)$ pour $x \in \mathbb{R}$. On suppose de plus que Φ'' est bornée sur $\mathbb{R}$.

On va utiliser le schéma (4.5) : $v_m^{n+1} = (1+\lambda c)v_m^n - \lambda c v_{m+1}^n$, où $\lambda = k/h$.

Pour $x = mh$, posons $e(x,t_n) = u(x,t_n) - v_m^n$, c'est l'erreur commise par le schéma au point (x,t_n).

Alors $L_{h,k}e = L_{h,k}u = E_{h,k}u$, car v est solution de $L_{h,k}v = 0$ et u est solution de $Lu = 0$, d'où

$$e(x,t_{n+1}) = (1+\lambda c)\, e(x,t_n) - \lambda c\, e(x+h,t_n) + k\, E_{h,k}u(x,t_n)$$

et

$$\sup_x |e(x,t_{n+1})| \leq (|1+\lambda c| + |\lambda c|) \sup_x |e(x,t_n)| + k \sup_x |E_{h,k}u(x,t_n)|\,.$$

On a ainsi majoré l'erreur à l'instant t_{n+1} grâce à l'erreur à l'instant t_n et de l'erreur de troncature en t_n ; par récurrence

$$\sup_x |e(x,t_n)| \leq (|1+\lambda c|+|\lambda c|)^n \sup_x |e(x,0)| + k \sum_{p=1}^{n} (|1+\lambda c|+|\lambda c|)^{p-1} \sup_x |E_{h,k}u(x,t_{n-p})|\,.$$

Or comme l'on verra plus loin, l'analyse de VON NEUMANN permet de montrer que le schéma est stable si et seulement si λ est tel que $-1 \leq \lambda c \leq 0$, *i.e.* $|1+\lambda c| + |\lambda c| = 1$. Comme de plus $e(x,0) = 0$ pour tout x, on a

$$\sup_x |e(x,t_n)| \leq k \sum_{i=0}^{n-1} \sup_x |E_{h,k}u(x,t_i)|\,.$$

On doit maintenant évaluer l'erreur de troncature. Or le schéma (4.5) est consistant et d'ordre $(1,1)$, de plus $E_{h,k}u(x,t_n) = L_{h,k}u(x,t_n) = (L_{h,k}u - Lu)(x,t_n) = O(k)$ car $k = \lambda h$. On obtient ainsi une estimation de l'erreur de troncature qui dépend de u_{xx}, u_{tt} et k, pour obtenir une estimation globale et montrer la convergence du schéma on va utiliser l'hypothèse sur Φ. On a

$$|E_{h,k}u(x,t_i)| = |L_{h,k}\Phi(x-ct_i)| = \left|\frac{h}{2}c\,\Phi''(\zeta_1) + \frac{k}{2}c^2\,\Phi''(\zeta_2)\right| \leq k\,C \sup_\zeta |\Phi''(\zeta)|$$

et finalement, pour tout $n \leq T/k$:

$$\|e(.,t_n)\|_\infty = \sup_x |e(x,t_n)| \leq k\,n\,k\,C_{(\lambda,\Phi'',c)} \leq T\,C_{(\lambda,\Phi'',c)}\,k = C_{(\lambda,\Phi'',c,T)}\,k\,.$$

L'erreur commise en chaque point de la grille est d'ordre k et le schéma est convergent.

Remarque : Dans cet exemple nous avons montré une convergence en chaque point de la grille en utilisant la norme $\|.\|_\infty$ sur $h\mathbb{Z}$ et des hypothèses fortes sur $u(x,0)$.
Nous allons appliquer l'équivalence de LAX-RICHTMYER uniquement dans le cadre L^2, en effet on a alors, grâce à la méthode de VON NEUMANN, une méthode générale pour l'étude de la stabilité pour des schémas numériques associés à des équations aux dérivées partielles d'évolution linéaires à coefficients constants.
On montre par ailleurs que la régularité des données initiales a une influence sur la précision de la solution numérique.

4.3.2 Condition de Courant-Friedrichs-Lewy

Nous allons étudier différents schémas discrets pour le problème de transport avec la vitesse $c \in \mathbb{R}^*$:

$$\begin{cases} u_t + c\,u_x = 0 & \text{dans } \mathbb{R} \times \mathbb{R}_+^* \\ u(x,0) = \Phi(x) & \text{pour } x \in \mathbb{R} \end{cases} \tag{4.8}$$

• Le schéma progressif en temps et en espace (4.5) est de la forme : $v_m^{n+1} = \alpha v_m^n + \beta v_{m+1}^n$ et on a

$$\|v_\cdot^{n+1}\|_{l^2(h\mathbb{Z})}^2 = h \sum_{m\in\mathbb{Z}} |v_m^{n+1}|^2 = h \sum_{m\in\mathbb{Z}} |\alpha v_m^n + \beta v_{m+1}^n|^2 \le (|\alpha| + |\beta|)^2 \|v_\cdot^n\|_{l^2(h\mathbb{Z})}^2 .$$

Une condition nécessaire de stabilité de ces schémas est donc $|\alpha| + |\beta| \le 1$.
En particulier pour le schéma (4.5) on a $|1+c\lambda| + |c\lambda| \le 1$, ce schéma est donc stable pour $-1 \le c\lambda \le 0$. On va montrer plus loin que ceci est une condition suffisante de stabilité.

• Le schéma de LAX-FRIEDRICHS est centré en espace et progressif en temps mais utilise une moyenne centrée pour approximer v_m^n :

$$\frac{v_m^{n+1} - \frac{1}{2}(v_{m+1}^n + v_{m-1}^n)}{k} + c\,\frac{v_{m+1}^n - v_{m-1}^n}{2h} = 0 . \tag{4.9}$$

C'est un schéma consistant et qui est de la forme $v_m^{n+1} = \alpha v_{m+1}^n + \beta v_{m-1}^n$.
On montre que ces schémas sont stables si $|\alpha| + |\beta| \le 1$, en particulier le schéma de LAX-FRIEDRICHS (4.9) est stable si $|c\lambda| \le 1$.

Les résultats précédents se généralisent et on a le

THÉORÈME 4.3.2
Soit $v_m^{n+1} = \alpha v_{m-1}^n + \beta v_m^n + \gamma v_{m+1}^n$ un schéma explicite pour l'équation aux dérivées partielles (4.8). Alors, si le rapport $k/h = \lambda$ est constant, une condition nécessaire de stabilité du schéma est la condition de COURANT-FRIEDRICHS-LEWY (CFL) :

$$|c\lambda| \le 1 .$$

THÉORÈME 4.3.3
Il n'existe pas de schéma explicite, consistant et inconditionnellement stable pour l'équation aux dérivées partielles (4.8).

Remarque :

1. Les théorèmes précédents s'appliquent de façon plus générale aux systèmes d'équations aux dérivées partielles hyperboliques, définis à la page 41 .
2. La vitesse d'advection c détermine la pente des droites caractéristiques de (4.8), la condition (CFL) assure que le *domaine de dépendance discret contient le domaine de dépendance de l'équation aux dérivées partielles*, voir la figure page 42.

Exemple : Pour montrer que le dernier théorème ne s'applique pas aux schémas implicites considérons le schéma suivant, progressif en temps et rétrograde en espace, à l'instant t_{n+1} :

$$\frac{v_m^{n+1} - v_m^n}{k} + c\,\frac{v_m^{n+1} - v_{m-1}^{n+1}}{h} = 0\,.$$

En écrivant ce schéma sous la forme : $(1 + c\lambda)v_m^{n+1} = v_m^n + c\lambda v_{m-1}^{n+1}$
on montre que si $c > 0$, on a pour tout $\lambda \in \mathbb{R}_+$:

$$\|v_\cdot^{n+1}\|^2_{l^2(h\mathbb{Z})} \leq \|v_\cdot^n\|^2_{l^2(h\mathbb{Z})}\,.$$

Le schéma est *inconditionnellement stable* pour $c > 0$.

4.4 Analyse de stabilité de von Neumann

L'*analyse de stabilité de* VON NEUMANN utilise l'analyse de FOURIER pour étudier le comportement d'un schéma discret. Avant de montrer des applications à l'étude de schémas discrets, nous allons motiver l'utilisation de l'analyse de FOURIER grâce à une analogie et nous allons rappeler quelques résultats sur $l^2(h\mathbb{Z})$.

Motivations :

Si on considère que $v_\cdot^n$ est un signal numérique discret monodimensionnel on peut interpréter $v_\cdot^{n+1}$ comme la sortie d'un système discret $\mathcal{S}$: $v_\cdot^{n+1} = \mathcal{S}[v_\cdot^n]$.
Pour étudier le système $\mathcal{S}$ on introduit la transformée de FOURIER des signaux discrets v^n et v^{n+1}.
Ceci est particulièrement intéressant quand $\mathcal{S}$ est un système linéaire et invariant, un *filtre*, dans ce cas la sortie de $\mathcal{S}$ est obtenue grâce à la convolution de l'entrée du système avec la réponse impulsionnelle s de $\mathcal{S}$: $\mathcal{S}[e] = s \star e$. Un filtre peut aussi être représenté par une équation aux différences finies, récursive ou non.
Dans le domaine fréquentiel la convolution devient une multiplication de la transformée de FOURIER du signal en entrée avec la réponse fréquentielle $\widehat{s}$ de $\mathcal{S}$. Le signal sinusoïdal discret $e_\omega = (e^{i\omega m})_{m\in\mathbb{Z}}$ est un vecteur propre de $\mathcal{S}$: $\mathcal{S}[e_\omega] = \sqrt{2\pi}\,\widehat{s}(\omega)\,e_\omega$.

L'espace $l^2(h\mathbb{Z})$:

Nous rappelons ici brièvement quelques éléments de la structure de l'espace de HILBERT $l^2(h\mathbb{Z})$. Les définitions et résultats présentés s'obtiennent à partir de l'étude des séries de FOURIER des fonctions de $L^2([-\pi, +\pi])$: grâce à un changement de variable, on tient compte du pas de discrétisation spatiale $h > 0$.

Ainsi pour $v \in l^2(h\mathbb{Z})$ on définit la norme : $\|v\|_{l^2(h\mathbb{Z})} = \left(h \sum_{m=-\infty}^{+\infty} |v_m|^2 \right)^{1/2}$

et le produit scalaire sous-jacent, pour tout $v, w \in l^2(h\mathbb{Z})$:

$$(v, w)_{l^2(h\mathbb{Z})} = h \sum_{m=-\infty}^{+\infty} v_m w_m \ .$$

Soit $v \in l^2(h\mathbb{Z})$, on définit sa transformée de FOURIER, pour $\xi \in [-\pi/h, +\pi/h]$, par :

$$\widehat{v}(\xi) = \frac{1}{\sqrt{2\pi}} \sum_{m=-\infty}^{+\infty} h v_m e^{-imh\xi} \ .$$

La fonction $\widehat{v}$ est $2\pi/h-$périodique, c'est un élément de $L^2([-\pi/h, +\pi/h])$ et l'identité ci-dessus est à prendre au sens de la convergence en moyenne quadratique.
La formule de reconstruction s'écrit pour tout $m \in \mathbb{Z}$:

$$v_m = \frac{1}{\sqrt{2\pi}} \int_{-\frac{\pi}{h}}^{+\frac{\pi}{h}} \widehat{v}(\xi) e^{+imh\xi} \, d\xi \ ,$$

et on a l'identité de PARSEVAL :

$$\|\widehat{v}\|^2_{L^2([-\pi/h,\pi/h])} = \int_{-\frac{\pi}{h}}^{+\frac{\pi}{h}} |\widehat{v}(\xi)|^2 \, d\xi = \sum_{m=-\infty}^{+\infty} h|v_m|^2 = \|v\|^2_{l^2(h\mathbb{Z})} \ .$$

Pour $v \in l^2(h\mathbb{Z})$ on a : $\widehat{\mathcal{T}_{\pm h} v}(\xi) = e^{\mp ih\xi} \widehat{v}(\xi)$, où $\mathcal{T}_{\pm h} v^n_m = v^n_{m\mp 1}$;
une variation spatiale (translation) de v se traduit par un déphasage de $\widehat{v}$.

Grâce à la formule de PARSEVAL on obtient une forme équivalente pour le critère de stabilité (S) :

$$(\forall T > 0)(\exists C_T)(\forall t_n \in [0, T]) \ : \qquad \|\widehat{v^n_\cdot}\|_{L^2([-\pi/h,\pi/h])} \leq C_T \, \|\widehat{v^0_\cdot}\|_{L^2([-\pi/h,\pi/h]))} \ . \qquad \text{(S')}$$

Méthode de VON NEUMANN :

Considérons un schéma discret à un pas $L_{h,k} v = 0$, linéaire et à coefficients constants. En appliquant la transformée de FOURIER et en réarrangeant les termes on obtient la relation :

$$\widehat{v^{n+1}}(\xi) = g(\xi, h, k) \, \widehat{v^n}(\xi) \ .$$

Le facteur $g(\xi, h, k)$ est appelé *facteur d'amplification* : le module de g, $|g|$, est l'amplification et la phase de g, $\arg(g)$, est le déphasage que subit chaque fréquence de la solution discrète quand on avance d'un pas de temps k. Par itération on a :

$$\widehat{v^n}(\xi) = g(\xi, h, k)^n \, \widehat{v^0}(\xi) \ .$$

Remarques :

1. Pour déterminer le facteur d'amplification en pratique, on remplace dans le schéma discret les termes v^n_m par $g^n \, e^{imh\xi}$, *i.e.* on cherche des solutions de la forme $g^n \, e^{imh\xi}$.

2. Si u vérifie (4.8) et, pour tout $t \in \mathbb{R}_+$, $u(.,t) \in L^2(\mathbb{R})$, on a, en appliquant la transformée de FOURIER par rapport à x :
$\widehat{u_t}(\omega,t) = -c\,i\omega\,\widehat{u}(\omega,t)$, donc $\widehat{u}(\omega,t) = \widehat{u}(\omega,0)\,e^{-ict\omega} = \widehat{\Phi}(\omega)\,e^{-ict\omega}$ et

$$\widehat{u}(\omega,t+k) = e^{-ick\omega}\,\widehat{u}(\omega,t)\,.$$

Le facteur d'amplification, $g = |g|\,e^{i\arg(g)}$, du schéma, doit approximer $e^{-ick\omega}$: Si $|g| \neq 1$ on a une erreur d'amplitude dans la solution discrète, en particulier si $|g(\xi,h,k)| < 1$, on atténue la fréquence ξ et on parle de *dissipation*, resp. *viscosité*, *numérique* ; la différence de phase, $\arg(g) - ck\xi$, introduit une erreur de phase, appelée *dispersion numérique*.

Exemples :

1. Considérons l'équation de transport $u_t + c\,u_x = 0$, pour $c > 0$, discrétisée grâce au schéma progressif en temps et en espace (4.5).
On a $g(\xi,h,k) = 1 + c\lambda - c\lambda\,e^{ih\xi}$ qui, pour λ fixé, ne dépend que de $h\xi$, et

$$|g(h\xi)|^2 = g(h\xi)\,\overline{g(h\xi)} = 1 + 4c\lambda(1+c\lambda)\sin^2(h\xi/2)\,.$$

Donc $$\|\widehat{v^n}\|^2_{L^2([-\pi/h,\pi/h])} = |g(h\xi)|^{2n}\,\|\widehat{v^0}\|^2_{L^2([-\pi/h,\pi/h])}\,,$$

or $|g(h\xi)| = 1$ seulement si $h\xi = 0$; sinon $|g(h\xi)| > 1$, on va en déduire que le schéma est instable.

Soit $(h_0,k_0) \in (\mathbb{R}_+^*)^2$, et $K \in \mathbb{R}_+^*$:
alors il existe $(h,k) \in]0,h_0]\times]0,k_0]$ et $(\xi_1,\xi_2) \in]0,\pi/h[^2$ avec

$$|g(h\xi)| \geq 1 + Kk\,,\ \text{pour tout } \xi \in [\xi_1,\xi_2]\,.$$

On construit v^0 tel que $\widehat{v^0}(\xi) = \begin{cases} (\xi_2-\xi_1)^{-1/2} & \text{si } \xi \in [\xi_1,\xi_2] \\ 0 & \text{si } \xi \notin [\xi_1,\xi_2] \end{cases}$ et $\|v^0\|_{l^2(h\mathbb{Z})} = 1$.

Ainsi
$$\begin{aligned} \|v^n\|^2_{l^2(h\mathbb{Z})} &= \|\widehat{v^n}\|^2_{L^2([-\pi/h,\pi/h])} \\ &= \int_{-\frac{\pi}{h}}^{+\frac{\pi}{h}} |g(h\xi)|^{2n}\,|\widehat{v^0}(\xi)|^2\,d\xi \\ &= \int_{\xi_1}^{\xi_2} |g(h\xi)|^{2n}\,\frac{1}{\xi_2-\xi_1}\,d\xi \\ &\geq (1+Kk)^{2n} \geq \frac{1}{2}\,e^{2KT}\,\|v^0\|^2_{l^2(h\mathbb{Z})}\,, \end{aligned}$$

pour $n \simeq T/k$ et k_0 suffisamment petit.
Comme K peut être pris aussi grand que l'on veut, le critère de stabilité (S) n'est pas vérifié et le schéma (4.5) est instable, donc divergent, pour $c > 0$.

2. Considérons l'équation aux dérivées partielles $u_t + c\,u_x - u = 0$.
Ce problème d'advection-réaction est discrétisé grâce à (4.9) :

$$\frac{v_m^{n+1} - \frac{1}{2}(v_{m+1}^n + v_{m-1}^n)}{k} + c\,\frac{v_{m+1}^n - v_{m-1}^n}{2h} - v_m^n = 0\,.$$

Le facteur d'amplification est $g(\xi,h,k)=\cos(h\xi)+i\,c\lambda\sin(h\xi)+k$,
et $|g|^2=(\cos(h\xi)+k)^2+c^2\lambda^2\sin^2(h\xi)$.
On montre que $|g(\xi,h,k)|\le 1+k$ pour $|c|\lambda\le 1$, or pour $n\le T/k$:

$$\begin{aligned}\|v^n\|^2_{l^2(h\mathbb{Z})} &= \|\widehat{v^n}\|^2_{L^2([-\pi/h,\pi/h])}\\ &= |g(\xi,h,k)|^{2n}\,\|\widehat{v^0}\|^2_{L^2([-\pi/h,\pi/h])}\\ &\le (1+k)^{2n}\,\|v^0\|^2_{l^2(h\mathbb{Z})}\\ &\le (1+k)^{2T/k}\|v^0\|^2_{l^2(h\mathbb{Z})} \;\le\; e^{2T}\|v^0\|^2_{l^2(h\mathbb{Z})}.\end{aligned}$$

D'après le critère (S) on a la stabilité du schéma.
La solution de l'équation aux dérivées partielles s'écrit $u(x,t)=\Phi(x-ct)\,e^t$,
ainsi même si la condition initiale Φ est bornée sur $\mathbb{R}$, la solution u n'est pas bornée sur $\mathbb{R}\times\mathbb{R}_+$. En appliquant la transformée de Fourier en espace, on a $\widehat{u}(\omega,t)=\widehat{\Phi}(\omega)\,e^{ict\omega}e^t$ et donc $\widehat{u}(\omega,t+k)=\widehat{u}(\omega,t)\,e^{ick\omega}e^k$.
Afin de pouvoir suivre l'évolution de la solution $u(x,t)$ au cours du temps, il faut que $|g|$ soit plus grand que 1.

De façon générale on a le

Théorème 4.4.1
Un schéma à un pas $L_{h,k}v=0$, linéaire et à coefficients constants, est stable si et seulement si il existe une constante $K\in\mathbb{R}_+$ et une région de stabilité $\Lambda\subset(\mathbb{R}_+^*)^2$, avec $(0,0)\in\overline{\Lambda}$, tel que

$$|g(\xi,h,k)|\le 1+Kk\,,$$

pour tout ξ et tout $(h,k)\in\Lambda$.
Si le facteur d'amplification ne dépend que de $h\xi$ la condition nécessaire et suffisante de stabilité s'écrit :

$$|g(h\xi)|\le 1\,.$$

Exemple de Strikwerda :
On considère l'équation aux dérivées partielles $u_t+c\,u_{xxx}=f$,
où l'on discrétise u_t comme dans le schéma de Lax-Friedrichs (4.9) et u_{xxx} grâce aux différences centrées, pour obtenir :

$$\frac{v_m^{n+1}-\frac12(v_{m+1}^n+v_{m-1}^n)}{k}+c\,\frac{v_{m+2}^n-2v_{m+1}^n+2v_{m-1}^n-v_{m-2}^n}{2h^3}=f_m^n\,.$$

Soit φ une fonction régulière, alors $L\varphi=\varphi_t+c\,\varphi_{xxx}$ et

$$L_{h,k}\varphi=\frac{\varphi_m^{n+1}-\frac12(\varphi_{m+1}^n+\varphi_{m-1}^n)}{k}+c\,\frac{\varphi_{m+2}^n-2\varphi_{m+1}^n+2\varphi_{m-1}^n-\varphi_{m-2}^n}{2h^3}\,.$$

En utilisant la formule de Taylor en (x_m,t_n) :

$$\begin{aligned}\varphi_m^{n+1} &= \varphi_m^n+k\,\varphi_t+\frac{k^2}{2}\,\varphi_{tt}+O(k^3)\,,\\ \varphi_{m\pm1}^n &= \varphi_m^n\pm h\,\varphi_x+\frac{h^2}{2}\,\varphi_{xx}\pm\frac{h^3}{6}\,\varphi_{xxx}+\frac{h^4}{4!}\,\varphi_{xxxx}+O(h^5)\,,\\ \varphi_{m\pm2}^n &= \varphi_m^n\pm 2h\,\varphi_x+2h^2\,\varphi_{xx}\pm\frac{4h^3}{3}\,\varphi_{xxx}+\frac{2h^4}{3}\,\varphi_{xxxx}+O(h^5)\,.\end{aligned}$$

Donc

$$L_{h,k}\varphi = \varphi_t + \frac{k}{2}\,\varphi_{tt} - \frac{h^2}{k}\,\varphi_{xx} + O(k^2 + h^4/k) + c\varphi_{xxx} + O(h^2)$$

et

$$L_{h,k}\varphi - L\varphi = \frac{k}{2}\,\varphi_{tt} - \frac{h^2}{k}\,\varphi_{xx} + O(k^2 + h^4/k + h^2)\,.$$

Le schéma est donc consistant seulement si h^2/k tend vers 0 quand $h, k \longrightarrow 0$.
Supposons que $k = H(h)$, avec H régulière et $H(0) = 0$, dans ce cas $L_{h,k}\varphi - L\varphi = O(h)$ et on dit que le schéma est d'ordre 1.

Pour étudier la stabilité on écrit le schéma sous la forme :

$$v_m^{n+1} = \frac{1}{2}(v_{m+1}^n + v_{m-1}^n) - \frac{ck}{2h^3}(v_{m+2}^n - 2v_{m+1}^n + 2v_{m-1}^n - v_{m-2}^n)\,,$$

on obtient le facteur d'amplification

$$g(\xi, h, k) = \cos(h\xi) + i\,4ckh^{-3}\,\sin(h\xi)\sin^2(h\xi/2)\,.$$

On a $\quad |g(\xi, h, k)|^2 = \cos^2(h\xi) + 16c^2k^2h^{-6}\,\sin^2(h\xi)\sin^4(h\xi/2)\,,$
une condition nécessaire de stabilité est alors que la quantité $h^{-3}k$ reste borné.
Or ceci est incompatible avec la contrainte de consistance sur h^2k^{-1}. Le schéma ne peut donc pas être consistant et stable en même temps.

Schémas pour l'équation de transport.

Pour terminer on présente quelques schémas classiques pour la résolution du problème homogène (4.8) pour $c \in \mathbb{R}^*$:

• **Schéma centré en espace :**

Ce schéma s'écrit $\dfrac{v_m^{n+1} - v_m^n}{k} + c\,\dfrac{v_{m+1}^n - v_{m-1}^n}{2h} = 0$;
il est consistant et d'ordre $(2,1)$. En écrivant

$$v_m^{n+1} = v_m^n - \frac{c\lambda}{2}\,(v_{m+1}^n - v_{m-1}^n)\,,$$

où $\lambda = k/h$, on montre que le facteur d'amplification $g(h\xi) = 1 - i\,c\lambda\,\sin(h\xi)$
et $|g|^2 = 1 + c^2\lambda^2\sin^2(h\xi) \geq 1$: on a un schéma instable.

• **Schéma «upwind» :**

On a vu que pour $c > 0$ il faut utiliser une différence finie rétrograde pour garantir la stabilité et pour $c < 0$ il faut utiliser une différence finie progressive. En intégrant le test sur le signe de c, on obtient un schéma qui utilise v_{m-1}^n ou v_{m+1}^n, suivant la direction d'advection, d'où le nom du schéma. On a

$$\frac{v_m^{n+1} - v_m^n}{k} + \left(\frac{c - |c|}{2}\right)\,\frac{v_{m+1}^n - v_m^n}{h} + \left(\frac{c + |c|}{2}\right)\,\frac{v_m^n - v_{m-1}^n}{h} = 0\,;$$

ce qui s'écrit encore

$$\frac{v_m^{n+1} - v_m^n}{k} + c\,\frac{v_{m+1}^n - v_{m-1}^n}{2h} - |c|\frac{h}{2}\,\frac{v_{m+1}^n - 2v_m^n + v_{m-1}^n}{h^2} = 0\,.$$

On obtient donc un schéma explicite centré en espace avec un terme correcteur provenant d'une discrétisation de $(|c|h/2)\,u_{xx}$.
Le schéma est consistant, d'ordre $(1,1)$ et $g(h\xi) = 1 - 2|c|\lambda\sin^2(h\xi/2) - i\,c\lambda\sin(h\xi)$.
On a $|g(h\xi)|^2 = 1 - 4|c|\lambda(1 - |c|\lambda)\sin^2(h\xi) \leq 1$ si et seulement si $|c|\lambda \leq 1$.
On constate que le terme de *dissipation* ou *viscosité numérique* $(|c|h/2)\,u_{xx}$ a rendu stable le schéma centré en espace, par contre on a perdu dans l'ordre du schéma.

• **Schéma de** Lax-Friedrichs **:**

Ce schéma se présente sous la forme suivante

$$\frac{v_m^{n+1} - \frac{1}{2}(v_{m+1}^n + v_{m-1}^n)}{k} + c\,\frac{v_{m+1}^n - v_{m-1}^n}{2h} = 0\,.$$

Pour φ régulière on montre que

$$L_{h,k}\varphi - L\varphi = \frac{k}{2}\,\varphi_{tt} - \frac{h^2}{2k}\,\varphi_{xx} + O(k^2 + h^4/k) + \frac{ch^2}{6}\,\varphi_{xxx} + O(h^4)\,.$$

Le schéma est consistant si $h^2/k \longrightarrow 0$ pour $h, k \to 0$.

Si $k = \lambda\, h$, avec $\lambda \in \mathbb{R}_+^*$ fixé, on a un schéma consistant d'ordre 1.
Le facteur d'amplification est $g(h\xi) = \cos(h\xi) + i\, c\lambda \sin(h\xi)$ et
$|g|^2 = \cos^2(h\xi) + c^2\lambda^2 \sin^2(h\xi) \leq 1$ si et seulement si $|c|\lambda \leq 1$.
On peut écrire ce schéma sous la forme

$$\frac{v_m^{n+1} - v_m^n}{k} + c\,\frac{v_{m+1}^n - v_{m-1}^n}{2h} - \frac{1}{2}\,\frac{v_{m+1}^n - 2v_m^n + v_{m-1}^n}{k} = 0$$

et on constate que le schéma de LAX-FRIEDRICHS correspond à la discrétisation par différences finies centrées de l'équation aux dérivées partielles

$$u_t + c\, u_x - \frac{h^2}{2k}\, u_{xx} = 0\,.$$

On observe encore le phénomène de *viscosité numérique* qui rend le schéma stable.

• **Schéma de LAX-WENDROFF :**

Ce schéma est basé sur l'utilisation du développement de TAYLOR et de l'équation aux dérivées partielles à discrétiser. Si u est une solution régulière de l'équation aux dérivées partielles on a :

$$u_m^{n+1} = u_m^n + k\,\frac{\partial u}{\partial t} + \frac{k^2}{2}\,\frac{\partial^2 u}{\partial t^2} + O(k^2)\,,$$

or $u_t = -cu_x$ et $u_{tt} = (-cu_x)_t = -cu_{xt} = c^2\, u_{xx}$, d'où

$$u_m^{n+1} = u_m^n - ck\,\frac{\partial u}{\partial x} + c^2\frac{k^2}{2}\,\frac{\partial^2 u}{\partial x^2} + O(k^2)\,.$$

En utilisant une discrétisation spatiale centrée pour approximer u_x et u_{xx}, on obtient le schéma discret :

$$v_m^{n+1} = v_m^n - ck\,\frac{v_{m+1}^n - v_{m-1}^n}{2h} + c^2\frac{k^2}{2}\,\frac{v_{m+1}^n - 2v_m^n + v_{m-1}^n}{h^2}\,.$$

Ce schéma est consistent et d'ordre $(2,1)$ car

$$L_{h,k}\varphi - L\varphi = \frac{k}{2}\,\varphi_{tt} - \frac{c^2 k}{2}\,\varphi_{xx} + \frac{c^2 k h^2}{4!}\,\varphi_{xxxx} + O(h^3 + k^2)\,.$$

En posant $\lambda = k/h$:

$$v_m^{n+1} = v_m^n - \frac{c\lambda}{2}(v_{m+1}^n - v_{m-1}^n) + \frac{c^2\lambda^2}{2}(v_{m+1}^n - 2v_m^n + v_{m-1}^n)\,,$$

on montre que le facteur d'amplification est $g(h\xi) = 1 - 2c^2\lambda^2 \sin^2(h\xi/2) - i\, c\lambda \sin(h\xi)$
et $|g|^2 = 1 - 4c^2\lambda^2(1 - c^2\lambda^2)\sin^4(h\xi/2)$.
Ce schéma est donc stable pour la condition (CFL) : $|c\lambda| \leq 1$.

• Schéma «leapfrog» (saute-mouton) :

Comme son nom l'indique, ce schéma est centré en espace et en temps :

$$\frac{v_m^{n+1} - v_m^{n-1}}{2k} + c\,\frac{v_{m+1}^n - v_{m-1}^n}{2h} = 0$$

ou

$$v_m^{n+1} = v_m^{n-1} - c\lambda\,(v_{m+1}^n - v_{m-1}^n)\,.$$

Ce schéma est à deux pas, il faut donc donner les conditions initiales en v_m^0 et v_m^1. En pratique on utilise un schéma à un pas pour initialiser un schéma à pas multiples.

Comme $\qquad L_{h,k}\varphi - L\varphi = \dfrac{k^2}{6}\,\varphi_{ttt} - \dfrac{ch^2}{6}\,\varphi_{xxx} + O(h^2+k^2)\,,$

le schéma est consistant et d'ordre $(2,2)$.

L'analyse de VON NEUMANN ne s'applique pas directement à ce schéma. En appliquant la transformé de FOURIER spatiale on a

$$\widehat{v^{n+1}}(\xi) + i2c\lambda\sin(h\xi)\,\widehat{v^n}(\xi) - \widehat{v^{n-1}}(\xi) = 0\,. \tag{4.11}$$

Cette récurrence d'ordre deux s'exprime grâce à un système en introduisant une variable supplémentaire :

$$\begin{pmatrix} \widehat{v^{n+1}}(\xi) \\ \widehat{v^n}(\xi) \end{pmatrix} = \begin{pmatrix} -i2c\lambda\sin(h\xi) & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \widehat{v^n}(\xi) \\ \widehat{v^{n-1}}(\xi) \end{pmatrix}$$

Si on pose $\widehat{V^n} = \begin{pmatrix} \widehat{v^n} \\ \widehat{v^{n-1}} \end{pmatrix}$ on a $\widehat{V^{n+1}} = G(h\xi)\,\widehat{V^n}$, où G est la *matrice d'amplification*.

Par récurrence $\widehat{V^{n+1}} = G(h\xi)^n\,\widehat{V^1}$

et la condition de stabilité est donc de la forme $\|G^n\| \le C_T$.

Si le rayon spectral $\rho(G) = \max\{|g_i| \,/\, g_i$ valeur propre de $G\}$ est strictement plus petit que 1, alors on sait que G^n tend vers 0 et $\|G^n\|$ est bornée.

Mais si l'on veut généraliser le cas scalaire, on sait qu'il faut permettre $|g_i| \le 1 + Kk$. Or pour les systèmes ceci est une condition nécessaire de stabilité mais non suffisante, comme on le verra dans ce qui suit.

Nous allons étudier les valeurs propres de G. Remarquons d'abord que G est la *matrice compagne* de la relation de récurrence (4.11), son polynôme caractéristique correspond à l'équation en g, obtenue en remplaçant v_m^n par $g^n\,e^{imh\xi}$ dans le schéma discret :

$$g^2 + i2c\lambda\sin(h\xi)\,g - 1 = 0\,.$$

On a les racines $g_\pm(h\xi) = -ic\lambda\sin(h\xi) \pm \sqrt{1 - c^2\lambda^2\sin^2(h\xi)}$.

Cas $c\lambda < 1$: on a deux racines complexes conjuguées g_+ et g_- avec $|g_\pm| = 1$.
La matrice d'amplification $G(h\xi)$ est diagonalisable et la solution de (4.11) est

$$\widehat{v^n}(\xi) = \alpha(h\xi)\,g_+(h\xi)^n + \beta(h\xi)\,g_-(h\xi)^n\,,$$

où α et β dépendent des données initiales $\widehat{v^0}$ et $\widehat{v^1}$. On montre que

$$\widehat{v^n} = \frac{-g_-\widehat{v^0} + \widehat{v^1}}{g_+ - g_-}\,g_+ + \frac{g_+\widehat{v^0} - \widehat{v^1}}{g_+ - g_-}\,g_- = \frac{g_+g_-^n - g_+^n g_-}{g_+ - g_-}\,\widehat{v^0} + \frac{g_+^n - g_-^n}{g_+ - g_-}\,\widehat{v^1}\,.$$

On en déduit que

$$|\widehat{v^n}|^2 \leq \frac{2}{1-c^2\lambda^2}\left(|\widehat{v^0}|^2 + |\widehat{v^1}|^2\right)$$

et, en utilisant l'identité de PARSEVAL, on obtient finalement

$$\|v^n\|_{l^2(h\mathbb{Z})} \leq \sqrt{\frac{2}{1-c^2\lambda^2}}\left(\|v^0\|_{l^2(h\mathbb{Z})} + \|v^1\|_{l^2(h\mathbb{Z})}\right) .$$

Le schéma est donc stable : à tout instant $t_n \in [0,T]$, la norme de v^n est bornée par celle des conditions initiales.

Cas $|c\lambda| > 1$: si $h\xi = -\pi/2$ on a deux racines distinctes $g_\pm(-\pi/2)$.
Or $|g_+(-\pi/2)|^2 = 1 + 2c^2\lambda^2 > 1$: la matrice $G(-\pi/2)$ est diagonalisable mais $G^n(-\pi/2)$ n'est pas borné et le schéma n'est pas stable.

Cas $|c\lambda| = 1$: on a des racines doubles pour $h\xi = \pm\pi/2$ et la solution de (4.11) s'écrit

$$\widehat{v^n}(\xi) = \alpha(h\xi)\, g_+(h\xi)^n + \beta(h\xi)\, n g_+(h\xi)^{n-1} ,$$

où α et β dépendent de $\widehat{v^0}$ et $\widehat{v^1}$. On a

$$\widehat{v^n}(\xi) = g_+(h\xi)^n(1-n)\,\widehat{v^0}(\xi) + n g_+(h\xi)^{n-1}\,\widehat{v^1}(\xi) .$$

Or, pour $h\xi = \pm\pi/2$, on trouve $g_+(h\xi) = \pm i$ et, si l'on suppose que $\widehat{v^1}(\xi) = g_0(h\xi)\,\widehat{v^0}(\xi)$, on a

$$\widehat{v^n} = (\pm i)^{n-1}\,(n g_0 \pm i(1-n))\,\widehat{v^0} .$$

Pour $\xi = \pm\frac{\pi}{2h}$ on a donc $|\widehat{v^n}(\xi)| \sim n C_\pm |\widehat{v^0}(\xi)|$ pour n grand, on peut donc construire des solutions pour lesquelles $\|v^n\|_{l^2(h\mathbb{Z})}$ croît de façon linéaire avec n.

Dans ce cas la matrice d'amplification $G(\pm\pi/2)$ a une valeur propre double de module 1 et elle peut se mettre sous forme de JORDAN mais le schéma est instable d'après la théorie. On peut remarquer que cette instabilité est très faible et rarement nuisible en pratique.

Finalement, le schéma leapfrog est stable si et seulement si $|c\lambda| < 1$.

Note : On remarque que les valeurs de v_m^{n+1} et v_{m+1}^{n+1} sont indépendantes, on pourra voir apparaître deux solutions indépendantes (en échiquier), ceci peut poser des problèmes en pratique même si $|c\lambda| < 1$ (en particulier pour des problèmes non linéaires).

4.5 Équations paraboliques

Les définitions de convergence, consistance, stabilité et d'ordre d'un schéma s'appliquent aussi aux équations paraboliques. Notre équation type sera

$$u_t = d\, u_{xx}\,, \qquad \text{avec } d \in \mathbb{R}_+^*\,.$$

• Schéma explicite centré en espace :

On a

$$\frac{v_m^{n+1} - v_m^n}{k} - d\,\frac{v_{m-1}^n - 2v_m^n + v_{m+1}^n}{h^2} = 0\,,$$

et $L_{h,k}\varphi - L\varphi = \dfrac{k}{2}\,\varphi_{tt} + O(k^2) - \dfrac{dh^2}{12}\,\varphi_{xxxx} + O(h^4)\,,$
le schéma est consistant et d'ordre $(2,1)$. En posant $\mu = k/h^2$:

$$v_m^{n+1} = d\mu\,(v_{m-1}^n + v_{m+1}^n) + (1 - 2d\mu)\,v_m^n$$

et le facteur d'amplification $g(h\xi) = 1 - 4d\mu\sin^2(h\xi/2)$. Donc $|g| \le 1$ si et seulement si $0 \le 4d\mu\sin^2(h\xi/2) \le 2$ pour tout ξ. La condition de stabilité est donc

$$d\mu \le \frac{1}{2}\,.$$

On constate que la condition de stabilité entraîne que $k/h \longrightarrow 0$ quand $h, k \to 0$. Ceci exprime le fait que la solution discrète v doit avoir un domaine de dépendance proche de celui de la solution u de l'équation aux dérivées partielles qui est ici l'axe $\{t = 0\}$.

L'équation de la chaleur modélisant une diffusion il est intéressant d'avoir un schéma dissipatif, *i.e.* $|g| < 1$ on choisira donc $\mu < 1/(2d)$.

Notons finalement que pour $\mu \in]0, \frac{1}{2d}[$ fixé et $k = \mu h^2$ le schéma est d'ordre 2.

• Schéma implicite centré en espace :

En écrivant un schéma rétrograde en temps et centré en espace on a :

$$\frac{v_m^{n+1} - v_m^n}{k} - d\,\frac{v_{m-1}^{n+1} - 2v_m^{n+1} + v_{m+1}^{n+1}}{h^2} = 0\,,$$

ou,

$$(1 + 2d\mu)\,v_m^{n+1} - d\mu\,(v_{m-1}^{n+1} + v_{m+1}^{n+1}) = v_m^n\,.$$

À chaque étape il faut résoudre un système tridiagonal pour obtenir v^{n+1} à partir de v^n.
Ce schéma est consistant, d'ordre $(2,1)$ et *inconditionnellement stable* car

$$g(h\xi) = \frac{1}{1 + 4d\mu\sin^2(h\xi/2)}\,.$$

• **Schéma de** CRANK-NICOLSON

Ce schéma est basée sur les deux précédents, on évalue le terme de diffusion en faisant la moyenne de l'écriture explicite et implicite :

$$\frac{v_m^{n+1}-v_m^n}{k}-\frac{d}{2}\left(\frac{v_{m-1}^n-2v_m^n+v_{m+1}^n}{h^2}+\frac{v_{m-1}^{n+1}-2v_m^{n+1}+v_{m+1}^{n+1}}{h^2}\right)=0\,.$$

Ce schéma implicite est consistant, d'ordre $(2,2)$ et on a

$$d\mu v_{m-1}^{n+1}-2(1+d\mu)v_m^{n+1}+d\mu v_{m+1}^{n+1}=-d\mu v_{m-1}^n-2(1-d\mu)v_m^n-d\mu v_{m+1}^n\,.$$

Le facteur d'amplification est $g(h\xi)=\dfrac{1-2d\mu\sin^2(h\xi/2)}{1+2d\mu\sin^2(h\xi/2)}=1-\dfrac{4d\mu\sin^2(h\xi/2)}{1+2d\mu\sin^2(h\xi/2)}$,
donc $|g|\leq 1$ et le schéma est inconditionnellement stable.

• **Schéma de** DU FORT-FRANKEL

Le schéma leapfrog est instable pour tout μ, or une modification de la discrétisation spatiale donne le schéma stable suivant

$$\frac{v_m^{n+1}-v_m^{n+1}}{2k}-d\,\frac{v_{m-1}^n-(v_m^{n+1}+v_m^{n-1})+v_{m+1}^n}{h^2}=0\,,$$

avec $L_{h,k}\varphi-L\varphi=-dk^2h^{-2}\,\varphi_{tt}+O(h^2)+O(k^2)\,.$
Ce schéma est donc consistant seulement si $k/h\longrightarrow 0$ quand $h,k\to 0$. En notant $\mu=k/h^2$ on a

$$(1+2d\mu)\,v_m^{n+1}=2d\mu\,(v_{m+1}^n+v_{m-1}^n)+(1-2d\mu)\,v_m^{n-1}\,.$$

L'étude de la stabilité nous amène à considérer le polynôme caractéristique de la matrice d'amplification G :

$$(1+2d\mu)\,g^2-4d\mu\cos(h\xi)\,g-(1-2d\mu)=0\,.$$

Dont les racines s'écrivent : $g_\pm(h\xi)=\dfrac{2d\mu\cos(h\xi)\pm\sqrt{1-4d^2\mu^2\sin^2(h\xi)}}{1+2d\mu}$.

Si $4d^2\mu^2\sin^2(h\xi)<1$, on a deux racines réelles distinctes et

$$|g_\pm(h\xi)|\leq\frac{2d\mu|\cos(h\xi)|+\sqrt{1-4d^2\mu^2\sin^2(h\xi)}}{1+2d\mu}<1\,.$$

Si $4d^2\mu^2\sin^2(h\xi)>1$, on a deux racines complexes conjuguées et

$$|g_\pm(h\xi)|^2=\frac{1}{(1+2d\mu)^2}\left((2d\mu\cos(h\xi))^2+4d^2\mu^2\sin^2(h\xi)-1\right)=\frac{4d^2\mu^2-1}{(1+2d\mu)^2}<1\,.$$

Si $4d^2\mu^2\sin^2(h\xi)=1$, on a une racine réelle double et $\quad|g_+(h\xi)|=\dfrac{2d\mu|\cos(h\xi)|}{1+2d\mu}<1\,.$

Dans tous les cas le rayon spectral est strictement plus petit que 1 et $G(h\xi)^n$ tend vers zéro. On a donc un schéma explicite qui est inconditionnellement stable mais conditionnellement consistant.

4.6 Applications

4.6.1 Équation d'advection-diffusion-réaction

On s'intéresse à la résolution numérique de l'équation aux dérivées partielles suivante :

$$\frac{\partial u}{\partial t}(x,t) + c\,\frac{\partial u}{\partial x}(x,t) = d\,\frac{\partial^2 u}{\partial x^2}(x,t) + r\,u(x,t)\,, \qquad \text{pour } (x,t) \in \mathbb{R} \times \mathbb{R}_+^* \tag{4.12}$$

où $c \in \mathbb{R}^*$, $r \in \mathbb{R}$, $d \in \mathbb{R}_+^*$ et $u(x,0) = \Phi(x)$.
On va utiliser un schéma progressif en temps et centré en espace :

$$\frac{v_m^{n+1} - v_m^n}{k} + c\,\frac{v_{m+1}^n - v_{m-1}^n}{2h} = d\,\frac{v_{m-1}^n - 2v_m^n + v_{m+1}^n}{h^2} + r\,v_m^n \tag{4.13}$$

Ce schéma est consistant et d'ordre $(2,1)$, si on note $\mu = k/h^2$ alors $g(\xi,h,k) \leq 1 + kK$ si et seulement si $d\mu \leq \frac{1}{2}$ (la constante K ne dépend que de μ, c et r).
Pour $\mu \in]0, \frac{1}{2d}[$ fixé et $k = \mu h^2$ on a un schéma d'ordre 2.
En notant $\alpha = \dfrac{ch}{2d}$, le schéma (4.13) s'écrit aussi

$$v_m^{n+1} = (1 - 2d\mu + rk)v_m^n + d\mu(1-\alpha)v_{m+1}^n + d\mu(1+\alpha)v_{m-1}^n\,.$$

Pour $\mu \in]0, \frac{1}{2d}[$ fixé, $\alpha \leq 1$ et $1 - 2d\mu + r\mu h^2 \geq 0$, on a

$$|v_m^{n+1}| \leq (1 - 2d\mu + rk)|v_m^n| + d\mu(1-\alpha)|v_{m+1}^n| + d\mu(1+\alpha)|v_{m-1}^n| \leq (1 + r^+k)\sup_m |v_m^n|\,,$$

où $r^+ = \max(0,r)$. Par récurrence on obtient le *principe du maximum discret* suivant :

$$\forall t_n \in [0,T] \;:\; \sup_m |v_m^n| \leq (1+kr^+)^n \sup|v_m^0| \leq e^{Tr^+} \sup_m |v_m^0|\,.$$

Pour obtenir ce résultat, on a imposé des contraintes sur la discrétisation spatiale h :

$$h \leq \frac{2d}{c} \qquad \text{et, si } r < 0,\ k = \mu h^2 \leq \frac{2d\mu - 1}{r}\,.$$

Si ces inégalités ne sont pas vérifiées, la solution discrète n'a pas le même comportement que la solution de l'équation aux dérivées partielles parabolique (4.12) et on pourra observer des oscillations dans la solution. Par contre, si pour satisfaire ces contraintes il faut prendre h trop petit, il se peut que le schéma devient inefficace.

Remarque : En modélisation d'un flux de chaleur, la quantité $Pe = c/d\,[1/m]$ est appelée *nombre de* Peclet, en mécanique des fluides $Re = c/d$ est le *nombre de* Reynolds. Ce nombre contrôle l'intensité relative de l'advection par rapport à la diffusion : si Pe est grand, c'est l'advection qui domine, si Pe est petit c'est la diffusion. On peut interpréter c/d comme étant le rapport du temps de diffusion, $t_{diff} = x^2/d$, et du temps d'advection, $t_{adv} = x/c$.

Ci-dessous on propose un code `Scilab` [1] pour implémenter le schéma (4.13). Noter que l'on s'est intéressé principalement aux calculs et non pas à l'utilisation du code (p.ex. entrée interactive des paramètres).

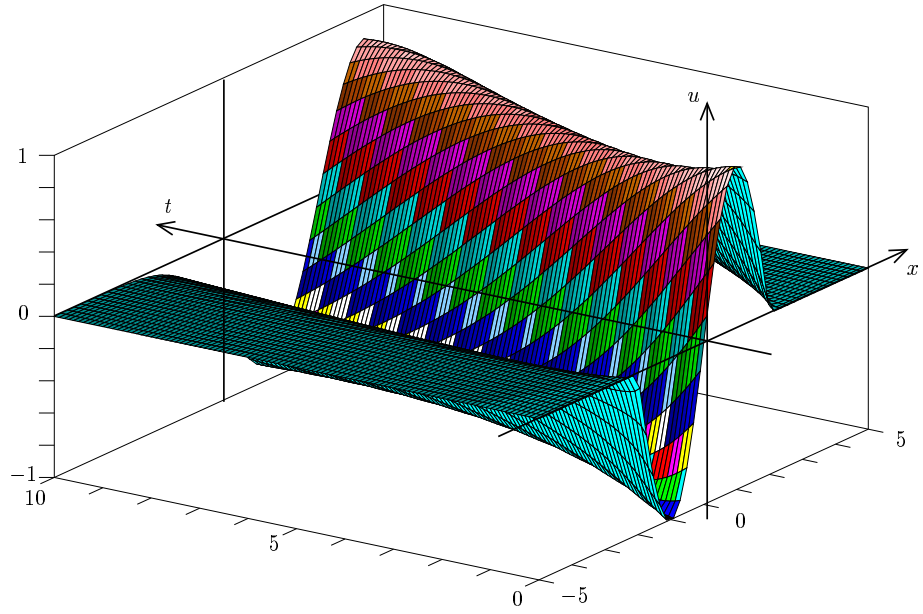
[1] © INRIA, logiciel disponible à `http://www-rocq.inria.fr/scilab/`

```
1   // Paramètres de l'edp
2   c=0.30;
3   d=0.10;
4   r=0.13;
5
6   // Paramètres du schéma
7   k=0.1;                        // pas de temps
8   T=10;                         // instant final
9   N=ceil(T/k)+1;                // nb. de points dans [0,T]
10  t=linspace(0,T,N);            // intervalle discret de temps
11
12  h=0.2;                        // pas de discrétisation spatiale
13  xg=-5; xd=10;                 // limites spatiales
14  M=ceil((xd-xg)./h)+1;         // nb. de points dans [xg,xd]
15  x=linspace(xg,xd,M);          // intervalle discret d'espace
16
17  v=zeros(N,M);                 // v(n,:) solution à l'instant n
18
19  // Afficher les paramètres du schéma
20  mu=k/h^2;
21  alpha=c.*h./(2.*d);
22  mprintf('c=%f\td=%f\tr=%f\n',c,d,r)
23  mprintf('h=%f\tk=%f\nmu=%f\talpha=%f \n',h,k,mu,alpha)
24  mprintf('mu*d=%f\t2/Pe=%f\t',mu.*d,(2.*d)./c)
25  if (r<0) then
26    mprintf('(2d*mu-1)/r=%f \n',(2.*d.*mu- 1)./r)
27  else
28    mprintf('\n')
29  end
30
31  // Condition initiale phi() définie sur [xg0,xd0]
32  xg0=-2;xd0=2;
33  Ind_xg0_xd0=bool2s((xg0 <= x)&(x <= xd0)); // indicatrice de  [xg0,xd0]
34  deff('[y]=phi(z)','y=sin(%pi *z/2)');
35  v(1,:)=Ind_xg0_xd0 .* phi(x);              // définition de la c.i.
36
37  // Itérations sur le vecteur v(n,:)
38  c1=1-2.*d.*mu+r.*k; c2=d.*mu.*(1-alpha); c3=d.*mu.*(1+alpha);
39  for n=2:N
40    v(n,:)=c1*v(n-1,:) + c2*[v(n-1,2:M) 0] + c3*[0 v(n-1,1:M-1)];
41  end
42
43  // Afficher la solution v dans [xg,xd]x[0,T]x[min(v),max(v)]
44  plot3d1(x,t,v',-111,86,"x@t@u",[2 2 4])
```

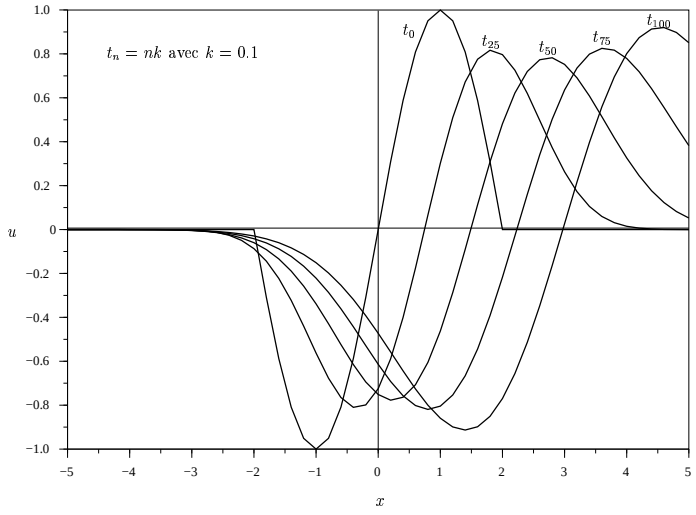
On obtient l'affichage des paramètres :

```
c=0.300000      d=0.100000      r=0.130000
h=0.200000      k=0.100000
mu=2.500000     alpha=0.300000
mu*d=0.250000   2/Pe=0.666667
```

et, après calculs, la figure suivante (on s'est restreint à $x \in [-5, +5]$) :



On constate le transport et la diffusion, à laquelle s'oppose la réaction, de la concentration initiale Φ. Souvent on trace la solutions dans le plan xu à des instants t_n différents :



Finalement, on s'intéresse à la précision de v, solution de (4.13) sur la grille discrète, par rapport à u, solution de (4.12) sur $\mathbb{R} \times [0, T]$. On a à l'instant $t_{100} = 10$, avec $h = 0,2$:

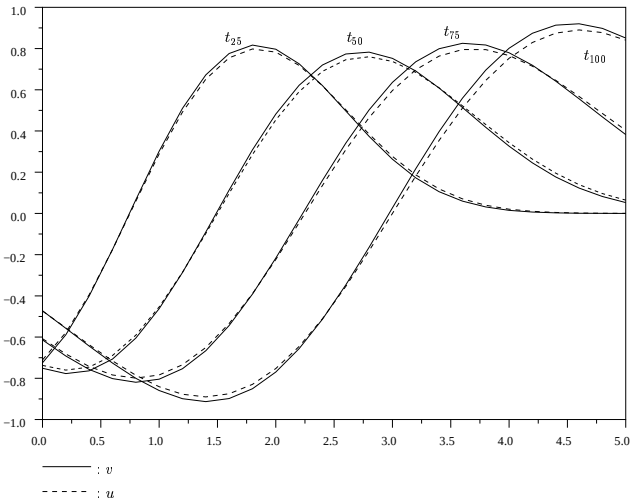
$$err_\infty(t_{100}) = \sup_{x_m \in [-5,10]} \left| u(x_m, 10) - v_m^{100} \right| = 0,0497294$$

et

$$err_2(t_{100}) = \left(h \sum_{x_m \in [-5,10]} \left(u(x_m, 10) - v_m^{100} \right)^2 \right)^{1/2} = 0,0722882\,.$$

Sur la figure suivante on a représenté la solution continue u et la solution discrète v, on constate des différences de l'ordre de $5 \cdot 10^{-2}$, cf. err_∞. L'erreur relative, $|u - v|/|v|$, peut atteindre 2.6 aux points où $u \sim 0$.

Pour conclure on peut dire que le schéma reproduit assez bien le comportement de la solution de l'équation aux dérivées partielles, par contre la discrétisation grossière et des phénomènes de bord perturbent rapidement le résultat numérique et rendent le résultat quantitativement inutilisable.
Pour améliorer le résultat il faut diminuer h (donc k) et augmenter le domaine spatial discret, on augmente la mémoire et le temps de calcul nécessaires.



4.6.2 Équation de la chaleur

On se propose de résoudre numériquement l'équation de la chaleur dans un ouvert de $\mathbb{R}^2$:

$$\begin{cases} \dfrac{\partial u}{\partial t}(x,y,t) = d\left(\dfrac{\partial^2 u}{\partial x^2}(x,y,t) + \dfrac{\partial^2 u}{\partial y^2}(x,y,t)\right) & \text{pour } (x,y,t) \in]0,a[^2 \times]0,T] \\ u(x,y,0) = \Phi(x,y) & \text{pour } (x,y) \in]0,a[^2 \\ u(x,y,t) = 0 & \text{pour } (x,y,t) \in \partial(]0,a[^2) \times [0,T]\,, \end{cases}$$

où $d \in \mathbb{R}_+^*$, $T > 0$ et la condition initiale Φ est donnée.

Comme dans le cas des fonctions à une dimension spatiale, on utilise la formule de Taylor pour approximer les dérivées partielles de u. Les valeurs de v sur la grille discrète sont notées $v_{m,p}^n$, où $n \in \mathbb{N}$ et $m, p \in \mathbb{Z}$. Pour simplifier on suppose que $(x_m, y_p) = (mh, ph)$, c'est-à-dire que l'on a une grille spatiale carrée.
Le schéma de Crank-Nicolson s'écrit

$$\frac{v_{m,p}^{n+1} - v_{m,p}^n}{k} = \frac{d}{2}\left(\frac{v_{m+1,p}^n - 2v_{m,p}^n + v_{m-1,p}^n}{h^2} + \frac{v_{m,p+1}^n - 2v_{m,p}^n + v_{m,p-1}^n}{h^2} + \frac{v_{m+1,p}^{n+1} - 2v_{m,p}^{n+1} + v_{m-1,p}^{n+1}}{h^2} + \frac{v_{m,p+1}^{n+1} - 2v_{m,p}^{n+1} + v_{m,p-1}^{n+1}}{h^2} \right)$$

et, en posant $\mu = k/h^2$, on a :

$$2(1+2d\mu)\, v_{m,p}^{n+1} - d\mu\,(v_{m+1,p}^{n+1} + v_{m-1,p}^{n+1} + v_{m,p+1}^{n+1} + v_{m,p-1}^{n+1}) = 2(1-2d\mu)\, v_{m,p}^n + d\mu\,(v_{m+1,p}^n + v_{m-1,p}^n + v_{m,p+1}^n + v_{m,p-1}^n)\,.$$

Souvent on utilise les masques pour exprimer la relation entre v^n et v^{n+1} :

$$\begin{bmatrix} 0 & -d\mu & 0 \\ -d\mu & 2(1+2d\mu) & -d\mu \\ 0 & -d\mu & 0 \end{bmatrix} v^{n+1} = \begin{bmatrix} 0 & d\mu & 0 \\ d\mu & 2(1-2d\mu) & d\mu \\ 0 & d\mu & 0 \end{bmatrix} v^n . \qquad (4.14)$$

Cette relation est valable pour tous les points intérieurs au domaine spatial, *i.e.* les points $(x_m, y_p) \in]0, a[^2$. Les valeurs sur le bord étant imposées par le problème, il faut déterminer $v_{m,p}^{n+1}$ en tout point intérieur, grâce à la relation ci-dessus. Pour cela il faut résoudre un système linéaire que l'on va construire maintenant.

Supposons que $[0, a]$ est discrétisé en $M+2$ points, les points intérieurs sont alors dans l'ensemble $\{(x_m, y_p) \,/\, 1 \le m, p \le M\}$, car $x_0 = y_0 = 0$ et $x_{M+1} = y_{M+1} = a$.
En utilisant l'ordre lexicographique, $i = m+(p-1)M$, pour ordonner les points intérieurs, on peut écrire la relation (4.14) sous forme matricielle :

$$A \cdot V^{n+1} = B^n , \qquad (4.15)$$

où $B^n \in \mathbb{R}^{M^2}$ et, pour $i = 1, \ldots, M^2$:

$$B_i^n = 2(1-2d\mu)\, v_{m,p}^n + d\mu\, (v_{m+1,p}^n + v_{m-1,p}^n + v_{m,p+1}^n + v_{m,p-1}^n).$$

On a $v_{m,p}^{n+1} = V_i^{n+1}$ et, pour les quatre voisins,

$$v_{m-1,p}^{n+1} = V_{i-1}^{n+1}, \quad v_{m+1,p}^{n+1} = V_{i+1}^{n+1}, \quad v_{m,p-1}^{n+1} = V_{i-M}^{n+1}, \quad v_{m,p+1}^{n+1} = V_{i+M}^{n+1} .$$

Tous les éléments de la i^{e} ligne, $1 \le i \le M^2$, de la matrice A sont nuls, sauf $A_{i,i} = \alpha$ et,

$$\begin{array}{llll} \text{si} \quad i \notin M\mathbb{N}+1 : & A_{i,i-1} = \beta \ ; & \text{si} \quad i > M : & A_{i,i-M} = \beta \ ; \\ \text{si} \quad i \notin M\mathbb{N}-1 : & A_{i,i+1} = \beta \ ; & \text{si} \quad i < M^2 - M : & A_{i,i+M} = \beta \, ; \end{array}$$

où $\alpha = 2(1+2d\mu)$ et $\beta = -d\mu$.
Pour $M = 4$, la matrice A s'écrit :

$$A = \left(\begin{array}{cccc:cccc:cccc:cccc} \alpha & \beta & & & \beta & & & & & & & & & & & \\ \beta & \alpha & \beta & & & \beta & & & & & & & & & & \\ & \beta & \alpha & \beta & & & \beta & & & & & & & & & \\ & & \beta & \alpha & & & & \beta & & & & & & & & \\ \hdashline \beta & & & & \alpha & \beta & & & \beta & & & & & & & \\ & \beta & & & \beta & \alpha & \beta & & & \beta & & & & & & \\ & & \beta & & & \beta & \alpha & \beta & & & \beta & & & & & \\ & & & \beta & & & \beta & \alpha & & & & \beta & & & & \\ \hdashline & & & & \beta & & & & \alpha & \beta & & & \beta & & & \\ & & & & & \beta & & & \beta & \alpha & \beta & & & \beta & & \\ & & & & & & \beta & & & \beta & \alpha & \beta & & & \beta & \\ & & & & & & & \beta & & & \beta & \alpha & & & & \beta \\ \hdashline & & & & & & & & \beta & & & & \alpha & \beta & & \\ & & & & & & & & & \beta & & & \beta & \alpha & \beta & \\ & & & & & & & & & & \beta & & & \beta & \alpha & \beta \\ & & & & & & & & & & & \beta & & & \beta & \alpha \end{array}\right)$$

La matrice A est tridiagonale par blocs, chacun des M^2 blocs est carré d'ordre M et A est une matrice $(M^2 \times M^2)$.

Pour chaque pas de temps t_n il faut résoudre (4.15), c'est-à-dire un système d'ordre M^2. On montre que la matrice A est définie positive et admet une décomposition de CHOLESKY : $A = L\,L^t$, où L est une matrice triangulaire inférieure. Cette décomposition simplifie la résolution du système (4.15). Afin d'économiser de la place mémoire, on utilise de plus une représentation adaptée à la matrice A qui est une matrice *creuse* (angl. *sparse*).
On propose le code `Scilab` suivant :

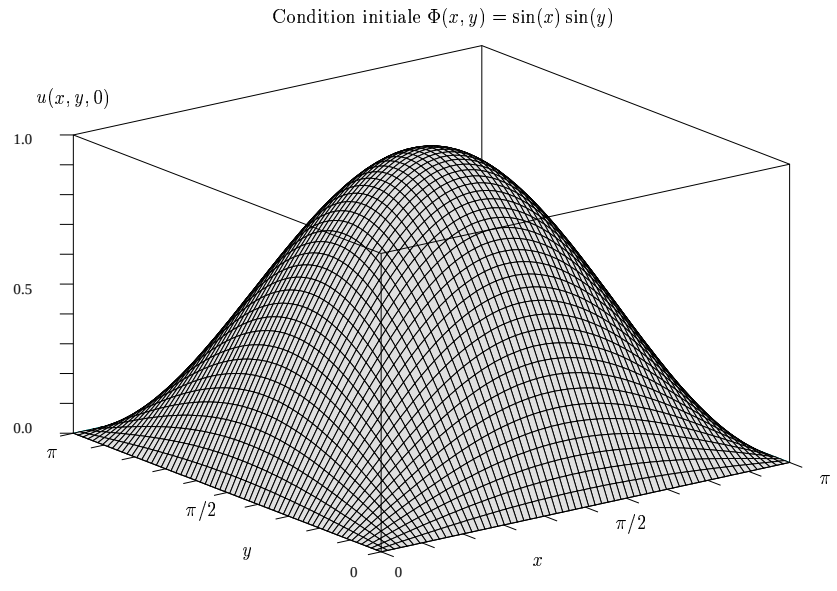
```
1   // Paramètres du problème
2   d=0.5;                    // coefficient de diffusion
3   a=%pi;                    // domaine spatial ]0,a[x]0,a[
4   T=1;                      // domaine temporel [0,T]
5
6   // Augmenter la mémoire de Scilab
7   stacksize(5000000);
8
9   // Paramètres du schéma
10  h=0.05;                   // pas de discrétisation spatiale
11  MM=ceil(a./h)+1;          // nb. de points dans [0,a]
12  M=MM-2;                   // nb. de points dans ]0,a[
13  xx=linspace(0,a,MM);      // intervalle discret d'espace de 0 à a
14  x=xx(1,2:MM-1);           // intervalle discret d'espace sans 0 et a
15
16  k=0.01;                   // pas de temps
17  N=ceil(T/k)+1;            // nb. de points dans [0,T]
18  t=linspace(0,T,N);        // intervalle discret de temps
19
20  mu=k./h^2;
21  M2=M^2;                   // taille du système linéaire
22  v=zeros(MM,MM,N);         // solution dans [0,a]^2x[0,T]
23
24  mprintf('d=%f\ta=%f\tT=%f\n',d,a,T)
25  mprintf('h=%f\tk=%f\n',h,k)
26  mprintf('mu=%f\tlambda=%f\n',mu,k/h)
27  mprintf('N=%i\tM=%i\tM^2=%i\n',N,M,M2)
28
29  // Condition initiale à évaluer sur les points intérieurs
30  deff('[z]=phi(x,y)','z=sin(x).*sin(y)');
31  v(2:MM-1,2:MM-1,1)=eval3d(phi,x,x);
32
33  // Construction des matrices A et B
34  // noter que l'on n'utilise que des matrices creuses
35  D1=sparse([1:M2-1;2:M2]',ones(M2-1,1),[M2 M2])..
36      -sparse([M:M:M2-M;1+M:M:M2-M+1]',ones(M-1,1),[M2 M2]);
37  DM=sparse([1:M2-M;1+M:M2]',ones(M2-M,1),[M2 M2]);
38
39  A=-(d*mu).*(D1+DM);               // triangle sup. de A
40  A=A+A';                           // A est symétrique
41  B=-A;
42  A=A+2.*(1+2*d*mu).*speye(M2,M2);  // on ajoute la diagonale
43  B=B+2.*(1-2*d*mu).*speye(M2,M2);
```

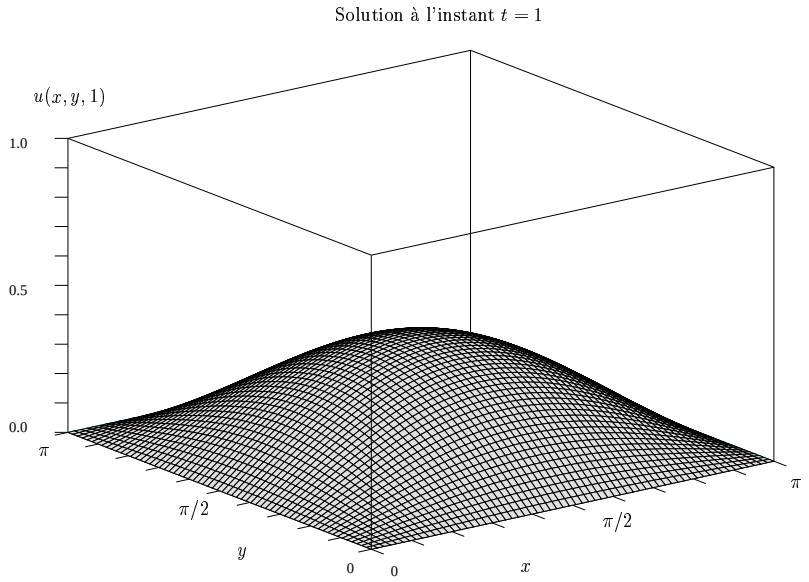
```
44  clear D1,DM;
45  mprintf('Stacksize=%i, used=%i\n',stacksize())
46  // Factorisation de A par la méthode de CHOLESKY
47  spchoA=chfact(A);
48  clear A;
49
50  // Affichage de la condition initiale
51  xbasc();
52  rect=[0,a,0,a,0,1];
53  xtitle(['Condition initiale']);
54  plot3d(xx,xx,v(:,:,1),233,81,"x@y@v(x,y,0)",[2,1,4],rect);
55
56  // Boucle sur le temps t_n : à chaque fois on résout le système
57  //    A V^(n+1) = B V^(n)  grâce à la décomposition de CHOLESKY
58  // on obtient le vecteur V=A^{-1} B V^(n)
59  // ensuite reconstruction de v(x,y,t(i)) et affichage
60  for i=2:N
61    V=chsolve(spchoA,B*matrix(v(2:MM-1,2:MM-1,i-1),M^2,1));
62    v(2:MM-1,2:MM-1,i)=matrix(V,M,M);
63    xclear();
64    xtitle(['Solution à l''instant t('+string(i)+')='+string(t(i))]);
65    plot3d(xx,xx,v(:,:,i),233,81,"x@y@u(x,y,t)",[2,1,4],rect)
66  end
67  clear B,V;
```

On obtient l'affichage des paramètres :

```
d=0.500000      a=3.141593      T=1.000000
h=0.050000      k=0.010000
mu=4.000000     lambda=0.200000
N=101   M=62    M^2=3844
Stacksize=5000000, used=1024430
```

et, les graphes suivants :





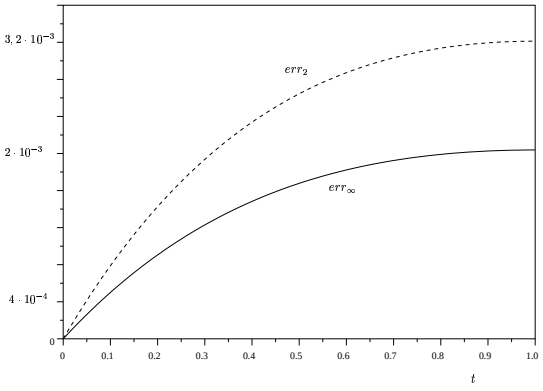
On peut de nouveau facilement calculer la solution u de l'équation aux dérivées partielles et comparer à chaque étape la solution discrète v à u. On utilise pour celà

$$err_\infty(t_n) = \sup_{(x_m,y_p)} |u(x_m, y_p, t_n) - v(x_m, y_p, t_n)|$$

et

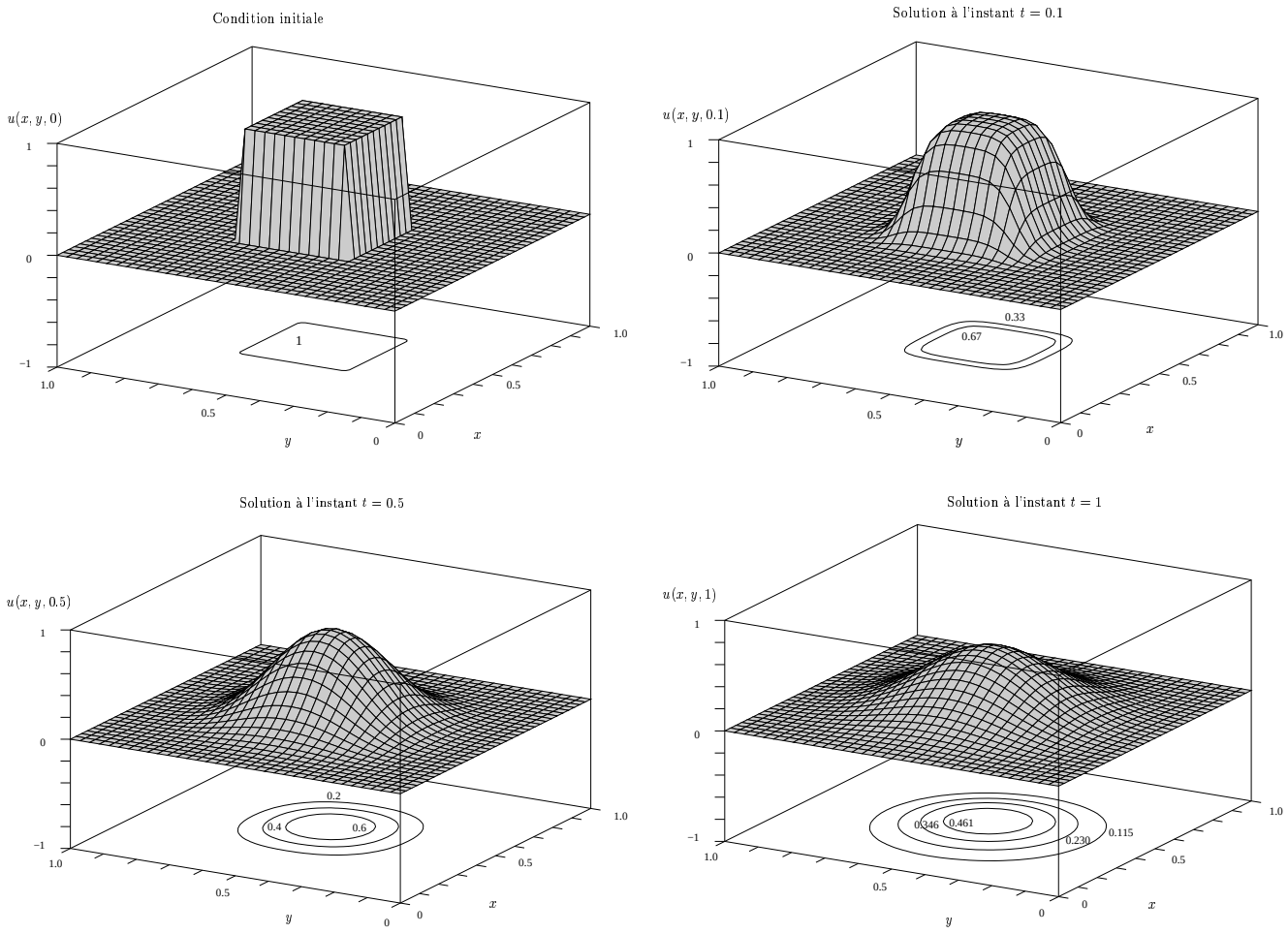
$$err_2(t_n) = h \left(\sum_{(x_m,y_p)} |u(x_m, y_p, t_n) - v(x_m, y_p, t_n)|^2 \right)^{1/2},$$

où $h = 0,05$. La figure suivante représente l'évolution de ces erreurs au cours du temps.

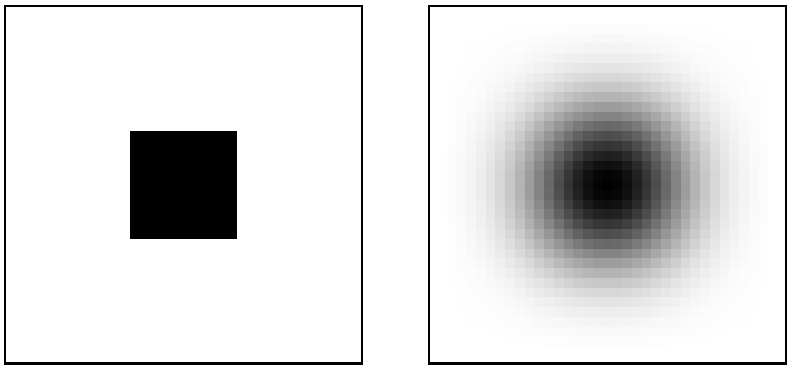


Pour voir l'effet de la diffusion sur une donnée initiale irrégulière, on se propose d'étudier sur $]0, 1[^2$ l'évolution de la fonction $\Phi(x, y) = \mathbb{I}_{[1/3,2/3]}(x, y)$. On obtient des résultats numériques satisfaisants pour $\lambda = k/h$ petit.
En adaptant le code précédent, on obtient les résultats suivants :

```
d=0.010000      a=1.000000      T=1.000000
h=0.030000      k=0.002000
mu=2.222222     lambda=0.066667
N=501   M=33    M^2=1089
Stacksize=5000000, used=672021
```



Sur ces graphiques on a représenté aussi quelques lignes de niveau de v, ceci afin de visualiser la «diffusion» du cube initial. Une autre façon de représenter v est d'associer des niveaux de gris aux valeurs que prend la fonction, ce qui donne les figures suivantes, où 0=blanc et 1=noir :



$v(x_m, y_p, 0)$ $v(x_m, y_p, 1)$

Remarque : Pour les images des pages 34 et suivantes, on a 255=blanc et 0=noir ; on y a utilisé des schémas moins coûteux. En effet, l'algorithme de cette section qui utilise la méthode de Crank-Nicolson, est très précis, mais il nécessite beaucoup de place mémoire et de temps de calcul. En dimension spatiale deux on utilise souvent d'autres méthodes.

4.6.3 Système de réaction-diffusion

Un exemple de système de réaction-diffusion qui engendre des rayures a été proposé par MEINHARDT [2] :

$$\begin{cases} \dfrac{\partial c_1}{\partial t} = d_c\,\Delta c_1 + \dfrac{c_1^2 s_2}{r} - \rho_c\, c_1 \\ \dfrac{\partial c_2}{\partial t} = d_c\,\Delta c_2 + \dfrac{c_2^2 s_1}{r} - \rho_c\, c_2 \\ \dfrac{\partial s_1}{\partial t} = d_s\,\Delta s_1 + \rho_s\,(c_1 - s_1) \\ \dfrac{\partial s_2}{\partial t} = d_s\,\Delta s_2 + \rho_s\,(c_2 - s_2) \\ \dfrac{\partial r}{\partial t} = c_1^2 s_2 + c_2^2 s_1 - \rho_r\, r \end{cases}$$

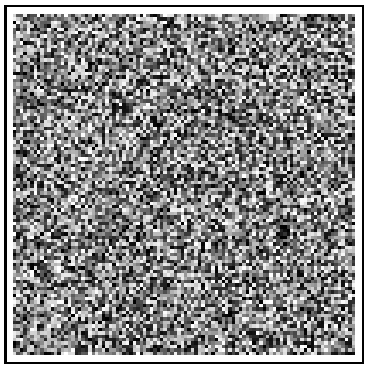
où les cinq «morphogènes» c_1, c_2, s_1, s_2 et r sont définies sur $[0, M]^2 \times \mathbb{R}_+$. Le comportement du système dépend des paramètres réels d_c, d_s, ρ_c, ρ_s et ρ_r et des conditions initiales.

Pour ce type de simulation on a besoin d'un nombre élevé d'itérations, N, et il devient nécessaire d'utiliser un programme compilé. Ainsi le système ci-dessus a été programmé en langage "C" en utilisant les librairies de `Megawave`[3]. On a utilisé un schéma explicite, progressif en temps et centré en espace, avec des conditions au bord périodiques. Les conditions initiales sont données par

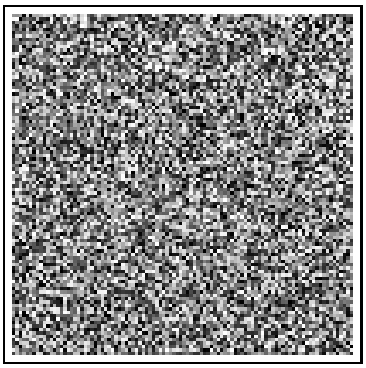
$$c_1(x,y,0) = \frac{\rho_r}{2\rho_c}\,(1+\epsilon_1(x,y))\,, \qquad c_2(x,y,0) = \frac{\rho_r}{2\rho_c}\,(1+\epsilon_2(x,y))\,,$$
$$s_1(x,y,0) = s_2(x,y,0) = \frac{\rho_r}{2\rho_c}\,, \qquad r(x,y,0) = \frac{\rho_r^2}{4\rho_c^3}\,,$$

pour $(x,y) \in [0,M]^2$ et où ϵ_1 et ϵ_2 est un bruit uniforme, de valeurs dans $[-\rho_\epsilon, \rho_\epsilon]$, avec $\rho_\epsilon > 0$ donné. La discrétisation spatiale est fixée à $h = 1$ dans les deux directions, le pas de temps k est obtenu en choisissant μ : $k = \mu h^2 = \mu$. Le choix des paramètres pour les premières simulations est indiqué dans la table suivante :

$\mu = 0,0100$	$M = 100$	$N = 30.000$
$d_c = 0,0500$	$d_s = 0,1000$	$\rho_\epsilon = 0,1000$
$\rho_c = 0,4000$	$\rho_s = 0,4000$	$\rho_r = 0,6000$



$c_1(x_m, y_p, 0)$
min=0,60000 / max=0,74996

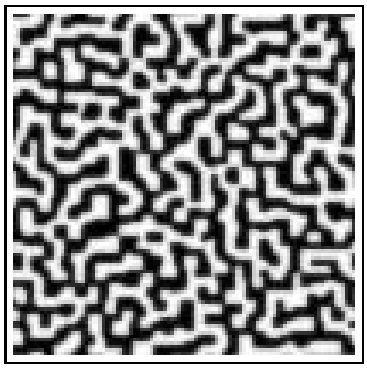


$c_2(x_m, y_p, 0)$
min=0,60002 / max=0,75000

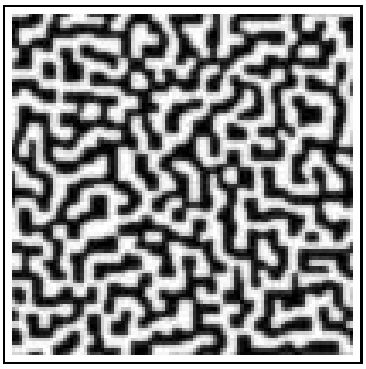
[2] *Models of Biological Pattern Formation*, Academic Press, 1982

[3] © logiciel disponible à `http://www.cmla.ens-cachan.fr/Cmla/Megawave`

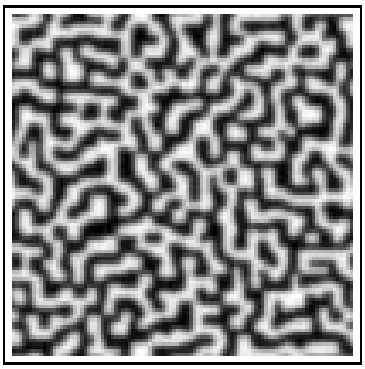
Note : Toutes les images ont été renormalisées dans $[0, 255]$.



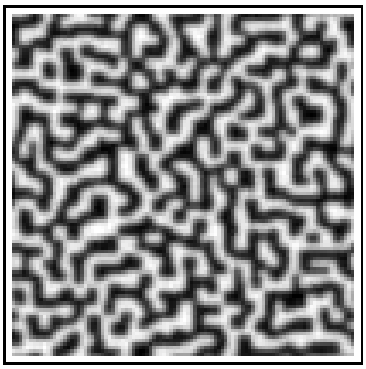
$c_1(x_m, y_p, kN)$
min=0,01389 / max=1,98951



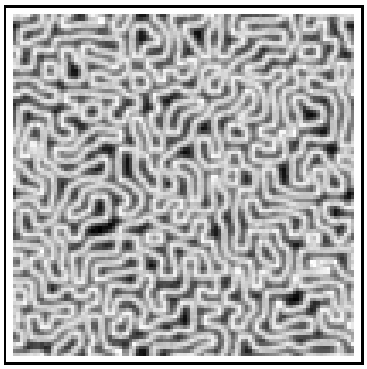
$c_2(x_m, y_p, kN)$
min=0,01164 / max=1,98889



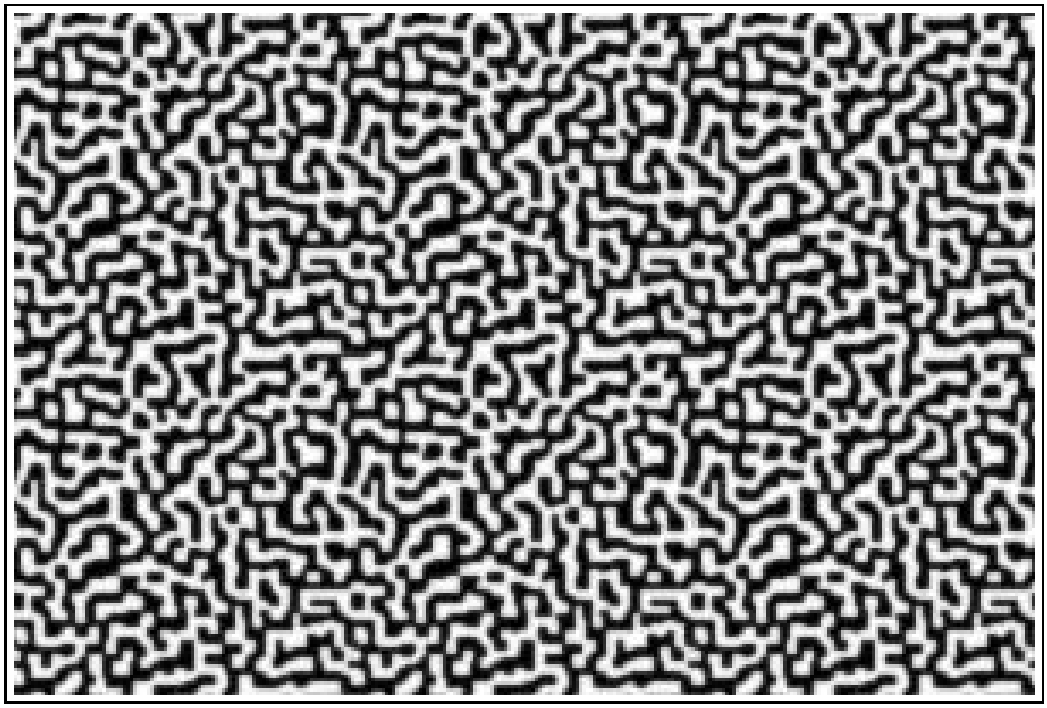
$s_1(x_m, y_p, kN)$
min=0,01889 / max=3,82714



$s_2(x_m, y_p, kN)$
min=0,01734 / max=3,81700



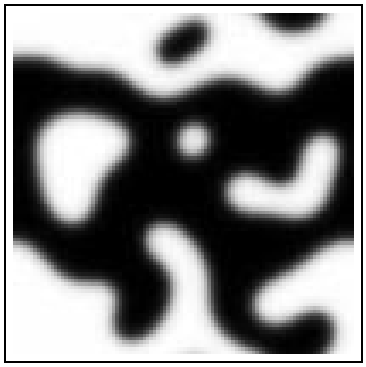
$r(x_m, y_p, kN)$
min=1,14387 / max=7,58884



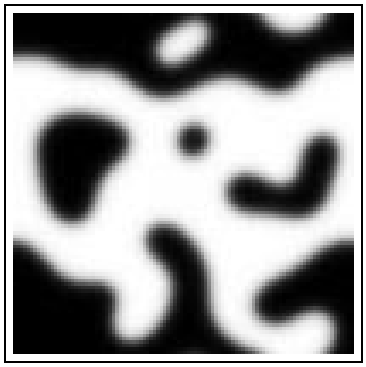
Utilisation de la périodicité de c_1 pour paver le plan.

Pour les paramètres du tableau suivant on obtient les zébrures représentées en dessous.

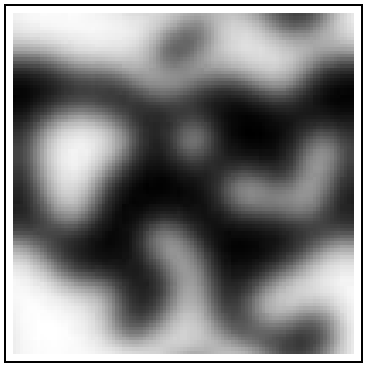
$\mu = 0,0050$	$M = 100$	$N = 50.000$
$d_c = 0,1500$	$d_s = 0,2000$	$\rho_\epsilon = 0,1000$
$\rho_c = 0,4000$	$\rho_s = 0,4000$	$\rho_r = 0,8000$



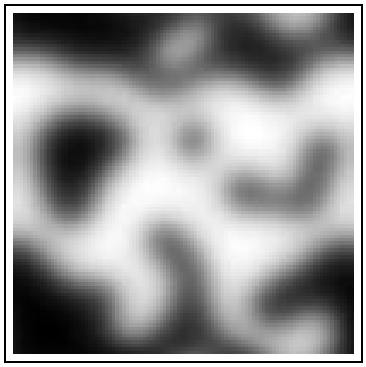
$c_1(x_m, y_p, kN)$



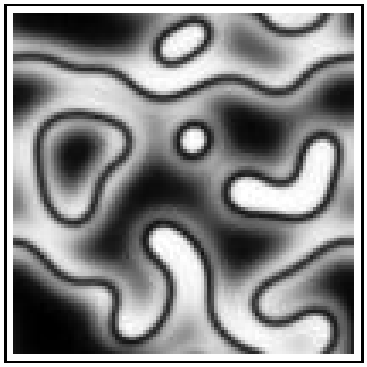
$c_2(x_m, y_p, kN)$



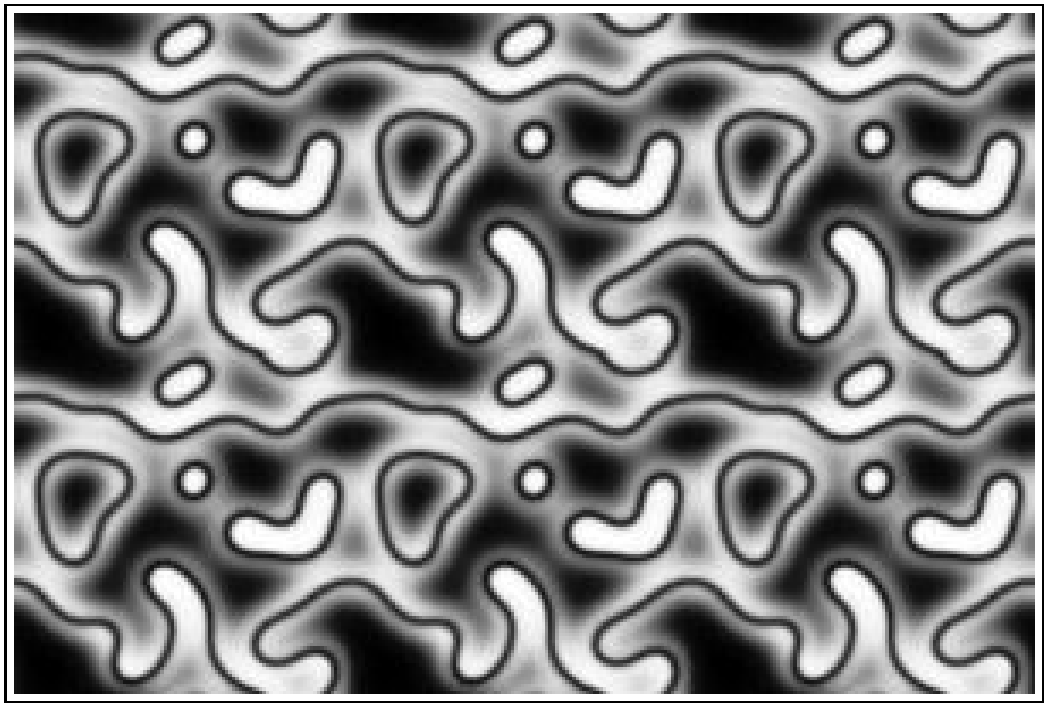
$s_1(x_m, y_p, kN)$



$s_2(x_m, y_p, kN)$



$r(x_m, y_p, kN)$



Utilisation de la périodicité de r pour paver le plan.

4.7 Équations elliptiques

Voir cours...

4.8 Équations non linéaires

Voir cours...

Annexe A

Rappels

A.1 Calcul différentiel dans $\mathbb{R}^n$

Un point $x = (x_1, \ldots, x_n)$ de $\mathbb{R}^n$ considéré comme vecteur sera noté $x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$
dont la transposé , $x^t = \begin{pmatrix} x_1 & \cdots & x_n \end{pmatrix}$, est une matrice $1 \times n$.
La norme euclidienne de x est noté $\|x\|_2 = \left(\sum_{i=1}^{n} x_i^2 \right)^{1/2}$ et le produit scalaire des vecteurs x et y s'écrit $(x, y)_2 = x.y = x^t y = \sum_{i=1}^{n} x_i y_i$.
Par $\|.\|$ on désigne une norme quelconque sur $\mathbb{R}^n$.

Soit $f \in \mathcal{C}^1(\mathbb{R}^n, \mathbb{R})$, pour $a = (a_1, \ldots, a_n) \in \mathbb{R}^n$, on note $\mathrm{D}_i f(a) = \dfrac{\partial f}{\partial x_i}(a) = f_{x_i}(a)$
la dérivée partielle de f en a par rapport à la variable x_i, $i = 1, \ldots, n$.
Pour $k \in \mathbb{N}$ on a $\mathrm{D}_i^k f(a) = \dfrac{\partial^k f}{\partial x_i^k}(a)$.

On note $\mathrm{D}f(a) = \Big(\mathrm{D}_1 f(a)\, \mathrm{D}_2 f(a)\, \ldots\, \mathrm{D}_n f(a) \Big)$ la matrice ligne $(1 \times n)$ associé à l'application différentielle $\mathrm{d}f_a \in \mathcal{L}(\mathbb{R}^n; \mathbb{R})$.

Le *vecteur gradient* est le vecteur colonne $(n \times 1)$ $\nabla f(a) = \mathrm{D}f(a)^t = \begin{pmatrix} \frac{\partial f}{\partial x_1}(a) \\ \vdots \\ \frac{\partial f}{\partial x_n}(a) \end{pmatrix}$.

Soit $f \in \mathcal{C}^2(\mathbb{R}^n, \mathbb{R})$ et $a = (a_1, \ldots, a_n) \in \mathbb{R}^n$, le *Hessien* de f au point a est la matrice symétrique

$$\nabla^2 f(a) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1}(a) & \frac{\partial^2 f}{\partial x_1 \partial x_2}(a) & \ldots & \frac{\partial^2 f}{\partial x_1 \partial x_n}(a) \\ \frac{\partial^2 f}{\partial x_1 \partial x_2}(a) & \frac{\partial^2 f}{\partial x_2 \partial x_2}(a) & \ldots & \frac{\partial^2 f}{\partial x_2 \partial x_n}(a) \\ \vdots & \vdots & & \vdots \\ \frac{\partial^2 f}{\partial x_1 \partial x_n}(a) & \frac{\partial^2 f}{\partial x_2 \partial x_n}(a) & \ldots & \frac{\partial^2 f}{\partial x_d \partial x_n}(a) \end{pmatrix}$$

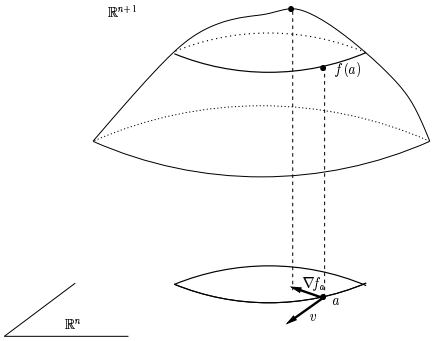
C'est la matrice associé à l'application bilinéaire symétrique $\mathrm{d}^2 f_a \in \mathcal{L}_2(\mathbb{R}^n; \mathbb{R})$.
Pour $(h, k) \in \mathbb{R}^n \times \mathbb{R}^n$ on a $\mathrm{d}^2 f_a(h, k) = (\nabla^2 f(a)h, k)_2 = h^t \nabla^2 f(a) k$.

Soit $f \in \mathcal{C}^1(\mathbb{R}^n, \mathbb{R})$, on définit la *dérivée directionnelle* de f au point $a \in \mathbb{R}^n$ et dans la direction de $v \in \mathbb{R}^n$, par

$$\frac{\mathrm{d}f}{\mathrm{d}v}(a) = \lim_{h \to 0} \frac{1}{h} \left(f(a + hv) - f(a) \right) .$$

On montre que

$$\frac{\mathrm{d}f}{\mathrm{d}v}(a) = \nabla f(a).v \qquad \Big\|$$



Soit Ω un ouvert de $\mathbb{R}^n$ et $f \in \mathcal{C}^2(\Omega, \mathbb{R})$. Pour $x \in \Omega$ et $h \in \mathbb{R}^n$ on suppose que le segment $[a, a + h] \subset \Omega$, alors la *formule de* Taylor-Young à l'ordre deux s'écrit

$$f(a + h) = f(a) + \nabla f(a).h + \frac{1}{2} h^t \nabla^2 f(a) h + o(\|h\|_2^2) .$$

A.2 Algèbre et analyse matricielle

On note $A = (a_{ij})$ une matrice et $a_{ij} = (A)_{ij}$ est l'élément de la i-ième ligne , j-ième colonne. Les majuscules seront en général réservées aux matrices et les minuscules aux éléments des matrices.

Définition A.2.1 (Matrices blocs)
– *Soient A et B des matrices de même dimensions avec*

$$A = \begin{pmatrix} A_{11} & \dots & A_{1q} \\ \vdots & & \vdots \\ A_{p1} & \dots & A_{pq} \end{pmatrix} \quad et \quad B = \begin{pmatrix} B_{11} & \dots & B_{1q} \\ \vdots & & \vdots \\ B_{p1} & \dots & B_{pq} \end{pmatrix}$$

où les blocs respectifs A_{ij} et B_{ij} ont même dimension. Alors on obtient l'addition par blocs de A et B :

$$A + B = \begin{pmatrix} A_{11} + B_{11} & \dots & A_{1q} + B_{1q} \\ \vdots & & \vdots \\ A_{p1} + B_{p1} & \dots & A_{pq} + B_{pq} \end{pmatrix} .$$

– *Soient A et B des matrices tel que AB est défini et*

$$A = \begin{pmatrix} A_{11} & \dots & A_{1q} \\ \vdots & & \vdots \\ A_{p1} & \dots & A_{pq} \end{pmatrix} \quad et \quad B = \begin{pmatrix} B_{11} & \dots & B_{1s} \\ \vdots & & \vdots \\ B_{r1} & \dots & B_{rs} \end{pmatrix} ,$$

si $q = r$ et si les produits $A_{ik}B_{kj}$ $(i = 1, \ldots, i\,;\, j = 1, \ldots, s\,;\, k = 1, \ldots, q)$ sont définis, alors la multiplication par blocs de A et B s'écrit :

$$AB = \begin{pmatrix} \sum_{k=1}^{q} A_{1k}B_{k1} & \ldots & \sum_{k=1}^{q} A_{1k}B_{ks} \\ \vdots & & \vdots \\ \sum_{k=1}^{q} A_{pk}B_{k1} & \ldots & \sum_{k=1}^{q} A_{pk}B_{ks} \end{pmatrix} .$$

– *Si A est une matrice tridiagonale par blocs $A = (A_{ij})$ $(1 \leq i, j \leq r)$, c.-à-d. les blocs A_{ij} ont comme dimension (p_i, q_j) $(1 \leq i, j \leq r)$ avec $A_{ij} = 0$ si $i \neq j$ et $p_i = q_i$ pour A_{ii}. On peut calculer le déterminant par blocs :*

$$dét(A) = \prod_{i=1}^{r} dét(A_{ii}) .$$

Définition A.2.2
– *Un matrice réelle carrée A est symétrique si $A^t = A$;*
elle est orthogonale si $A\,A^t = A^t\,A = I$.
– *On dit qu'une matrice A réelle symétrique est définie positive si*
pour tout $w \in \mathbb{R}^n$, $w \neq 0$ $w^t A w > 0$;
la matrice A est semi-définie positive si pour tout $w \in \mathbb{R}^n$ $w^t A w \geq 0$;.
– *Le rayon spectral d'une matrice carrée A d'ordre n est défini par*

$$\rho(A) = \max_{1 \leq i \leq n} \{|\lambda_i| \mid \lambda_i \text{valeur propre de } A\} .$$

Proposition A.2.1
Étant donné une matrice symétrique A, il existe une matrice orthogonale O telle que la matrice $O^t AO$ est diagonale.

Définition A.2.3
Soit $\|.\|$ une norme sur $\mathbb{R}^n$, on appelle norme matricielle subordonnée de la matrice carrée A d'ordre n

$$\|A\| = \max_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|}{\|x\|} = \max_{x \in \mathbb{R}^n, \|x\| = 1} \|Ax\| .$$

Proposition A.2.2
(a) Une norme matricielle subordonnée vérifie les propriétés suivantes :

$$\begin{aligned} &\|A\| = 0 \Leftrightarrow A = 0, && \text{pour tout } A; \\ &\|\alpha A\| = |\alpha|\,\|A\|, && \text{pour tout } A, \alpha \in \mathbb{R}; \\ &\|A + B\| \leq \|A\| + \|B\|, && \text{pour tout } A, B; \\ &\|AB\| \leq \|A\|\,\|B\|, && \text{pour tout } A, B. \end{aligned}$$

b) Soit A une matrice carrée réelle, pour les normes vectorielles usuelles on a les normes matricielles subordonnées :

$$\|A\|_\infty = \max_{1\le i\le n} \sum_{j=1}^{n} |a_{ij}| \quad = \text{«maximum sur les lignes»}$$

$$\|A\|_1 = \max_{1\le j\le n} \sum_{j=i}^{n} |a_{ij}| \quad = \text{«maximum sur les colonnes»}$$

$$\|A\|_2 = \sqrt{\rho(A^t A)} = \left\|A^t\right\|_2 \quad \text{où } \rho(A^t A) \text{ est le rayon spectral de de la matrice symétrique } A^t A$$

c) Si O et P sont des matrices orthogonales, alors $\|O\,A\,P\|_2 = \|A\|_2$.
Si $AA^t = A^t A$ (i.e. *normale), alors $\|A\|_2 = \rho(A)$*

Toutes les normes de matrices considérées dans ce cours vont vérifier A.2.2 (a) . Par contre il n'existe pas que des normes matricielles subordonnées, comme le montre le résultat de la

PROPOSITION A.2.3
La norme de FROBENIUS *définie par*

$$\|A\|_F = \left(\sum_{1\le i,j\le n} |a_{ij}|^2 \right)^{\frac{1}{2}} = (trace(A^t A))^{\frac{1}{2}} ,$$

vérifie les propriétés de la proposition A.2.2 (a) , de plus :
– *Si O et P sont des matrices orthogonales, alors $\|O\,A\,P\|_F = \|A\|_F$.*
– *On a l'estimation utile suivante*

$$\|A\|_2 \le \|A\|_F \le \sqrt{n}\,\|A\|_2 .$$