Compléments de cours pour l'épreuve de modélisation

Table des matières

1 Erreurs de calculs, condition d'un problème et d'une matrice

Pour les références, on pourra consulter [Dem91] (pour la partie erreurs de calcul) et [Cia] (pour la partie condition d'une matrice).

1.1 Précision et erreurs de calculs

Représentation en virgule flottante

Dans un ordinateur, un nombre réel x est représenté en général par son approximation en virgule flottante :

$$x \approx \pm m \cdot b^p,$$

où b est la base de numérotation, m la mantisse et p l'exposant. Les calculs internes sont généralement effectués en base $b = 2$, même si les résultats affichés sont finalement traduits en base 10. La représentation $\pm m \cdot b^p$ n'est pas unique : par exemple, $3,1416$ en base $b = 10$ peut s'écrire $0,31416 \cdot 10^1$ ou $314,16 \cdot 10^{-2}$; pour garantir l'unicité, il faut définir une convention sur la mantisse (par exemple, virgule après le premier chiffre non nul, ou avant celui-ci,...).

Le codage d'un nombre en virgule flottante est fixé par la norme IEEE 754, norme adoptée par la majorité des ordinateurs (cf. article "Virgule flottante" de Wikipédia"), soit sur 32 bits ("simple précision"), soit sur 64 bits ("double précision"). La répartition des bits est la suivante :

	Encodage	**S**igne	**E**xposant	**M**antisse	Valeur d'un nombre
Simple p.	32 bits	1 bit	8 bits	23bits	$(-1)^S \times \mathrm{M} \times 2^{E-127}$
Double p.	64 bits	1 bit	11bits	52bits	$(-1)^S \times \mathrm{M} \times 2^{E-1023}$

Le tableau ci-dessus indique les bits représentés. Pour un nombre *normalisé* (virgule après le premier chiffre non nul), le premier bit de la mantisse est toujours 1 (qu'on ne stocke pas), et $1 \leq M < 2$, de sorte que

$$m = M = 1, a_1 a_2 a_3 \dots a_N = 1 + \sum_{k=1}^{N} a_k 2^{-k},$$

où les chiffres $a_1, \dots, a_N$ prennent la valeur 0 ou 1, et où N est le nombre de bits dans la mantisse. Cette écriture implique que m représente tous les réels de $[1, 2)$ valant m au dernier chiffre près, i.e. à $\pm 0.5 \cdot 2^{-N}$ près. Autrement dit, m représente un réel avec la précision (absolue) $\Delta m = 2^{-N}$ (NB : pour cette raison, on appelle parfois "précision" le nombre de N de bits utilisé pour la mantisse).

En SCILAB, les calculs sont faits en double précision (64bits=8octets ou "bytes" en anglais) ; la précision est de $N = 52$ bits, et elle est stockée dans la constante prédéfinie %eps=2^{-52}. La valeur de l'exposant p varie entre $E_{max} - 1023 = (2^{11} - 1) - 1023 = 1024$ et $E_{min} - 1023 = -1023$.

Pour raisonner sur la précision des calculs, un concept important est celui de *précision relative*. Ainsi, la mantisse d'un nombre normalisé est connu avec la précision relative

$$\frac{\Delta m}{m} \leq \frac{2^{-N}}{1} = 2^{-N}.$$

Si x est un réel représentable en virgule flottante par une matrice qui est un nombre normalisé, la précision relative sur x correspond à celle de sa mantisse : $\Delta x / x = \Delta m / m$.

Erreur d'arrondi sur une somme

Ces erreurs d'arrondis provoquent de fait des erreurs de calcul dans les opérations élémentaires. Ainsi, supposons que les réels sont calculés avec 3 chiffres significatifs et arrondis à la décimale la plus proche. Si on additionne $x = 3,36$ et $y = 0,00245$ on trouve $x+y = 3.36245 \approx 3.36 = x$: on perd dans le calcul tous les chiffres significatifs de y. Additionnons maintenant x, y et $z = 0,00471$. Si on effectue les opérations dans le sens $(x + y) + z$, on trouve

$$x + y \approx 3.36 \text{ et } (x + y) + z \approx 3,3647 \approx 3.36;$$

si on effectue dans le sens opposé, $(z + y) + x$, on trouve

$$z + y = 0,00716 \text{ et } (z + y) + x \approx 3,36716 \approx 3.37.$$

On remarque que l'addition n'est plus associative. D'autre part, le résultat exact est $x+y+z = 3.36716$, donc la deuxième manière est meilleure parce que les petits termes s'additionnent. C'est un principe général : quand on veut sommer une suite de termes positifs, il vaut mieux le faire du plus petit au plus grand.

Phénomène de compensation

Une source importante d'erreur réside dans les phénomènes de compensation, quand on additionne des termes de signes opposés. Ainsi, si les réels sont calculés avec 6 chiffres significatifs, si on a $x = 1,00004$ et $y = 1,00001$, tous deux connus avec 6 chiffres significatifs, la différence $x - y = 0,00003$ est connue avec seulement 1 chiffre significatif. La précision relative était de 10^{-5} sur x et y et elle est de l'ordre de 1 sur la différence !

Calculs itératifs

Considérons la suite récurrente définie par

$$x_0 = 1/3 \quad \text{ et } x_{n+1} = 1 - 2x_n \text{ pour } n \geq 0.$$

En théorie, on devrait avoir $x_n = 1/3$ pour tout n. En pratique, à cause des erreurs d'arrondis, au bout d'un certain nombre d'itérations, la valeur x_n obtenue n'a plus rien à voir avec $1/3$. La précision étant de 10^{-16}, si l'erreur d'arrondi est multiplié par 2 à chaque étape, on a une erreur de l'ordre de l'unité pour n tel que $2^n > 10^{16}$, i.e. $n \approx 60$ environ. Cela est bien vérifié en pratique.

Considérons la suite récurrente à deux pas

$$\begin{cases} x_0 = 1,\ x_1 = (1 - \sqrt{5})/2, \\ x_{n+2} = x_{n+1} + x_n \qquad\qquad n \geq 0. \end{cases} \tag{1}$$

L'équation caractéristique est $r^2 - r - 1 = 0$, de racines $r_1 = (1 - \sqrt{5})/2 \in]-1, 0[$ et $r_2 = (1 + \sqrt{5})/2 > 1$. Toute suite satisfaisant la relation de récurrence ci-dessus est de la forme $Ar_1^n + Br_2^n$, où A et B sont imposés par les conditions initiales. En particulier, pour $x_0 = 1$ et $x_1 = r_1$ comme ci-dessus, on trouve $A = 1$ et $B = 0$ donc $x_n = r_2^n \ \forall n \in \mathbb{N}$, et $x_n \to 0$ quand $n \to \infty$.

Dans un calcul par ordinateur, x_1 sera remplacé par une valeur approchée $\hat{x_1}$, donc A et B sont remplacés par des valeurs approchées $\hat{A}$ et $\hat{B}$. Ainsi la suite calculée sera (en première approximation, car il faudrait tenir compte des erreurs de calcul à chaque étape) de la forme $\hat{x}_n = \hat{A}r_1^n + \hat{B}r_2^n$, et en général $\hat{x}_n \to signe(\hat{B}) \cdot \infty$ quand $n \to \infty$.

1.2 Condition d'un problème

On dit qu'**un problème mathématique est bien posé** au sens de Hadamard s'il admet une unique solution et si la solution dépend continûment des données.

En analyse numérique, on considérera toujours des problèmes bien posés au sens de Hadamard. En vue d'un traitement par ordinateur, un problème sera **discrétisé**, c'est-à-dire posé en dimension finie. En fin de compte, un problème bien posé et discrétisé peut se concevoir comme une fonction $\mathcal{P}$ qui à une donnée $x \in \mathbb{R}^n$ associe la solution du problème $\mathcal{P}(x) \in \mathbb{R}^p$, la fonction $\mathcal{P}$ étant continue sur l'ensemble des données qui sera typiquement $\mathbb{R}^n$ ou un ouvert de $\mathbb{R}^n$.

On souhaite quantifier l'influence de perturbations dans x sur le résultat $\mathcal{P}(x)$. En se ramenant aux fonctions coordonnées, on pourra toujours supposer que l'espace d'arrivée est $\mathbb{R}$. Considérons donc une application continue $\mathcal{P} : U \to \mathbb{R}$ où U est un ouvert de $\mathbb{R}^n$, et qui représente un calcul à effectuer sur ordinateur, dont la précision relative est ε.

Définition 1.1 *Soit $x = (x_1, \ldots, x_n) \in U$ tel que $x_i \neq 0$ $\forall i$ et $\mathcal{P}(x) \neq 0$. La condition κ_x du problème $\mathcal{P}(x)$ est le plus petit nombre tel que*

$$\forall \hat{x} = (\hat{x}_1, \ldots, \hat{x}_n) \in U, \quad \max_{1 \leq i \leq n} \frac{|\hat{x}_i - x_i|}{|x_i|} \leq \varepsilon \Rightarrow \frac{|\mathcal{P}(\hat{x}) - \mathcal{P}(x)|}{|\mathcal{P}(x)|} \leq \kappa_x \varepsilon.$$

On dit que le problème $\mathcal{P}(x)$ est **bien conditionné** *si κ_x n'est pas trop grand. Sinon, il est* **mal conditionné**.

La condition d'un problème dépend donc de la précision de l'ordinateur à notre disposition, ainsi que de la signification de "pas trop grand". On souhaite surtout que $\kappa\varepsilon$ reste très inférieur à 1. On peut éventuellement avoir $\kappa = +\infty$ dans la définition ci-dessus. On notera $\kappa_x = \kappa$ pour simplifier. Remarquons que

$$\kappa = \frac{1}{\varepsilon |\mathcal{P}(x)|} \sup_{\hat{x} \in K} |\mathcal{P}(\hat{x}) - \mathcal{P}(x)|,$$

où

$$K = \{\hat{x} \in U, \ \max_{1 \leq i \leq n} \frac{|\hat{x}_i - x_i|}{|x_i|} \leq \varepsilon\}.$$

En particulier, si $\mathcal{P}$ est continue sur $\overline{U}$ alors $\overline{K}$ est compact et $\kappa < \infty$.

Addition de deux nombres réels

Soient donnés les nombres x_1 et x_2, et considérons le problème de calculer $\mathcal{P}(x_1, x_2) = x_1 + x_2$. Pour les deux valeurs perturbées

$$\hat{x_1} = x_1(1 + \epsilon_1), \quad \hat{x_2} = x_2(1 + \epsilon_2), \quad |\epsilon_i| \leq \varepsilon,$$

on a

$$\left| \frac{(\hat{x_1} + \hat{x_2}) - (x_1 + x_2)}{x_1 + x_2} \right| = \left| \frac{x_1\varepsilon_1 + x_2\varepsilon_2}{x_1 + x_2} \right| \leq \frac{|x_1| + |x_2|}{|x_1 + x_2|} \cdot \varepsilon.$$

La dernière inégalité est une égalité si $\varepsilon_i = signe(x_i)\varepsilon$, de sorte que

$$\kappa = \frac{|x_1| + |x_2|}{|x_1 + x_2|}.$$

Si $signe(x_1) = signe(x_2)$, alors $\kappa = 1$ et le problème est bien conditionné. Par contre, si $x_2 \approx -x_1$, la condition κ devient très grande et on est confronté à un problème mal conditionné. Ce mauvais conditionnement explique le phénomène de compensation signalé ci-dessus.

Produit de deux nombres réels

Avec les mêmes notations que précédemment, on souhaite déterminer la condition du calcul x_1x_2. On trouve

$$\left|\frac{\hat{x}_1\hat{x}_2 - x_1x_2}{x_1x_2}\right| = |(1+\varepsilon_1)(1+\varepsilon_2) - 1| = |\varepsilon_1 + \varepsilon_2 + \varepsilon_1\varepsilon_2| \leq (2+\varepsilon)\varepsilon.$$

La dernière inégalité est une égalité si $\varepsilon_1 = \varepsilon_2 = \varepsilon$, donc $\kappa = 2 + \varepsilon \approx 2$ et le problème est bien conditionné.

1.3 Condition d'une matrice

Pour étudier la condition du problème "résoudre $Ax = b$", on a besoin de normes de vecteurs et de matrices. On rappelle que pour une matrice A à m lignes et n colones, la norme de A est définie par

$$\|A\| = \sup_{x\neq 0} \frac{\|Ax\|}{\|x\|} = \max_{\|x\|=1} \|Ax\|,$$

où on a évidemment choisi auparavant des normes dans $\mathbb{R}^n$ et $\mathbb{R}^m$. Sur la définition même, on voit que

$$\|Ax\| \leq \|A\|\ \|x\| \quad \forall x \in \mathbb{R}^n.$$

La norme d'une matrice possède toutes les propriétés d'une norme. De plus, elle vérifie $\|I\| = 1$ pour la matrice identité et $\|A \cdot B\| \leq \|A\| \cdot \|B\|$.

Si on prend la même norme dans les deux espaces, alors,
pour $\|x\|_1 = \sum_{i=1}^n |x_i|$, on a

$$\|A\|_1 = \max_{j=1,\ldots,n} \left(\sum_{i=1}^m |a_{ij}| \right); \tag{2}$$

pour $\|x\|_2 = (\sum_{i=1}^n |x_i|^2)^{1/2}$, on a

$$\|A\|_2 = \sqrt{\text{plus grande valeur propre de } {}^tA \cdot A}; \tag{3}$$

pour $\|x\|_\infty = \max_{i=1,\ldots,n} |x_i|$, on a

$$\|A\|_\infty = \max_{i=1,\ldots,m} \left(\sum_{j=1}^n |a_{ij}| \right). \tag{4}$$

On appelle **condition de la matrice** A (relativement à la norme matricielle considérée) le nombre

$$\kappa(A) = \|A\|\|A^{-1}\|.$$

Si on considère un système linéaire $Ax = b$, ce nombre mesure la sensibilité de la solution x du système linéaire vis-à-vis des variations sur les données A et b. On a plus précisément le résultat fondamental suivant, où $\|\cdot\|$ est une norme quelconque sur $\mathbb{R}^n$.

Théorème 1.2 *Soient x, $\hat{x}$ les solutions des systèmes linéaires $Ax = b$ et $A\hat{x} = \hat{b}$, où A est une matrice inversible, et $b, \hat{b}$ sont des vecteurs de $\mathbb{R}^n$ ($b \neq 0$). Alors*

$$\frac{\|\hat{x} - x\|}{\|x\|} \leq \kappa(A) \frac{\|\hat{b} - b\|}{\|b\|}.$$

De plus, il existe des vecteurs b et $\hat{b}$ pour lesquels cette inégalité est une égalité

Ce résultat dit que l'erreur relative commise sur x est au plus de l'ordre de l'erreur relative sur b multiplié par $\kappa(A)$. Remarquons que ce résultat est un peu moins contraignant que celui demandé dans la définition de la condition d'un problème (pour laquelle on regardait l'erreur relative composante par composante).
Preuve. Comme $\hat{x} - x = A^{-1}(\hat{b} - b)$ et $b = Ax$,

$$\|\hat{x} - x\| \leq \|A^{-1}\|\|\hat{b} - b\| \text{ et } \|b\| \leq \|A\|\|x\|,$$

d'où le résultat. Pour le cas d'égalité, il suffit de choisir $x \neq 0$ tel que $\|Ax\| = \|A\|\|x\|$ et $\hat{b} \neq b$ tel que $\|A^{-1}(\hat{b} - b)\| = \|A^{-1}\|\|\hat{b} - b\|$, ce qui est possible d'après la définition d'une norme matricielle.

Il est facile de vérifier que pour tout matrice A inversible, $\kappa(A) \geq \kappa(I) = 1$, $\kappa(\alpha A) = \kappa(A)$ pour $\alpha \neq 0$ et $\kappa(A^{-1}) = \kappa(A)$. Si A est une matrice symétrique définie positive, et que l'on choisit la norme $\|\cdot\|_2$,

$$\kappa_2(A) = \frac{\lambda_n(A)}{\lambda_1(A)},$$

où $\lambda_n(A)$ et $\lambda_1(A)$ désignent la plus grande et la plus petite valeur propre de A respectivement.

Si $U \in O(n)$, alors $\kappa_2(U) = 1$. Un exemple célèbre de matrice mal conditionnée est la matrice de Van der Monde.

1.4 Exercices mathématiques

1. Déterminez la condition du calcul des racines du polynôme $x^2 - 2x + q$, avec $q \in]0, 1[$. Montrer que le calcul devient mal conditionné lorsque q se rapproche de 1, c'est-à-dire lorsque l'on se rapproche d'une racine double.
2. Etablir les relations (2), (3) et (4).
3. Quelle est la condition d'une matrice diagonale, en norme $\|\cdot\|_p$, $p = 1, 2$ et ∞ ? Donner un encadrement de $\kappa(A^n)$ pour une matrice A symétrique.
4. Montrer que $A \mapsto tr({}^t\overline{A} \cdot A)$ est une norme sur l'espace des matrices complexes de taille $m * n$, puis montrer qu'il ne s'agit pas d'une norme subordonnée.

1.5 Exercices de programmation

1. Vérifier que $log_2(\%eps) = -52$. Puis, retrouvez cette précision à l'aide d'une soustraction (dans la ligne de commande Scilab).
2. A l'aide d'une opération ou d'un programme élémentaire en SCILAB, déterminez le plus petit réel strictement positif, et le plus grand réel représentables. Trouve-t-on le résultat attendu ? En particulier, comment déduire de ces nombres le nombre de chiffres en écriture binaire de la mantisse M et de l'exposant E.
3. Calculez le conditionnement en norme $\|\cdot\|_2$, $\|\cdot\|_1$ et $\|\cdot\|_\infty$ de la matrice $A = \begin{pmatrix} 1 & 10 \\ 0 & 1 \end{pmatrix}$.
4. Ecrire un programme qui trace sur un graphique la condition de la matrice A_N suivante en fonction de N, pour les normes $\|\cdot\|_2$ et $\|\cdot\|_1$:

$$A_N = \frac{1}{(N+1)^2} \begin{pmatrix} 2 & -1 & 0 & \dots & 0 \\ -1 & 2 & -1 & & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & -1 \\ 0 & \dots & 0 & -1 & 2 \end{pmatrix}.$$

Vérifier numériquement que $\kappa_2(A_N) \sim 4N^2/\pi^2$ quand $N \to +\infty$.
5. Ecrire un programme qui tracer le conditionnement en fonction de N pour la matrice de Van der Monde

$$V[x_0, x_1, \ldots, x_n] = \begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \vdots & & & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{pmatrix}$$

pour des points équirépartis sur $[0, 1]$, $x_i = i/n$, $i = 0, 1, \ldots, n$. On tracera sur un même graphique les conditions en norme $\|\cdot\|_2$, $\|\cdot\|_1$ et $\|\cdot\|_\infty$. Même question avec les points de Tchebychev $x_i = \cos \frac{(2i+1)}{2n+2}\pi$, $i = 0, 1, \ldots, n$.
6. Comment l'ordinateur calcule-t-il un conditionnement ?
7. Programmer la suite récurrente (1). Le comportement est-il celui attendu ?

2 Résolution de systèmes linéaires

Pour les références, on pourra consulter [PR, Cia, TL].

Le problème est de trouver $x \in \mathbb{R}^n$ solution de $Ax = b$ où $A \in M_n(\mathbb{R})$ est une matrice carrée à coefficients réels (ou plus généralement complexes), que l'on suppose inversible, et $b \in \mathbb{R}^n$.

Dans cette section, on considère tout d'abord les méthodes directes de résoution de ce type de système, basées sur la méthode de Gauss, puis des méthodes itératives qui sont utiles notamment lorsque les méthodes directes ne sont plus applicables (par exemple pour des raison de taille du système).

Enfin, nous évoquerons la méthode des moindres carrés qui sert à résoudre $Ax = b$ lorsque la matrice A est rectangulaire (cas des systèmes surdéterminés, notamment).

2.1 Factorisation LU

De la méthode de Gauss à la factorisation LU

On rappelle la

Définition 2.1 *Une matrice carrée $L = (l_{ij})_{1 \leq i,j \leq n}$ est dite triangulaire inférieure si tous les l_{ij} avec $j > i$ sont nuls.*

Une combinaison linéaire de matrices triangulaires inférieures est une matrice triangulaire inférieure. Le produit de deux matrices triangulaire inférieure est également une matrice triangulaire inférieur. En termes savants, l'ensemble des matrices triangulaires inférieures est une sous-algèbre de $M_n(\mathbb{R})$ (ou $M_n(\mathbb{C})$). De même, l'ensemble des matrices triangulaires inférieures inversible est un sous-groupe de $GL_n(\mathbb{R})$.

Les termes a priori non nuls de L sont dans le triangle inférieur, d'où la notation L pour "Lower". On a la même définition et les mêmes considérations avec triangulaire supérieure, notée U pour "upper".

Considérons sur *un exemple* la méthode de Gauss. On va écrire sous forme matricielle les diverses opérations qui mènent du système $AX = X'$ à un système triangulaire supérieur pour la matrice

$$A = \begin{pmatrix} 4 & 8 & 12 \\ 3 & 8 & 13 \\ 2 & 9 & 18 \end{pmatrix},$$

on part de

$$\left[\begin{array}{ccc|ccc} 4 & 8 & 12 & 1 & 0 & 0 \\ 3 & 8 & 13 & 0 & 1 & 0 \\ 2 & 9 & 18 & 0 & 0 & 1 \end{array}\right] \begin{array}{l} \\ l_2 \leftarrow l_2 - (3/4)l_1 \\ l_3 \leftarrow l_3 - (2/4)l_1 \end{array}$$

La grande matrice rectangulaire représente l'équation $AX = I_3X'$, où X et X' sont deux vecteurs de $\mathbb{R}^3$. En faisant les opérations indiquées sur les lignes, on obtient

$$\left[\begin{array}{ccc|ccc} 4 & 8 & 12 & 1 & 0 & 0 \\ 0 & 2 & 4 & -3/4 & 1 & 0 \\ 0 & 5 & 12 & -1/2 & 0 & 1 \end{array}\right] \begin{array}{l} \\ \\ l_3 \leftarrow l_3 - 5/2l_2 \end{array}$$

Si on note L_1 la matrice triangulaire inférieur de droite ci-dessus, la grande matrice rectangulaire traduit l'équation $L_1AX = L_1X'$. On a multiplié à gauche par une matrice triangulaire inférieure.

La dernière étape indiquée ci-dessus donne

$$\left[\begin{array}{ccc|ccc} 4 & 8 & 12 & 1 & 0 & 0 \\ 0 & 2 & 4 & -3/4 & 1 & 0 \\ 0 & 0 & 2 & 11/8 & -5/2 & 1 \end{array}\right],$$

Cela traduit l'équation $L_2L_1AX = L_2L_1X'$, où L_2 est la matrice triangulaire inférieure correspondant aux opérations faites, i.e.

$$L_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -5/2 & 1 \end{pmatrix}$$

On a ainsi $L_2L_1A = U$ avec U triangulaire supérieure. En mulpliant à gauche par $L = (L_2L_1)^{-1}$, on obtient bien une écriture

$$A = LU$$

avec L triangulaire inférieure ne comportant que des 1 sur la diagonale.

Calculer L parait fastidieux. Cependant, on remarque la chose suivante.

$$L_1^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 3/4 & 1 & 0 \\ 1/2 & 0 & 1 \end{pmatrix} \quad L_2^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 5/2 & 1 \end{pmatrix},$$

et

$$L = L_1^{-1}L_2^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 3/4 & 1 & 0 \\ 1/2 & 5/2 & 1 \end{pmatrix}$$

Ainsi, les coefficients non triviaux de L sont constitué des coefficients que l'on a placé devant les pivots dans les différentes opérations. Il n'y a donc pas de calcul supplémentaire à faire pour calculer L !

Cette façon de considérer la méthode de Gauss est tout à fait générale, *à condition que le pivot soit non nul à chaque étape.* Cela conduit au

Théorème 2.2 (Factorisation LU d'une matrice) *Soit $A = (a_{ij})_{1\leq i,j\leq n}$ une matrice carrée d'ordre n telle que les n sous-matrices de taille k,*

$$\Delta_k = \begin{pmatrix} a_{11} & \dots & a_{1k} \\ \vdots & & \vdots \\ a_{k1} & \dots & a_{kk} \end{pmatrix} \quad k = 1, \dots, n,$$

soient inversibles. Alors il existe une matrice triangulaire inférieure $L = (l_{ij})_{1\leq i,j\leq n}$ avec des 1 sur la diagonale et une matrice triangulaire supérieure U telles que $A = LU$. De plus, une telle factorisation est unique.

Preuve. L'existence est donnée par la méthode de Gauss, et l'unicité se traite à part, en 2 lignes. ■

Remarquons que si A est inversible sans l'hypothèse sur les sous-matrices, on peut être obligée pour la méthode de Gauss d'effectuer des permutations sur les lignes lors de la recherche du pivot. Dans ce cas on peut obtenir une décomposition $PA = LU$ où P est une matrice de permutation. Mais une telle décomposition n'est plus unique.

Coût de l'élimination de Gauss

Le tableau ci-dessous rassemble quelques valeurs de coûts de calcul.

A représente une matrice carrée "pleine" de taille n, U une matrice triangulaire supérieur (de taille n) et L une matrice triangulaire inférieure avec des 1 sur la diagonale, et x un vecteur de $\mathbb{R}^n$.

1 op= 1 opération = 1 addition ou 1 multiplication ou 1 division
NB : 1 soustraction = 1 addition

	x	+	/	op
Produit Ax	n^2	n(n-1)		$2n^2$-n
Produit Ux	n(n+1)/2	n(n-1)/2		n^2
Produit Lx	n(n-1)/2	n(n-1)/2		n^2-n
Calcul de $U^{-1}x$	n(n-1)/2	n(n-1)/2	n	n^2
Calcul de $L^{-1}x$	n(n-1)/2	n(n-1)/2		n^2-n
Factorisation $A=LU$	$\sum_{k=1}^{n}(n-k)^2$	$\sum_{k=1}^{n}(n-k)^2$	$\sum_{k=1}^{n}(n-k)$	$\sim 2n^3/3$

Pour la résolution de $Ax = b$ par la méthode LU, on peut ainsi faire la factorisation (de l'ordre de $n^3/3$ additions + multiplications), puis résoudre deux système triangulaires $Ly = b$ puis $Ux = y$ (chacun de l'ordre de $n^2/2$ multiplications + additions).

Détaillons le calcul de la factorisation LU. Pour une matrice $A = (a_{ij})$ de taille n, la première étape de la méthode de Gauss consiste à remplacer la ligne l_i par $l_i - a_{i1}/a_{11}l_1$: pour chaque ligne l_i on a donc besoin de 1 division (calcul de $c_i = a_{i1}/a_{11}$), $n-1$ multiplications (c_i* jème terme de l_i sauf $j = 1$), et $n-1$ additions. Donc $(n-1)$ divisions et $(n-1)^2$ multiplications et additions pour la première étape.

Pour compter le nombre d'opérations à l'étape k, notons $A^{(k)}$ la matrice obtenue à l'itération k ($1 \leq k \leq n$) et $b^{(k)}$ le second membre. On a $A^{(1)} = A$, $b^{(1)} = b$ et

$$[A^{(k)}, b^{(k)}] = \left[\begin{array}{ccccccc|c} a_{11}^{(k)} & \dots & a_{k1}^{(k)} & \times & \dots & \times & b_1^{(k)} \\ 0 & \ddots & \vdots & \vdots & & \vdots & \vdots \\ \vdots & 0 & a_{kk}^{(k)} & & & & \\ 0 & 0 & a_{k+1,k}^{(k)} & & & & \\ \vdots & & & & & & \\ 0 & 0 & a_{n,k}^{(k)} & & & a_{nn}^{(k)} & b_n^{(k)} \end{array}\right]$$

Pour la ligne $l_i^{(k)}$ de cette matrice, avec $k+1 \leq i \leq n$, on fait l'opération

$$l_i^{(k+1)} = l_i^{(k)} - (a_{i,k}^{(k)}/a_{kk}^{(k)})l_k^{(k)}. \tag{5}$$

Pour chaque i, cela représente, pour la partie factorisation (sans le second membre) : 1 division, $n-k$ multiplications et $n-k$ additions, et cela pour $n-k$ lignes (on ne calcule pas le 1er coefficient, qui va etre nul !). En sommant sur k de 1 à n, on obtient le résultat (il n'y a pas de calcul supplémentaire pour obtenir L, cf. remarque ci-dessous).

Par calcul, on obtient

$$\sum_{k=1}^{n}(n-k) = n(n-1)/2 \quad \text{et} \quad \sum_{k=1}^{n}(n-k)^2 = \frac{1}{3}n(n-1/2)(n-1).$$

Pour la somme des carrés, ce calcul exact peut être avantageusement remplacé par l'expression asymptotique (qui nous suffit)

$$(n-1)^2 + (n-2)^2 + \ldots + 1 \sim \int_0^n x^2 dx \sim \frac{1}{3}n^3.$$

opérations, où l'équivalent est quand $n \to +\infty$ (comparaison entre une série et une intégrale d'une fonction positive croissante).

Pour une matrice pleine, la factorisation LU (ou PLU) nécessite de l'ordre de $2/3n^3$ opérations. Ainsi, la méthode de Gauss permet de calculer un déterminant en $O(n^3)$ opérations (pour

une matrice pleine), en remarquant que $det(P)det(A) = det(U)$. C'est beaucoup mieux que les formules de Cramer, pour lesquelles on a besoin de $n! * n$ multiplications et $n! - 1$ additions.

Par contre, une fois connus L et U, qu'on garde sous la forme $\tilde{L}A = U$, la résolution de $Ax = b$ se fait en deux étapes : calcul de $c = \tilde{L}b$, puis résolution de $Ux = c$. Chacune de ces étapes nécessite de l'ordre de n^2 opérations. Si on a plusieurs systèmes linéaires à résoudre de même matrice A, on fait 1 fois le calcul $\tilde{L}A = U$, puis on applique cette résolution à chaque vecteur. Si on a K systèmes ainsi à résoudre, on est en $O(n^3) + O(Kn^2)$ opérations (plutôt que $O(Kn^3)$).

Remarque : pour le calcul de L, on remarque que la ligne (5) s'écrit aussi

$$l_i^{(k+1)} + (a_{i,k}^{(k)}/a_{kk}^{(k)})l_k^{(k+1)} = l_i^{(k)}, \quad i > k,$$

car $l_k^{(k+1)} = l_k^{(k)}$. Matriciellement, on peut récrire cela

$$L_k A^{(k+1)} = A^{(k)} \quad \text{avec} \quad L_k = \begin{pmatrix} 1 & 0 & \dots & 0 & \dots & 0 \\ 0 & \ddots & \ddots & & & \\ \vdots & \ddots & 1 & \ddots & & \\ \vdots & 0 & r_{i,k}^{(k)} & 1 & 0 & \\ \vdots & \vdots & \vdots & 0 & \ddots & \\ 0 & \dots & r_{n,k}^{(k)} & 0 & \dots & 1 \end{pmatrix},$$

où $r_{i,k}^{(k)} = a_{i,k}^{(k)}/a_{kk}^{(k)}$. Comme on a déjà calculé ce coefficient (c'est la division pour la ligne i), il n'y a pas de calcul supplémentaire. Par récurrence,

$$L_n L_{n^1} \cdots L_1 A^{(n)} = A,$$

avec $A^{(n)} = U$ triangulaire supérieure. La matrice

$$L = L_n L_{n-1} \cdots L_1$$

est la matrice triangulaire inférieure cherchée. On vérifie qu'elle est simplement composée des coefficients $r_{i,k}^{(k)}$ pour $i > k$ à la ligne i et à la colonne k. Il n'y a donc pas de calcul supplémentaire à faire.

Largeur de bande

Définition 2.3 *Une matrice $A = (a_{ij})_{1 \le i \le n}$ est une matrice-bande de largeur de bande l si*

$$\forall i, j \quad |j - i| > l \Rightarrow a_{ij} = 0.$$

En d'autres termes, seuls les coefficients des diagonales k avec $|k| \le l$ sont (a priori) non nuls. Géométriquement, l représente plutôt la demi-largeur de bande. Par exemple, une matrice tri-diagonale est une matrice de largeur de bande 1.

Un point important en pratique, pour réduire le nombre de calculs : la factorisation LU garde la structure par bandes : si A est une matrice de largeur de bande l, alors les matrices L ($\tilde{L}$) et U ont la même largeur de bande.

Si une matrice A est symétrique définie positive, elle vérifie les conditions d'applications du théorème précédent car les matrices Δ_k sont symétriques définies positives. Une telle matrice admet donc une décomposition LU. Mais on peut faire un peu mieux grâce à la symétrie, et l'écrire sous la forme $A = B \cdot {}^tB$: c'est la factorisation de Cholesky : algorithme en $O(n^3)$, et la structure-bande est également conservée.

2.2 Méthodes itératives de résolution de systèmes linéaires

Généralités

Pour résoudre un système linéaire $Ax = b$ où A est une matrice carrée de taille n, les méthodes directes comme la méthode de Gauss ne sont pas toujours applicables (matrice pleine de grande taille, le nombre d'opérations est $\sim 2/3n^3$) ou souhaitable (une valeur approchée de la solution suffit, par exemple). Dans ce cas, on a recours à des méthodes de point fixe, également appelées méthodes itératives.

On met le système sous une forme équivalente $x = Bx + c$, (cf. ci-après pour des exemples) et à partir d'un élément x_0, on construit la suite

$$x^{k+1} = Bx^k + c,$$

où B est une matrice carrée de taille n, c un vecteur de $\mathbb{R}^n$ et $x \in \mathbb{R}^n$ (ou $\mathbb{C}^n$). Si la suite converge, sa limite vérifie

$$x = Bx + c,$$

de sorte que par différence, l'erreur $e^k = x^k - x$ vérifie $e^{k+1} = Be^k$, et par récurrence $e^k = B^k e^0$. On en déduit

Proposition 2.4 *On suppose que l'équation $x = Bx + c$ admet une unique solution x (i.e. que B n'admet pas la valeur propre 1). La suite (x^k) définie par $x^{k+1} = Bx^k + c$ converge (vers x) pour tout choix de l'élément initial x^0 si et seulement si la suite de matrices (B^k) tend vers 0.*

Les deux question principales qui se posent sont :

1. Etant donnée une méthode itérative de matrice B, déterminer si la méthode est convergente, i.e. si la suite (x^k) définie par $x^{k+1} = Bx^k + c$ converge pour tout choix de l'élément initial x^0 ;

2. Etant données deux méthodes itératives convergentes, comparer leur vitesse de convergence.

L'étude qui suit donne une réponse satisfaisante à ces questions, en terme de rayon spectral de B.

On rappelle que si $|\cdot|$ désigne une norme sur $\mathbb{C}^n$, la norme matricielle subordonnée est définie pour $B \in M_n(\mathbb{C})$ par

$$\|B\| = \sup_{x \neq 0} \frac{|Bx|}{|x|} = \max_{|x|=1} |Bx|.$$

Le rayon spectral $\rho(B)$ de $B \in M_n(\mathbb{C})$ est défini par

$$\rho(B) = \max_{1 \leq i \leq n} |\lambda_i(B)|,$$

où les $\lambda_i(B)$ sont les valeurs propres de la matrice B. On a clairement

$$\rho(B) \leq \|B\| \tag{6}$$

pour toute norme matricielle subordonnée (considérer un vecteur propre correspondant à une valeur propre de module égal à $\rho(B)$). L'inégalité inverse n'est pas vraie, comme le montre l'exemple $A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$. En effet $\rho(A) = 0$ et $\|A\| > 0$ pour toute norme matricielle car A est non nulle. En revanche, on a le

Lemme 2.5 *Soit $B \in M_n(\mathbb{C})$. Pour tout $\varepsilon > 0$, il existe une norme matricielle subordonnée à une norme vectorielle sur $\mathbb{C}^n$ telle que $\|B\| \leq \rho(B) + \varepsilon$.*

On en déduit

Théorème 2.6 (CNS de convergence) *Soit $B \in M_n(\mathbb{C})$. La suite (B^k) des puissances de B tend vers 0 si et seulement si $\rho(B) < 1$.*

Dans ce cas, la convergence est d'autant plus rapide que $\rho(B)$ est petite, dans le sens où (pour toute norme matricielle subordonnée)

$$\lim_{k\to\infty} \|B^k\|^{1/k} = \rho(B). \tag{7}$$

Moralement, (7) dit que $\|B^k\|$ se comporte comme $\rho(B)^k$ (suite géométrique).
Preuve. Par trigonalisation de B on voit que $\rho(B^k) = \rho(B)^k$. Donc si $\rho(B) \geq 1$, $\|B^k\| \geq \rho(B^k) \geq 1$ par (6) pour tout k, et la suite (B^k) ne tend pas vers 0.

Inversement, si $\rho(B) < 1$, d'après le lemme, en choisissant $\varepsilon > 0$ assez petit, on peut trouver une norme subordonnée telle que $\|B\| < \rho(B) + \varepsilon < 1$. D'où $\|B^k\| \leq \|B\|^k \to 0$ quand $k \to \infty$.

Il reste à prouver (7). Si $\|\cdot\|$ est une norme matricielle subordonnée,

$$\|B^k\| \geq \rho(B^k) = \rho(B)^k.$$

D'autre part, soit $\varepsilon > 0$: $\rho(\dfrac{B}{\rho(B)+\varepsilon}) < 1$, donc $\dfrac{\|B^k\|}{(\rho(B)+\varepsilon)^k}$ tend vers 0 quand k tend vers ∞. Pour k assez grand, on a ainsi

$$\rho(B)^k \leq \|B^k\| \leq (\rho(B)+\varepsilon)^k,$$

ce qui achève la démonstration. ■

Deux exemples

Exemple 1 : la méthode de Jacobi
Pour résoudre $Ax = b$, on écrit $A = D + F$, où D est la diagonale de A, supposée inversible, et $F = A - D$ composée des parties triangulaire inférieure et triangulaire supérieur de A. On a

$$Ax = b \iff Dx = -Fx + b \iff x = -D^{-1}Fx + D^{-1}b,$$

de sorte que la méthode converge si et seulement si $\rho(D^{-1}F) < 1$. On peut alors montrer (admis) que si A est symétrique définie positive et tridiagonale, cette condition est vérifiée.

Exemple 2 : méthode du gradient à pas fixe
Si A est une matrice symétrique définie positive, l'algorithme de gradient à pas fixe pour la résolution de $Ax = b$ s'écrit

$$x^{k+1} = x^k - \mu(Ax^k - b),$$

où $\mu \in \mathbb{R}$ est un paramètre à fixer. D'après ce qui précède, il converge pour toute donnée initiale x^0 ssi $\rho(I - \mu A) < 1$, i.e. $0 < \mu < 2/\lambda_i(A)$ pour toute valeur propre $\lambda_i(A)$ de A. L'algorithme converge alors vers la solution de $Ax = b$, qui est également l'unique minimiseur de $J(x) = \dfrac{1}{2}\langle x, Ax\rangle - \langle b, x\rangle$ sur $\mathbb{R}^n$ (exercice). On a un algorithme de minimisation.

2.3 Méthodes des moindres carrés

On souhaite résoudre

$$Ax = b \tag{8}$$

mais cette fois la matrice $A \in M_{m,n}(\mathbb{R})$ est *rectangulaire*, $b \in \mathbb{R}^m$ et $x \in \mathbb{R}^n$. C'est le cas notamment des système *surdéterminés* pour lesquels $m \geq n$, i.e. on a plus d'équation que d'inconnus.

Un tel système n'a pas de solution en général. On le remplace par le problème suivant :

$$(P) \text{ trouver } x \in \mathbb{R}^n \text{ qui minimise } y \mapsto \|Ay - b\|_2 \text{ dans } \mathbb{R}^n.$$

On utilise la norme euclidienne $\|\cdot\|$ pour avoir un problème différentiable (en fait, on élève au carré pour avoir un problème différentiable, mais cela revient au même en terme de minimisation).

Il est clair que si x est solution de (8), alors x est solution de (P). Par contre, (P) a toujours une solution tandis que (8) n'en a pas en général.

Théorème 2.7 *Le problème (P) a toujours au moins une solution $x \in \mathbb{R}^n$. De plus x est solution de (P) si et seulement si x vérifie*

$$A^t \cdot Ax = A^t \cdot b. \tag{9}$$

Les équations (9) sont appelées *équations normales.*

Preuve. En posant $z = Ay$, trouver x solution de (P) revient à trouver $\bar{z}$ qui minimise $\|z - b\|_2$ quand z parcourt l'espace vectoriel

$$ImA = \{z \in \mathbb{R}^m \ : \ \exists y \in \mathbb{R}^n, \ z = Ay\}.$$

C'est un problème classique : on sait qu'il existe un unique $\bar{z}$ solution de ce problème : $\bar{z}$ est la projection orthogonale de b sur ImA. Comme $\bar{z} \in ImA$, on a $\bar{z} = Ax$ pour un $x \in \mathbb{R}^n$, d'où notre solution.

En tant que projection orthogonale, $\bar{z}$ est l'unique élément de ImA tel que

$$(\bar{z} - b)^t \cdot z = 0 \quad \forall z \in ImA.$$

De sorte que toute solution $x \in \mathbb{R}^n$ est caractérisée par

$$(Ax - b)^t \cdot Ay = 0 \quad \forall y \in \mathbb{R}^n,$$

et cette dernière équation est équivalent à (9).

Rappelons le résultat classique suivant

Lemme 2.8 *Le rang de A^tA est égal au rang de A. En particulier, si A est de rang n (et dans ce cas, $m \geq n$ nécessairement), alors $A^tA \in GL_n(\mathbb{R})$ et les équations* (9) *ont une unique solution.*

Preuve. En effet, il est facile de vérifier que $KerA = Ker(A^tA)$. On en déduit le résultat avec le théorème du rang.

Application : on a m point du plan (x_i, y_i). On cherche la droite du plan d'équation $y = ax+b$ qui passe au mieux par ces points. Si les points sont alignés, on a $y_i = ax_i + b$ pour tout $i \in \{1, \ldots, m\}$. En d'autres termes, on cherche (a, b) solution d'un problème linéaire de matrice

$$A = \begin{pmatrix} x_1 & 1 \\ \vdots & \vdots \\ x_m & 1 \end{pmatrix} \text{ et de second membre } b = \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix}.$$

Dans le cas général, il s'agit d'un système surdéterminé que l'on résout par moindres carrés.

D'après ce qui précède, il y a toujours une solution, et la solution est unique si $rang(A) = 2$, i.e. si au moins deux x_i sont distincts. La solution est appelée *droite de régression.*

2.4 Exercices mathématiques

0. Si $A = LU$ est la décomposition LU de la matrice A, quelle est la décomposition de la matrice αA ?
1. Montrer que $A \mapsto tr({}^t\overline{A} \cdot A)$ est une norme sur l'espace des matrices complexes de taille $m * n$, puis montrer qu'il ne s'agit pas d'une norme subordonnée.
2. Calculer le coût (en nombre d'opérations) du calcul de l'inverse d'une matrice A par la méthode de Gauss.
3. Calculer le coût de la décomposition LU d'une matrice bande (indication : $O(Nl^2)$, cf. référence ci-dessous). Même question pour la décomposition de Cholesky.
4. Quels sont les "droites de régression" dans le cas où tous les x_i sont égaux ?
5. On considère la droite de régression D associée à m points du plan (x_i, y_i) et la droite de régression D' associée aux points (y_i, x_i). On note enfin D'' l'image de D' par la transformation $(y, x) \mapsto (x, y)$. Montrer que $D = D''$ si et seulement si les points (x_i, y_i) sont alignés. Montrer que $0 \leq aa' \leq 1$, où a est le coefficient directeur de D et a' celui de D', et étudiez les cas d'égalité.

2.5 Exercices de programmation

1. Examinez la décomposition LU (`lu`) et de Cholesky (`chol`) des matrices suivantes :

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 1 \\ 2 & 3 \end{pmatrix}, \quad C = \begin{pmatrix} 4 & 1 \\ 1 & 4 \end{pmatrix}.$$

2. Faire un programme `tp2exo2.sce` qui :
- définit la matrice de taille N suivante :

$$A_N = \begin{pmatrix} 2 & -1 & 0 & \dots & 0 \\ -1 & 2 & -1 & & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & -1 \\ 0 & \dots & 0 & -1 & 2 \end{pmatrix};$$

- calcule sa décomposition LU et indique le temps de calcul nécessaire (`tic`, `toc`) ;
- calcule sa décomposition de Cholesky et indique le temps de calcul nécessaire.

Observez la structure-bande des matrices obtenues.
3. Dans le programme précédent, quelle taille maximale de matrice est-elle possible ? Pour quelle raison ? Pour N grand, quelle est la décomposition la plus rapide, LU ou Cholesky ? Pourquoi ?
4. Programmer en Scilab l'algorithme de Jacobi sous la forme d'une fonction `function x=Jacob(A,b, x0,kmax)`, où A est une matrice, b un vecteur, $x0$ la donnée initiale et $kmax$ le nombre d'itérations. Le tester sur deux ou trois exemples. Observer la convergence et trouver également un cas pour laquel l'algorithme de Jacobi ne converge pas.
5. Pour l'algorithme de Jacobi, faire un programme `tp2exo5.sce` qui trace l'erreur $\|x^k - x\|$ en fonction de k, pour un cas-test dont vous connaissez la solution x ; x^k est le $k$ème itéré de l'algorithme de Jacobi. Mettez en évidence la vitesse de convergence (échelle logarithmiques).
6. Reprenez la question 4 avec l'algorithme du gradient à pas fixe.
7. Reprenez la question 5 avec l'algorithme du gradient à pas fixe. Tracez cette fois sur un même graphique l'erreur $\|x^k - x\|$ en fonction de k, pour trois pas différents (de manière à mettre en évidence l'influence du choix du pas sur la vitesse de convergence).
8. Déterminer une droite de régression pour m points. Retrouvez le résultat à l'aide de la fonction `regress` de Scilab.

3 Calcul numérique de valeurs propres

On pourra consulter [TL, PR].

3.1 Introduction

Nous cherchons à calculer numériquement les valeurs propres et vecteurs propres d'une matrice carrée A de taille N.

Remarquons d'abord que le calcul de valeurs propres d'une matrice de taille N est une affaire délicate en général, car il équivaut à trouver les racines d'un polynôme de degré N. En effet, les valeurs propres de A sont les racines de son polynôme caractéristique $p_A(\lambda) = det(A - \lambda I)$. Réciproquement, étant donné un polynôme de degré N

$$p(\lambda) = \lambda^N + a_{N-1}\lambda^{N-1} + \cdots + a_1\lambda + a_0,$$

une manière de trouver ses racines est de chercher les valeurs propres de sa matrice compagnon (matrice de Frobenius)

$$A = \begin{pmatrix} -a_{N-1} & -a_{N-2} & \cdots & \cdots & -a_0 \\ 1 & 0 & & & \\ & 1 & 0 & & \\ & & \ddots & \ddots & \\ & & & 1 & 0 \end{pmatrix}.$$

Pour $N \geq 5$, nous savons qu'il n'existe pas de formule explicite pour les exprimer. D'où l'utilisation de méthodes itératives.

La remarque précédente est intéressante d'un point de vue théorique, mais pas d'un point de vue pratique (sauf si $N \leq 3$, ou si on fait des calculs en arithmétique exacte). En effet, des erreurs relatives petites sur les coefficients de A entraînent en général des erreurs relatives importantes sur les racines de son polynôme caractéristique, comme on peut le constater avec la matrice diagonale $A = diag(1, 2, \ldots, N)$. Un tel algorithme est *mal conditionné*.

3.2 Méthode de la puissance

Un algorithme simple pour le calcul de valeurs propres de A est la **méthode de la puissance**

$$y_{k+1} = Ay_k \quad k = 0, 1, 2, \ldots, \tag{10}$$

où $y_0 \in \mathbb{R}^N$ (ou plus généralement, $y_0 \in \mathbb{C}^N$) est arbitraire. Le théorème suivant montre que $y_k = A^k y_0$ "tend" vers un vecteur propre de A et que le quotient de Rayleigh $y_k^\star Ay_k / y_k^\star y_k$ tend vers la valeur propre de A la plus grande en valeur absolue.

On rappelle que la notation $z^\star$ pour un vecteur ou une matrice désigne la transposée du conjugué.

Théorème 3.1 *Soit $A \in M_n(\mathbb{C})$ une matrice diagonalisable de valeurs propres $\lambda_1, \lambda_2, \ldots, \lambda_n$ et de vecteurs propres associés $v_1, \ldots, v_n$ normalisés par $\|v_i\|_2 = 1$. Si*

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \cdots \geq |\lambda_n|,$$

les vecteurs y_k donnés par (10) *vérifient*

$$y_k = \lambda_1^k (a_1 v_1 + O(|\lambda_2/\lambda_1|^k),$$

où a_1 est défini par $y_0 = \sum_i a_i v_i$. Si $a_1 \neq 0$, le quotient de Rayleigh satisfait

$$\frac{y_k^\star A y_k}{y_k^\star y_k} = \lambda_1 + O(\left|\frac{\lambda_2}{\lambda_1}\right|^k). \tag{11}$$

Si A est une matrice normale, l'erreur dans (11) *est $O(|\lambda_2/\lambda_1|^{2k})$.*

On rappelle que A est une matrice normale si $A^\star$ commute avec A, ou de manière équivalente, si A est diagonalisable dans une base orthonormale.
Remarques :
1. Les éléments de y_k croissent exponentiellement avec k, donc il est recommandé de normaliser y_k après chaque itération, i.e. de remplacer y_k par $y_k/\|y_k\|$. L'algorithme devient : choisir $y_0 \in \mathbb{C}^N$ tel que $\|y_0\|_2 = 1$ et pour $k = 0, 1, 2 \ldots$ faire

$$\begin{cases} z_k = A y_k \\ \lambda_k = y_k^\star z_k & \text{(facultatif)} \\ y_{k+1} = z_k/\|z_k\|_2 \end{cases}$$

2. Si $|\lambda_1| = |\lambda_2|$, cette méthode ne s'applique plus en général. Par contre, si A admet une unique valeur propre dont le module est maximum, on a des résultats de convergence même si A n'est pas diagonalisable.
3. Si $|\lambda_2/\lambda_1|$ est proche de 1, la convergence est lente. Pour accélérer la convergence, on utilise la modification suivante :

3.3 Méthode de la puissance inverse

Supposons qu'on connaisse une approximation μ d'une valeur propre cherchée λ_1 (pas nécessairement la plus grande en valeur absolue). L'idée est d'appliquer l'itération (10) à la matrice $(A - \mu I)^{-1}$. Les valeurs propres de cette matrice sont $(\lambda_i - \mu)^{-1}$. Si μ est proche de λ_1, et si λ_1 est simple, on a

$$\frac{1}{|\lambda_1 - \mu|} >> \frac{1}{|\lambda_i - \mu|} \text{ pour } i \geq 2.$$

La convergence va être très rapide. L'itération devient donc

$$y_{k+1} = (A - \mu I)^{-1} y_k \quad k = 0, 1, 2, \ldots . \tag{12}$$

Elle nécessite à chaque étape la résolution d'un système linéaire. Après avoir calculé la décomposition LU de $A - \mu I$, une itération de (12) ne coûte pas plus cher qu'une de (10).

Plus μ est proche de λ_1, plus la convergence est rapide. Une variante de cette méthode consiste à modifier μ au cours des itérations, de manière à accélérer encore la convergence, en posant

$$\begin{aligned} \mu_k &= y_k^\star A y_k / y_k^\star y_k, \\ y_{k+1} &= (A - \mu_k I)^{-1} y_k. \end{aligned} \tag{13}$$

3.4 Méthodes LR et QR

On rappelle que si A a toutes ses sous-matrices principales régulières, il existe une unique décomposition $A = LR$ où L est triangulaire inférieure et R triangulaire supérieure avec des 1 sur la diagonale. (NB : dans le contexte du calcul de valeurs propres, la décomposition LU se note LR pour Left et Right).

L'algorithme LR pour le calcul des valeurs propres est : soit $A_0 = A$. Pour $k = 0, 1, 2, \ldots$, faire

$$\begin{aligned} L_k R_k &= A_k \quad \text{(décomposition LR)}, \\ A_{k+1} &= R_k L_k. \end{aligned}$$

Comme $R_k A_k R_k^{-1} = A_{k+1}$, les matrices A_k et A_{k+1} sont semblables, et donc A et A_k sont semblables. Cet algorithme converge vers une matrice triangulaire supérieure, sous certaines hypothèses sur A.

L'algorithme QR est similaire : on remplace la décomposition LR par la décomposition QR : si A est régulière, il existe une décomposition $A = QR$ où Q est orthogonale et R est triangulaire supérieure.

3.5 Exercices mathématiques

1. Montrer que l'algorithme de la puissance avec renormalisation à chaque étape converge (sous les mêmes hypothèses que dans le Théorème du cours).
2. Soit A la matrice diagonale définie par

$$A = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}.$$

Le vecteur initial est $y_0 = \begin{pmatrix} a_0 \\ b_0 \end{pmatrix}$. Montrer que si $0 < a_0 < b_0$, l'algorithme de la puissance inverse (13) vérifie $\mu_k \to 1$. De même, montrer que si $0 < b_0 < a_0$, $\mu_k \to 2$.
3. Démontrer le théorème de décomposition QR. Quelle condition mettre pour assurer existence et unicité ?

3.6 Exercices de programmation

1. Tester sur un cas-test la méthode de la puissance et de la puissance inverse. On pourra essayer par exemple les matrices suivantes, et expliquer ce qui se passe.

$$A_1 = \begin{pmatrix} 10 & 6 \\ -18 & -11 \end{pmatrix} \quad A_2 = \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix} \quad A_3 = \begin{pmatrix} 7 & 4 \\ -12 & -7 \end{pmatrix}$$

2. Pour la méthode de la puissance, sur un des cas-test précédents et à l'aide d'un graphique de l'erreur, mettre en évidence la vitesse de convergence démontrée dans le Théorème 3.1. Comparer avec un cas où A est normale.
3. Examiner en Maple ou Scilab les itérés de la matrice

$$A = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix},$$

pour la méthode QR et la méthode LR. Les valeurs propres de A sont :

$$\lambda_i = 2 - 2\cos(\frac{i\pi}{4}) \quad i = 1, 2, 3, \text{ i.e.}$$

$$0.5857864376, \quad 2, \quad 3.4142135624.$$

Que se passe-t-il si on remplace A par une matrice orthogonale ?

4 Méthodes Itératives - Equations non linéaires

En pratique, on est souvent confront à la résolution d'un système d'équations non linéaires, i.e. pour une fonction $f : \mathbb{R}^n \to \mathbb{R}^n$ donnée, on cherche un point $x \in \mathbb{R}^n$ tel que

$$f(x) = 0. \tag{14}$$

Par exemple, pour résoudre numériquement $x' = F(x)$ par la méthode d'Euler implicite :

$$y_{n+1} = y_n + hF(y_{n+1}),$$

qui définit implicitement y_{n+1} en fonction de y_n. Autre exemple, en optimisation, si $g : \mathbb{R}^n \to \mathbb{R}$ est une fonction différentiable de plusieurs variables, et si a est un extréma local de g, alors $g'(a) = 0$ dans $\mathbb{R}^n$, et le problème d'optimisation peut se ramener à une équation non linéaire.

Problèmes d'existence et d'unicité pour (14) (penser à $f(x) = \mathrm{e}^x$, $f(x) = \sin x$) : le théorème des fonctions implicites donne l'unicité locale.

En général, il n'y a pas d'algorithme fini pour trouver une solution de (14). On est donc obligé d'utiliser des méthodes itératives.

4.1 Méthode de point fixe (ou des approximations successives)

On considére le problème du calcul d'un point fixe de $\Phi : \mathbb{R}^n \to \mathbb{R}^n$, i.e. on cherche $x \in \mathbb{R}^n$ tel que

$$x = \Phi(x). \tag{15}$$

Les problèmes (14) et (15) sont équivalents et il y a beaucoup de possibilités pour écrire (14) sous la forme (15). Par exemple, écrire $\Phi(x) = x - f(x)$ ou $\Phi(x) = x - B \cdot f(x)$ où B est une matrice inversible bien choisie.

L'algorithme du point fixe est

$$\begin{cases} \underline{\text{Choisir}}\ x_0 \in \mathbb{R}^n, \\ x_{k+1} = \Phi(x_k), \qquad k \geq 0. \end{cases} \tag{16}$$

Si la suite (x_k) converge vers a, et si Φ est continue en a, alors a est une solution de (15). Concernant l'erreur, en supposant $\Phi \in \mathcal{C}^1(\mathbb{R}^n)$, on peut écrire

$$x_{k+1} - a = \Phi(x_k) - \Phi(a) = \Phi'(a) \cdot (x_k - a) + \|x_k - a\|\varepsilon(x_k), \tag{17}$$

où $\varepsilon : \mathbb{R}^n \to \mathbb{R}^n$ est une fonction qui tend vers 0 en a et $\Phi'(a)$ est la différentielle de Φ en a. Comme dans le cas linéaire, le comportement de l'erreur $x_k - a$ est donné par le rayon spectral de $\Phi'(a)$.

Heuristiquement, si $\Phi'(a)$ possède une valeur propre $|\lambda_1| > 1$, la composante de e_k dans la direction du vecteur propre va être agrandie : l'itération ne converge pas vers a. Cet argument peut être rendu rigoureux.

Inversement, si toutes les valeurs propres de $\Phi'(a)$ satisfont $|\lambda_i| < 1$, on peut choisir une norme dans $\mathbb{R}^n$ telle que pour la norme matricielle correspondant, $\|\Phi'(a)\| < 1$. Ceci implique que pour $\|x_k - a\|$ suffisamment petit, on a $\|x_{k+1} - a\| \leq \alpha\|x_k - a\|$, où $\alpha \in [\|\Phi'(a)\|, 1[$. En d'autres termes,

Théorème 4.1 *Soient $\Phi \in \mathcal{C}^1(\Omega, \mathbb{R}^n)$ où Ω est un ouvert de $\mathbb{R}^n$, et $a \in \Omega$ tel que $a = \Phi(a)$. Si le rayon spectral $\rho(\Phi'(a))$ de $\Phi'(a)$ est < 1, il existe une boule ouverte $B(a, r)$ centrée en a telle que pour tout $x_0 \in B(a, r)$, la suite définie par* (16) *converge vers a et de plus, la convergence est géométrique (ou linéaire), i.e. il existe $0 < \alpha < 1$ et une norme $\|\cdot\|$ sur $\mathbb{R}^n$ tels que*

$$\|x_{k+1} - a\| \leq \alpha\|x_k - a\| \qquad \forall k \in \mathbb{N}.$$

Application : méthode de point fixe pour la résolution d'un système linéaire $Ax = b$ (pour des grandes tailles de A). On écrit

$$A = M - N, \qquad Mx_{k+1} = Nx_k + b.$$

Dans la méthode de Jacobi, on prend $M=$ diagonale de A, et $N = M - A$. Dans la méthode de Gauss-Seidel, on prend $M =$ partie triangulaire inférieure, diagonale comprise, et $N = M - A=$ opposé de la partie triangulaire supérieure.

Remarque dans le cas critique où $\rho(\Phi'(a)) = 1$, les deux cas sont possibles, convergence ou divergence. Examiner par exemple les cas $\Phi(x) = \sin(x)$ en $a = 0$ puis $\Phi(x) = x + x^3$ en $a = 0$.

Critère d'arrêt pour la programmation

En pratique, il faut arrêter l'algorithme à une itération k assez grande pour que l'erreur soit petite. On choisit typiquement $\|x_{k+1} - x_k\| \leq tol$ où tol est une tolérance fixée à l'avance. Remarquer que si Φ est β contractante dans la boule $B(a, r)$ $(0 < \beta < 1)$ on a $\|x_k - a\| \leq 1/(1-\beta)^{-1}\|x_{k+1} - x_k\|$, d'où une majoration de l'erreur commise, si on a une majoration de β.

Notion de vitesse de convergence, d'ordre

Soit $(x_k)_{k\in\mathbb{N}}$ une suite réelle (ou complexe) convergeant vers $a \in \mathbb{R}$ (ou $a \in \mathbb{C}$), avec $x_k \neq a$ pour k assez grand. On dit que $(x_k)_k$ *converge linéairement vers* a si

$$\lim_{k\to\infty} \frac{|x_{k+1} - a|}{|x_k - a|} = \mu,$$

avec $0 < \mu < 1$. Si la limite ci-dessus existe avec $\mu = 0$, on dit que $(x_k)_k$ *converge super-linéairement vers* a, et si $\mu = 1$, la convergence est dite *sous-linéaire*. Pour préciser la notion de convergence super-linéaire, on dit que la suite x_k de limite a est *convergente d'ordre* $q > 1$ si

$$\lim_{k\to\infty} \frac{|x_{k+1} - a|}{|x_k - a|^q} = \nu,$$

pour un $\nu > 0$. En particulier, la convergence d'ordre 2 est dite *quadratique*.

D'après (17), pour une suite récurrente $x_{k+1} = \Phi(x_k)$ convergent vers a, on a $\mu = |\Phi'(a)|$. De même, en supposant Φ de classe C^2, si $\Phi'(a) = 0$ et $\Phi''(a) \neq 0$, la convergence est quadratique avec $\nu = |\Phi''(a)|/2$.

4.2 Méthode de Newton

Considérons maintenant le problème de la résolution de (14), où la fonction $f : \Omega \subset \mathbb{R}^n \to \mathbb{R}^n$ est supposée être au moins une fois différentiable. Si $x_0 \in \mathbb{R}^n$ est une approximation de la solution cherchée, on linéarise $f(x)$ autour de x_0

$$f(x) \approx f(x_0) + f'(x_0)(x - x_0)$$

et on calcule le zéro de cette linéarisation. Si l'on répète cette procédure avec la nouvelle approximation, on obtient l'algorithme de Newton

$$\begin{cases} \underline{\text{Choisir}}\ x_0 \in \mathbb{R}^n \\ x_{k+1} = x_k - f'(x_k)^{-1} f(x_k), \quad k \geq 0. \end{cases} \tag{18}$$

Remarquer que $f'(x_k)$ est la différentielle de f et que cet algorithme nécessite en dimension n la résolution d'un système *linéaire*. Sous certaines conditions, la suite (x_k) converge *localement* vers une solution de $f(x) = 0$, dans le sens suivant :

Théorème 4.2 *Supposons que f soit de classe $\mathcal{C}^2$ sur un ouvert $\Omega \subset \mathbb{R}^n$. Soit $a \in \mathbb{R}^n$ une solution de $f(x) = 0$ telle que $f'(a)$ soit inversible. Alors il existe une boule ouverte $B(a,r) \subset \Omega$ telle que $f'(x)$ soit inversible pour tout $x \in B(a,r)$ et telle que pour tout $x_0 \in B(a,r)$, la suite définie par* (18) *soit contenue dans $B(a,r)$ et converge vers a.*

Si de plus f est de classe C^3, alors la convergence est quadratique, i.e. il existe une constante $C > 0$ telle que

$$||x_{k+1} - a|| \leq C||x_k - a||^2 \quad \forall k \in \mathbb{N}.$$

Remarques

1. L'algorithme de Newton est un cas particulier d'algorithme de point fixe.
2. La convergence quadratique signifie que le nombre de chiffres significatifs double à chaque itération !
3. En pratique, on calcule $f'(x)$ par une dérivation numérique : Newton modifié. On peut aussi remplacer $f'(x_k)$ par $f'(x_0)$, qu'on calcule une fois pour toute.
4. Critères d'arrêt
5. Newton appliqué à un polynôme : il peut y avoir convergence même si la racine est multiple (cf exercices).

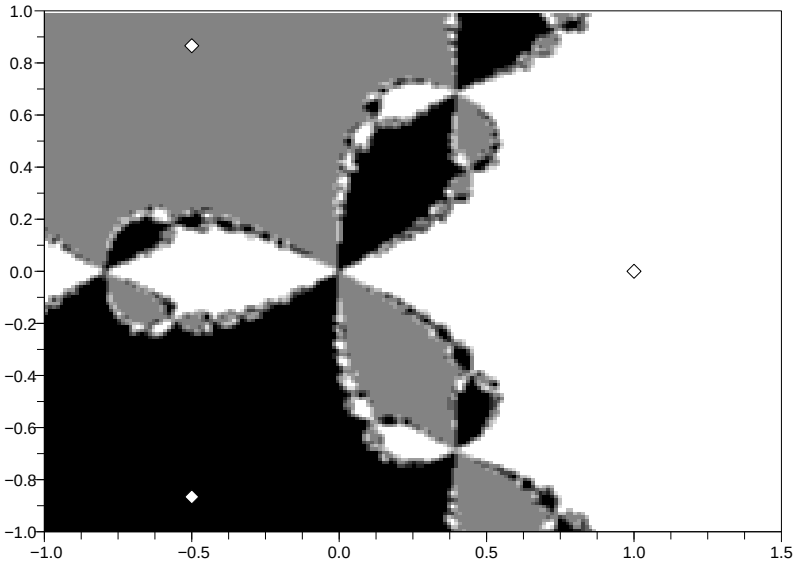

FIG. 1 – Bassins d'attraction pour l'itération (19)

6. La convergence est locale, i.e. si on démarre proche d'une solution. Concernant la convergence globale on sait très peu de choses. On observe souvent des cas de divergence et on ne sait analyser que quelques cas de fonctions simples. Un exemple (pris dans [Hai]) est

$$f(z) = z^3 - 1 \text{ avec } z = x + iy \in \mathbb{C} \iff f(x,y) = (x^3 - 3xy^2 - 1, 3x^2y - y^3),$$

pour lequel l'itération devient

$$z_{k+1} = z_k - \frac{z_k^3 - 1}{3z_k^2} = \frac{1}{3}(2z_k + \frac{1}{z_k}). \tag{19}$$

La figure 1 montre les ensembles

$$A(a) = \{z_0 \in \mathbb{C} \mid \{z_k\}_k \text{ converge vers } a\},$$

appelés *bassins d'attraction*, et calculés par ordinateur, pour les trois racines a de $f(z) = 0$. Les z_0 du domaine blanc donnent une suite convergeant vers $a = 1$, ceux du domaine gris vers $a = (-1 + i\sqrt{3})/2$ et ceux du domaine noir vers $a = (-1 - i\sqrt{3})/2$. On observe que proche de la racine, on converge vers la racine, mais que plus loin la solution ne converge pas nécessairement vers la racine la plus proche.

4.3 Exercices mathématiques

1. Démontrer les théorèmes 4.1 et 4.2.
2. Donner un exemple de suite récurrente convergeant sous-linéairement, puis un exemple de suite récurrente convergente d'ordre $q = 3/2$.
3. Etudier la convergence de la suite logistique $x_0 \in [0, 1]$, $x_{k+1} = m(1 - x_k)x_k$, pour $m = 4$, puis pour $m \in [1, 4]$. On montrera en particulier la convergence pour $m \in [1, 3]$.
4. Que devient l'algorithme de Newton pour la résolution du système linéaire $Ax = b$?
5. Quelle est la vitesse de convergence de l'algorithme de Newton pour le calcul de la racine du polynôme x^n ?
6. Montrer que si P est un polynôme à coefficients réels n'admettant que des racines réelles, dont on note β la plus grande racine, et si $x_0 > \beta$, alors la suite définie par (18) est décroissante et converge vers β.
7. Préciser la question précédente en montrant que si β est de multiplicité $n \geq 2$,

$$\lim_{k \to +\infty} \frac{|x_{k+1} - \beta|}{|x_k - \beta|} = 1 - \frac{1}{n}.$$

8. Donner une majoration de la valeur absolue des racines d'un polynôme en fonction de ses coefficients (en vue d'appliquer le résultat ci-dessus).

4.4 Exercices de programmation

1. Programmer une fonction $NewtonRacineCarree(a, x0, tol)$ calculant la racine carrée d'un réel a avec l'algorithme de Newton ; $x0$ est le point de départ et tol la tolérance Examiner l'influence de $x0$ sur la convergence, ainsi que le doublement du nombre de décimales à chaque itération (?Digits)
2. Programmer une méthode $PointFixeRacineCarre(a, x0, tol)$ calculant la racine carrée de a par un point fixe (différent de la méthode de Newton).
3. Pour un cas-test dont vous connaissez la solution, tracer sur un graphique l'erreur en fonction du nombre d'itérations pour la méthode du point fixe. Retrouver l'ordre de convergence.
4. Modifier la procédure ci-dessus pour tracer sur le même graphique l'erreur en fonction du nombre d'itération pour la méthode de Newton et pour deux autres méthodes de point fixe. Comparer les ordres de convergence.
4. En Maple ou Scilab, programmer une fonction $Newton(f, x_0, kmax)$ calculant la suite x_k donnée par (18) pour la fonction d'une variable réelle f.
5. Tester la fonction $Newton$ pour f égale au polynôme unitaire dont les racines sont 1 (simple), 3 (double) et -1 (triple). Tracer sur un même schéma, pour chaque racine de f, l'erreur en fonction des itérations k. Que remarque-t-on ?
6. Programmer l'algorithme de Newton (modifié) pour une application $f : \mathbb{R}^2 \to \mathbb{R}^2$. Tester la fonction sur le calcul des points critiques de $g(x, y) = (x^2 + x)(y^2 + 4y)$, puis au minimum de $g(x, y) = x^2y^2$.

4.5 Références

Pour les références, l'application de l'algorithme de Newton aux polynômes en MAPLE est faite dans [Fer]. On pourra consulter [Pom] pour l'algorithme de Newton appliqué à un polynôme.

5 Méthodes à un pas pour la résolution numérique des équations différentielles

5.1 Introduction

On souhaite résoudre numériquement le problème de Cauchy

$$\begin{cases} X'(t) = F(t, X(t)) \quad t \in [t_0, t_0 + T], \\ X(t_0) = X^0, \end{cases} \tag{20}$$

où $F : [t_0, t_0 + T] \times \mathbb{R}^d \to \mathbb{R}^d$ est (au moins) continue et $X^0 \in \mathbb{R}^d$.

Pour résoudre numériquement ce problème on se donne une subdivision régulière $t_0 < t_1 < \ldots < t_N = t_0 + T$ en posant $t_n = t_0 + nh$ où $h = T/N$ est le pas. Une méthode numérique **explicite à un pas** s'écrit

$$\begin{cases} X_{n+1} = X_n + h\phi(t_n, X_n, h) \\ X_0 = X^0 \end{cases} \tag{21}$$

où $\phi : [t_0, t_0 + T] \times \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}$ est (au moins) continue (ϕ s'obtient en général à partir de F et de formules d'intégration numérique).

Il faut connaître le schéma d'**Euler explicite** :

$$X_{n+1} = X_n + hF(t_n, X_n).$$

Citons la méthode du point milieu, obtenue à partir de la formule de quadrature du point milieu :

$$\begin{cases} X_{n+1} = X_n + hF(t_{n+\frac{1}{2}}, X_{n+\frac{1}{2}}), \\ \text{où } t_{n+\frac{1}{2}} = t_n + \frac{h}{2}, \quad X_{n+\frac{1}{2}} = X_n + \frac{h}{2}F(t_n, X_n). \end{cases}$$

La célèbre formule de **Runge-Kutta d'ordre 4**, plus précise, s'obtient à partir de la formule de quadrature de Simpson :

$$\begin{cases} X_{n+1} = X_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4), \\ \text{où } k_1 = F(t_n, X_n),\ k_2 = F(t_n + h/2, X_n + hk_1/2), \\ k_3 = F(t_n + h/2, X_n + hk_2/2),\ k_4 = F(t_n + h, X_n + hk_3). \end{cases}$$

A connaître également : le schéma d'**Euler implicite**

$$X_{n+1} = X_n + hF(t_n, X_{n+1}),$$

très utilisé dans certains cas (équation de la chaleur) pour ses propriétés de stabilité. Ici, pour calculer X_{n+1} à partir de X_n, il faut résoudre un système d'équations non linéaires. Il s'agit d'une méthode à un pas mais, car X_{n+1} se calcule à partir de X_n et t_n, mais implicite : la question de l'existence et l'unicité de X_{n+1} se pose notamment (on aura typiquement une condition de petitesse sur h pour assurer cela).

5.2 Hypothèses et théorème de Cauchy-Lipschitz global

Pour simplifier l'analyse numérique, on fait les hypothèses suivantes sur F :

$$F \in \mathcal{C}^0([t_0, t_0 + T] \times \mathbb{R}^d, \mathbb{R}^d), \tag{22}$$

et F est globalement L-lipschitzienne par rapport à la deuxième variable ($L > 0$), i.e.

$$\forall t \in [t_0, t_0 + T],\ \forall y, z \in \mathbb{R}^d, \quad \|F(t, y) - F(t, z)\| \leq L\|y - z\|. \tag{23}$$

Ic $\|\cdot\|$ désigne une norme quelconque sur $\mathbb{R}^d$. On prendra en général la norme ∞. On rappelle le fameux

Théorème 5.1 (Cauchy-Lipschitz) *Sous les hypothèses* (22)(23)*, pour tout* $X^0 \in \mathbb{R}^d$*, il existe une unique application* $X \in \mathcal{C}^1([t_0, t_0+T], \mathbb{R}^d)$ *solution du problème* (20)*. De plus,* X *dépend continûment de* X^0 *dans le sens suivant : si* $X^{0,p} \to X^0$ *dans* $\mathbb{R}^d$*, alors* X^p*, la solution de condition initiale* $X^{0,p}$*, converge uniformément vers* X *sur* $[t_0, t_0+T]$*.*

Preuve de l'existence et unicité par théorème du point fixe (écrire le système (20) sous forme intégrale). Cette preuve est valable si on remplace $\mathbb{R}^d$ par un espace vectoriel complet E. Stabilité par Gronwall (redonne aussi l'unicité).

5.3 Définitions et résultats

Les questions principales en analyse numérique sont celle de la *convergence du schéma*, et de la *vitesse de convergence* : la solution approchée est-elle proche de la solution exacte ? Et si oui, avec quelle erreur ? Pour illustrer ces questions, on considère le problème de Cauchy

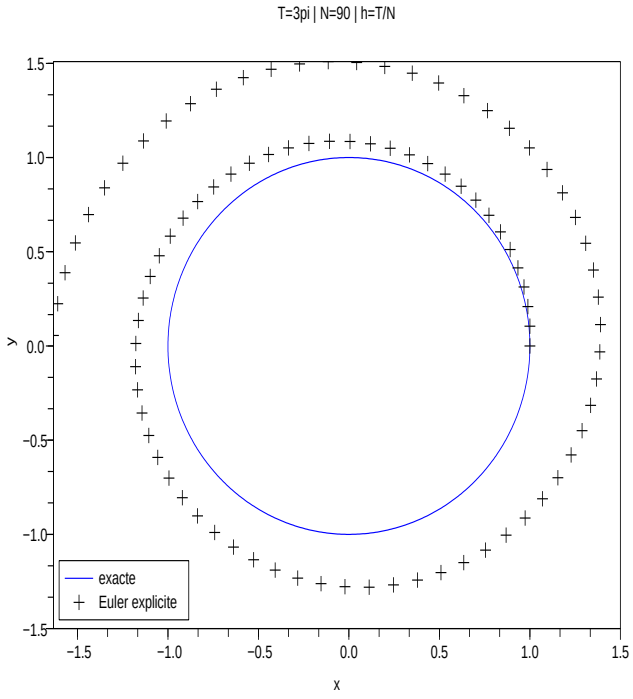

FIG. 2 – Euler explicite pour (24)

$$\begin{cases} x'(t) = -y(t), & t \in [0,T], \\ y'(t) = x(t), & t \in [0,T], \\ x(0) = 1,\ y(0) = 0, \end{cases} \tag{24}$$

dont la solution exacte est $(x(t), y(t)) = (\cos t, \sin t)$. La Figure 2 donne l'approximation de cette solution trouvée par le schéma d'Euler explicite appliqué à (24) pour $h = T/N = 3\pi/90 \approx 0,104$. On voit que pour des petits temps, l'approximation est proche de la solution exacte, tandis pour des temps plus grands, l'écart grandit de plus en plus, de sorte qu'au lieu d'un cercle, nous obtenons une spirale divergente.

La Figure 3 montre l'analogue pour le schéma d'Euler implicite : au lieu d'une spirale divergente, on obtient une spirale convergente.

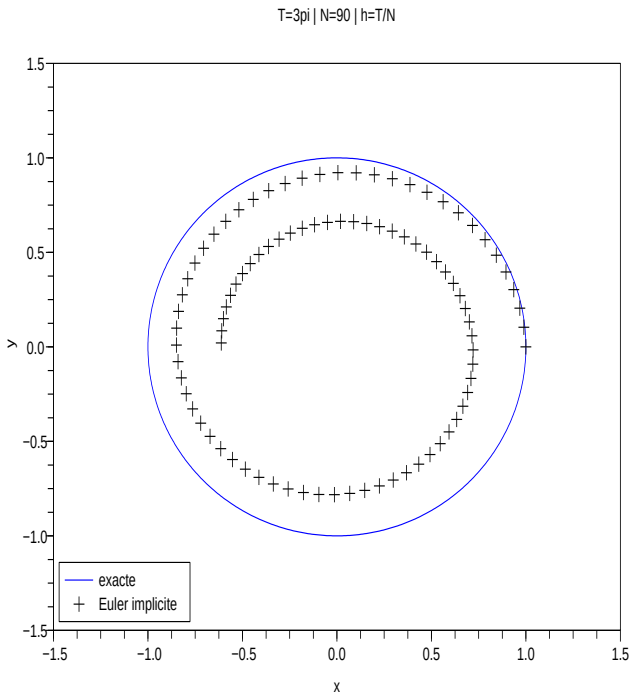

FIG. 3 – Euler implicite pour (24)

Pour répondre à la question de la convergence, on introduit deux notions : la *consistance*, et la *stabilité*. Pour la question de la vitesse de convergence, on précise la notion de consistance en introduisant la notion *d'ordre*.

Consistance

Définition 5.2 (Erreur de consistance) *C'est l'erreur avec laquelle la solution exacte satisfait le schéma sur un pas de temps. Pour un schéma explicite à un pas donné par* (21)*, cette erreur est*

$$e(t,h) := X(t+h) - [X(t) + h\phi(t, X(t), h)],$$

où X est la solution exacte de (20)*.*

Remarque : pour le schéma d'Euler implicite, on utilisera la formule

$$e(t,h) := X(t+h) - [X(t) + hF(t+h, X(t+h))].$$

Définition 5.3 (Ordre) *Un schéma est d'***ordre au moins** $p \geq 1$ *si*

$$e(t,h) = O(h^{p+1})$$

pour toute F de classe $\mathcal{C}^p$ et pour toute solution exacte X (la constante dans "O" dépend de F et X, mais pas de h, et est uniforme en t car t reste dans un compact). Le schéma est d'ordre exactement p s'il est d'ordre au moins p et n'est pas d'ordre au moins $p+1$.

Trouver l'ordre revient à effectuer un développement de Taylor. Par exemple, les schémas d'Euler explicite et implicite sont d'ordre 1.

Définition 5.4 (Consistance) *Un schéma est* ***consistant*** *si, pour toute solution exacte X, la somme des erreurs de consistance $\sum_{n=0}^{N-1} ||e(t_n, h)||$ tend vers 0 quand N tend vers $+\infty$ (ou, ce qui est équivalent, quand h tend vers 0).*

On rappelle qu'on travaille sur $[t_0, t_0 + T]$, on a posé $h = T/N$ et $t_n = nh$, donc la somme ci-dessus est une fonction de N. Cette somme dépend également (comme l'erreur de consistance) de la solution exacte X.

On a le critère de consistance suivant :

Théorème 5.5 *Le schéma à un pas* (21) *est consistant si et seulement si*

$$\Phi(t, z, 0) = f(t, z), \quad \forall (t, z) \in [t_0, t_0 + T] \times \mathbb{R},$$

avec Φ continue sur $[t_0, t_0 + T] \times \mathbb{R} \times [0, h^\star]$.

On a clairement la

Proposition 5.6 *Si $p \geq 1$, un schéma d'ordre au moins p est consistant.*

En fait, on peut montrer que si f et Φ sont de classe C^1, un schéma est consistant si et seulement s'il est consistant d'ordre au moins $p = 1$ (exercice).

Stabilité

Définition 5.7 (Stabilité) *Un schéma* (21) *est* ***stable*** *s'il existe une constante $S \geq 0$ (appelée constante de stabilité) telle que pour tout $N \in \mathbb{N}^*$ assez grand, pour toute suite finie $(\varepsilon_n)_{0 \leq n < N}$ d'éléments de $\mathbb{R}^d$, pour tout $X_0 \in \mathbb{R}^d$, $\tilde{X}_0 \in \mathbb{R}^d$, les suites $(X_n)_{0 \leq n \leq N}, (\tilde{X}_n)_{0 \leq n \leq N}$ définies par*

$$\begin{array}{ll} X_{n+1} = X_n + h\phi(t_n, X_n, h) & 0 \leq n < N \\ \tilde{X}_{n+1} = \tilde{X}_n + h\phi(t_n, \tilde{X}_n, h) + \varepsilon_n & 0 \leq n < N, \end{array}$$

satisfont

$$\max_{0 \leq n \leq N} ||\tilde{X}_n - X_n|| \leq S \left(||\tilde{X}_0 - X_0|| + \sum_{0 \leq n < N} ||\varepsilon_n|| \right).$$

Philosophie : une petite erreur initiale $||\tilde{X}_0 - X_0||$ et des erreurs d'arrondis ε_n provoquent une erreur finale contrôlable.

L'outil principal pour démontrer la stabilité est le

Lemme 5.8 (Gronwall discret) *Soient $N \in \mathbb{N}^*$, $h > 0$ et $(\theta_n)_{0 \leq n \leq N}$, $(\alpha_n)_{0 \leq n \leq N}$ deux suites de réels positifs telles que*

$$\forall n \in \{0, \ldots, N-1\}, \quad \theta_{n+1} \leq (1 + \Lambda h)\theta_n + \alpha_n,$$

pour un certain $\Lambda > 0$. Alors

$$\forall n \in \{0, \ldots, N\}, \quad \theta_n \leq e^{\Lambda n h}\theta_0 + \sum_{k=0}^{n-1} e^{\Lambda(n-k-1)}\alpha_k.$$

Sa démontration est immédiate par récurrence (utiliser $1 + \Lambda h \leq e^{\Lambda h}$). On montre ainsi que le schéma d'Euler explicite est stable, de constante de stabilité e^{LT}. On a le critére de stabilité suivant :

Proposition 5.9 *Si Φ est globalement lipschitzienne par rapport à la deuxième variable, i.e. s'il existe $M > 0$ telle que*

$$\forall (t,h) \in [t_0, t_0+T] \times [0, h^\star],\ \forall z \in \mathbb{R},\ \forall \bar{z} \in \mathbb{R}, \quad |\Phi(t,z,h) - \Phi(t,\bar{z},h)| \leq M|z-\bar{z}|,$$

alors le schéma est stable.

En pratique, si on suppose F globalement lipschitzienne par rapport à la deuxième variable, comme Φ est en général construit à partir de F, la stabilité devient facile à vérifier avec ce critère.

Convergence

Définition 5.10 *Soient $X^0 \in \mathbb{R}^d$ et $X \in \mathcal{C}^1([t_0, t_0+T], \mathbb{R}^d)$ la solution de (20). Le schéma (21) est **convergent** pour X si pour tout $X_0 \in \mathbb{R}^d$ et pour tout $N \in \mathbb{N}^*$, la suite $(X_n)_{0\leq n\leq N}$ définie par (21) vérifie*

$$\lim_{N\to\infty,\ X_0 \to X^0} \max_{0\leq n\leq N} ||X_n - X(t_n)|| \to 0,$$

où $(X_n)_{0\leq n\leq N}$ désigne la suite définie par (21).

La quantité $\max_{0\leq n\leq N} ||X_n - X(t_n)||$ est l'**erreur de convergence**. On a le

Théorème 5.11 *Si un schéma est consistant et stable, alors il est convergent.*

Preuve : on pose $\varepsilon_n = e(t_n, X(t_n))$= erreur de consistance au temps t_n. Alors la suite $(\tilde{X}_n)_{0\leq n\leq N}$ dans la définition de la stabilité est $\tilde{X}_n = X(t_n)$, i.e. la solution exacte au temps t_n. La définition de la stabilité donne alors une majoration de l'erreur de convergence.

La même démonstration implique

Théorème 5.12 *Si un schéma est consistant d'ordre $p \geq 1$ et stable, alors il est convergent avec une erreur en $O(h^p)$, i.e. il existe $C > 0$ tel que*

$$\max_{0\leq n\leq N} ||X_n - X(t_n)|| \leq Ch^p.$$

La rérérence obligée concernant les notions ci-dessus est [Dem91]. Pour aller plus loin dans l'analyse numérique, consulter le [CM84].

Quelques considérations pratiques

Si on a un schéma convergent, il reste des questions à résoudre dans la pratique :

1. Comment choisir le pas h "assez petit" pour être raisonnablement proche de notre solution : il peut y avoir des conditions de stabilité (cf. exercice 4 ci-dessous)

2. Cas des problèmes mal conditionnés (ou problèmes raides) : considèrer

$$x'(t) = 3x(t) - 3t,\ t \geq 0 \text{ et } x(0) = \alpha.$$

La solution exacte est donnée par $x(t) = (\alpha - 1/3)e^{3t} + t + 1/3$. Si on cherche à calculer la valeur de x pour $\alpha = 1/3$ au temps $t = 10$, on trouve $x(10) = 10 + 1/3$. Par contre, si on fait le calcul avec 0.333333 au lieu de 1/3, on trouve $x(10) = -(1/3)10^{-6} \cdot e^{30} + 31/3$, d'où une erreur de l'ordre de 10^7 (erreur relative de 10^6). Une erreur relative de 10^{-6} sur les données donne une erreur relative d'ordre d'ordre 10^6 sur les solutions.

NB : ce problème ne se produit pas si on considère l'équation $x'(t) = -3x - 3t$, de solution exacte $x(t) = (\alpha - 1/3)e^{-3t} - t + 1/3$. On dira qu'un problème est bien conditionné si la constante de stabilité n'est pas trop grande.

5.4 Exercices mathématiques

1. Montrer que si F et Φ sont de classe C^1, alors un schéma est consistant si et seulement s'il est consistant d'ordre au moins 1.
2. Montrer que le schéma d'Euler implicite est consistant d'ordre 1, stable, et convergent. On pourra adapter la définition de la stabilité.
3. Montrer que pour h assez petit, le schéma d'Euler implicite donne toujours une unique solution X_{n+1} en fonction de X_n et t_n. Puis, montrer qu'il s'agit d'un schéma à un pas d'ordre un et stable.
4. Expliciter la suite $(x_n)_{n\in\mathbb{N}}$ obtenue en appliquant le schéma d'Euler explicite de pas $h > 0$ au problème

$$\begin{cases} x'(t) = -ax(t), \quad t \geq 0, \\ x(0) = x_0, \end{cases}$$

où $a > 0$. Montrer que $x_n \to 0$ quand $n \to \infty$ si et seulement si $h < 2/a$. Reprendre l'étude avec Euler implicite. On montrera dans ce cas que $x_n \to 0$ pour toute valeur de $h > 0$. Reprendre l'étude quand $a = A$ est une matrice symétrique définie positive (diagonaliser A pour se ramener au cas précédent).
5. On considère le système différentiel suivant sur $[0, +\infty[$:

$$\begin{cases} x'(t) = -y(t), \\ y'(t) = x(t). \end{cases} \tag{25}$$

(a) Déterminer la solution exacte pour $x_0, y_0 \in \mathbb{R}^2$. On pourra poser $z(t) = x(t) + iy(t)$. Montrer que $|z(t)|$ est constant.
(b) Déterminer la suite $(x_n, y_n)_{n\in\mathbb{N}}$ obtenue en appliquant le schéma d'Euler explicite à pas constant $h > 0$. Montrer que $|z_n| \to \infty$ quand $n \to \infty$, où on a posé $z_n = x_n + y_n$.
(c) Même question avec Euler implicite. On montrera que $|z_n| \to 0$ quand $n \to \infty$.
(d) Essayer de construire un schéma pour (25) qui conserve la quantité $x_n^2 + y_n^2$, afin de reproduire au mieux l'équation différentielle.
6. Résoudre explicitement les équations différentielles suivantes. Dans chaque cas, vérifier si les hypothèses faites en cours sont satisfaites, et préciser l'intervalle (ou les intervalles) de définition des solutions.

$$i)y' + 4y = 1 \qquad ii)y' - \frac{x+1}{x^2}y = 0 \qquad iii)y' - xy = x$$
$$iv)xy' - y = x\sqrt{1 + \ln x} \qquad v)y' \sin x - y \cos x + 1 = 0 \qquad vi)[(1 + x^2)y]' = xy$$
$$vii)y' = y^2 \qquad viii)y' = xy - xy^2 \qquad ix)y' = \sin y \qquad (1 + x^2)^2 y' + 2x + 2xy^2 = 0$$

7. Même exercice que ci-dessus avec les équations différentielles du second ordre suivantes. Dans chaque cas on écrira de plus le système différentiel d'ordre un équivalent.

$$i)y'' - 3y' + 2y = 0 \qquad ii)y'' + 4y = 0.$$

8. On considère la méthode du point milieu, donnée par $x_0 = x^0$ et

$$x_{n+1} = x_n + hf(t_{n+1/2}, x_{n+1/2}), \text{ où}$$

$$t_{n+1/2} = t_n + h/2 \text{ et } x_{n+1/2} = x_n + \frac{h}{2} f(t_n, x_n),$$

pour $0 \leq n < N$. Expliquer comment cette méthode s'obtient à partir de la formule de quadrature du point milieu. Montrer qu'il s'agit d'une méthode à un pas, puis déterminer son ordre.

Montrer que la méthode est stable et convergente. On donnera une estimation de l'erreur de convergence.
9. Reprendre les questions ci-dessus avec la méthode de Heun donnée par $x_0 = x^0$ et

$$x_{n+1} = x_n + \frac{h}{2} f(t_n, x_n) + \frac{h}{2} f(t_n + h, x_n + hf(t_n, x_n)) \text{ pour } 0 \leq n < N.$$

10. On considére la méthode à un pas définie par

$$\begin{cases} x_{n+1} = x_n + hf(t_n, x_n) + h^\alpha, & n \geq 0 \\ x_0 = x_0. \end{cases}$$

Dire si les assertions suivantes sont vraies ou fausses : i) Il s'agit d'une méthode à un pas
ii) La méthode est consistante pour tout α. iii) La méthode est stable pour tout α. iv) La méthode est convergente si $\alpha > 1$.

5.5 Exercices de programmation

1. Résoudre le problème de Cauchy $x'(t) = tx(t)$, $x(0) = 1$ sur l'intervalle $[0, T]$. Pour $T = 2$, tracer sur une même figure la solution exacte et son approximation donnée par le schéma d'Euler explicite (à pas constant $h = T/N$). Représentez ensuite sur la figure la solution obtenue par un solveur ODE intégré à SCILAB (HELP ODE).
2. Pour le problème de Cauchy de l'exercice précédent, faire un autre programme pour tracer l'erreur de convergence en fonction de N pour la méthode d'Euler. Retrouvez l'ordre de la méthode d'Euler.
3. Modifiez le programme de l'exercice précédent pour rajouter sur la figure l'erreur de convergence pour la méthode du point milieu et la méthode de Runge-Kutta. Retrouvez les ordres de ces méthodes.
4. On considère (25) sur $[0, 2\pi]$, avec la condition initiale $(x_0, y_0) = (1, 0)$. Programmer en Maple (ou Scilab) le schéma d'Euler explicite pour ce problème de Cauchy. On tracera sur un même dessin la *trajectoire* de la solution obtenue et de la solution exacte. Essayer plusieurs pas. Tracez également une solution obtenue par un solveur intégré.

6 Etude qualitative de systèmes différentiels autonomes

6.1 Introduction

Un système différentiel *autonome* est de la forme

$$X'(t) = F(X(t)) \tag{26}$$

où F est une fonction d'un ouvert Ω de $\mathbb{R}^d$, à valeurs dans $\mathbb{R}^d$ que l'on supposera localement lipschitzienne, ce qui signifie que F est lipschitzienne sur tout compact de Ω. Une *solution* de (26) est un *couple* (I, X) où I est un intervalle de $\mathbb{R}$, $X \in \mathcal{C}^1(I, \mathbb{R}^d)$, $X(t) \in \Omega$ pour tout $t \in I$, et X vérifie (26) pour tout $t \in I$. Parce que le système est autonome, on a la propriété suivante :

Proposition 6.1 *Si (I, X) est une solution de (26), alors pour tout $t_1 \in \mathbb{R}$, $t \mapsto X(t_1 + t)$ est également une solution de (26) sur l'intervalle $I - t_1 = \{t \in \mathbb{R} : t_1 + t \in I\}$.*

En termes géométriques, $F : \Omega \to \mathbb{R}^d$ est un *champ de vecteurs*. Les solutions (I, X) de (26) sont appelées *courbes intégrales*. On appelle *trajectoires* du système (26) les courbes de $\mathbb{R}^d$ paramétrées par $I : t \mapsto X(t)$, où (I, X) est une solution. La proposition ci-dessus traduit l'invariance par les translations $(t, X) \mapsto (t + t_1, X)$ de l'ensemble des courbes intégrales.

Rappelons qu'une solution est *maximale* si elle n'admet aucun prolongement autre qu'elle-même. Le système vérifie les hypothèses du théorème de Cauchy-Lipschitz, donc

Théorème 6.2 *On suppose que $F : \Omega \to \mathbb{R}^d$ est localement lipschitzienne. Pour tout $(t_0, X^0) \in \mathbb{R} \times \Omega$, il existe une unique solution maximale (I, X) de (26) telle que $t_0 \in I$ et $X(t_0) = X^0$. De plus, $I =]T_-, T_+[$ avec $-\infty \leq T_- < T_+ \leq +\infty$, et*

(i) si $T_+ < +\infty$, $X(t)$ sort définitivement de tout compact contenu dans Ω lorsque $t \to T^+$;

(ii) de même, si $T_- > -\infty$, $X(t)$ sort définitivement de tout compact contenu dans Ω lorsque $t \to T^+$.

Remarques

0. Comme le système est autonome, il suffit d'étudier le cas $t_0 = 0$ pour comprendre le cas général ; c'est le choix fait en général.

1. En pratique, on a souvent $F \in C^1(\Omega, \mathbb{R}^d)$, ce qui garantit automatiquement que F est localement lipschitzienne.

2. Si $\Omega = \mathbb{R}^d$, ce qui arrive souvent, les conclusions (i) et (ii) deviennent :

(i) si $T_+ < +\infty$, $\lim_{t \to T_+} \|X(t)\| = +\infty$;

(ii) si $T_- > -\infty$, $\lim_{t \to T_-} \|X(t)\| = +\infty$.

3. L'unicité implique que les trajectoires de deux solutions maximales soit sont égales, soit ne se croisent jamais.

Exemples

En dimension 1, un système autonome est une ode de la forme $x'(t) = f(x(t))$. Il s'agit d'une ode à variables séparables. En choisissant $f(x) = x^2$, on voit qu'une solution n'est pas forcément globale. En choisissant $f(x) = |x|^\alpha$ avec $\alpha \in (0, 1)$ et $x_0 = 0$, on voit que sans l'hypothèse "localement lipschitzienne", l'unicité est en défaut.

En dimension 2, un système autonome s'écrit

$$\begin{cases} \dfrac{dx}{dt} = a(x, y) \\ \dfrac{dy}{dt} = b(x, y) \end{cases} \tag{27}$$

Toute solution (x, y) de (27) sur un intervalle I où $a(x, y) \neq 0$ vérifie

$$\frac{dy}{dx} = \frac{b(x, y)}{a(x, y)} = f(x, y).$$

Résoudre cette EDO permet de trouver les trajectoires (mais pas les courbes intégrales).

En dimension d, on peut donner l'exemple de la ligne de plus grande pente d'une fonction $J \in \mathcal{C}^2(\mathbb{R}^d, \mathbb{R})$, donnée par

$$X'(t) = -\nabla J(X(t)), \quad t \geq 0. \tag{28}$$

Ce système différentiel est liée à l' algorithme du gradient à pas fixe (l'algorithme de gradient à pas fixe est le schéma d'Euler explicite appliqué à ce système différentiel !). Le système différentiel (28) estun système gradient, cas particulier de système (différentiel) autonome.

6.2 Points d'équilibre et linéarisation

On appelle *point d'équilibre* du système (26) un point $X_0 \in \mathbb{R}^d$ tel que $F(X_0) = 0$. Remarquer que $X_0 \in \mathbb{R}^d$ est un point d'équilibre du système si et seulement si la fonction constante $\mathbb{R} : t \mapsto X_0$ est une courbe intégrale.

En linéarisant près de $Y_0 \in \mathbb{R}^d$, et en supposant $F \in C^1(\Omega, \mathbb{R}^d)$, on a

$$F(Y) = F(Y_0) + DF_{Y_0}(Y - Y_0) + g_{Y_0}(Y),$$

avec $g_{Y_0}(Y) = o(\|Y - Y_0\|)$, g continue, de sorte que si $Y_0 = X_0$ est un point d'équilibre,

$$X'(t) = F(X(t)) \iff X'(t) = DF_{X_0}(X(t) - X_0) + g_{X_0}(X(t).$$

En négligeant le terme en g, et en posant $\tilde{X}(t) = X(t) - X_0$, on obtient le système linéaire

$$\tilde{X}'(t) = DF_{X_0}\tilde{X}(t),$$

appelé **système linéarisé** en X_0 de (27).

Il est raisonnable d'espérer que le comportement du système (27) est proche de celui du système linéarisé. En fait cela est vrai sous certaines conditions, mais faux en général. Le théorème qui décrit ce rapport est énoncé (mais pas démontré) dans Zuily-Quéffelec (Théorème de linéarisation IV.4 p 382). Il s'agit d'un théorème difficile, et dont la démonstration ne sera pas exigée (mais qu'il peut être utile de connaître).

La méthode de linéarisation permet également de comprendre la stabilité des points d'équilibres. Le résultat qui suit et sa démonstration sont au programme.

Définition 6.3 *Un point d'équilibre $X_0 \in \mathbb{R}^d$ du système* (26) *est dit* **stable** *s'il existe $\delta > 0$ et $C \geq 0$ tels que pour toute solution maximale X vérifiant $\|X(0) - X_0\| \leq r$,*

1. *X est définie pour tout $t \geq 0$;*
2. *pour tout $t \geq 0$, $\|X(t) - X_0\| \leq C\|X(0) - X_0\|$.*

En d'autres termes, X_0 est stable si toute solution qui au temps initial est assez proche de X_0 reste proche de X_0 pour tous les temps ultérieurs. Remarquer que les définitions de stabilité varient légèrement selon le contexte et selon les auteurs.

Un point d'équilibre est dit **instable** s'il n'est pas stable.

Définition 6.4 *Un point d'équilibre $X_0 \in \mathbb{R}^d$ du système* (26) *est dit* **asymptotiquement stable** *s'il existe $\delta > 0$ et une fonction continue $C : [0, +\infty[\to [0, +\infty[$ avec $\lim_{t \to +\infty} C(t) = 0$ tels que pour toute solution maximale X vérifiant $\|X(0) - X_0\| \leq \delta$,*

1. *X est définie pour tout $t \geq 0$;*
2. *pour tout $t \geq 0$, $\|X(t) - X_0\| \leq C(t)\|X(0) - X_0\|$.*

Il est utile de connaître ce qui se passe en dimension 2 sur un système différentiel linéaire $X' = AX$, $A \in M_2(\mathbb{R})$ (cf. feuille).

Théorème 6.5 *Soit X_0 un point d'équilibre de (26), où $F \in C^1(\Omega, \mathbb{R}^d)$, et $A = DF_{X_0}$ la matrice du système linéarisé. Si toutes les valeurs propres (λ_k) de A ont des parties réelles strictement négatives, X_0 est asymptotiquement stable. De plus, on peut prendre $C(t) = e^{-\mu t}$ dans la fonction de stabilité, avec $\mu \in]0, -\max(Re(\lambda_k)[$.*

Ce théorème s'applique en particulier au cas d'un système linéaire (cf. exemples ci-dessus).

6.3 Plan d'étude en dimension 2

Pour étudier un système différentiel autonome de dimension 2, cas particulier important, i.e.

$$\begin{cases} x'(t) = f(x, y) \\ y'(t) = g(x, y), \end{cases}$$

une méthode générale est la suivante (cf. [ZQ95] p.383) :
0. Déterminer quel théorème de Cauchy-Lipschitz appliquer (régularité de F, ouvert Ω) ;
1. Examiner les symétries du système ;
2. Calculer les points d'équilibre du système et donner leur nature (noeud, col,...) ;
3. Tracer les courbes $\{(x, y) \in \mathbb{R}^2 : f(x, y) = 0\}$ et $\{(x, y) \in \mathbb{R}^2 : g(x, y) = 0\}$, appelées *isoclines* ;
4. Régionner le plan suivant le sens du champ ($f > 0$ $g > 0$, $f > 0$ $g < 0$ ... ;
5. Tenter un dessin ;
6. Affiner le protocole selon d'autres informations disponibles.

Exemple

Voir [ZQ95] p. 422 pour l'étude de

$$\begin{cases} \dfrac{dx}{dt} = x - xy - x^2 \\ \dfrac{dy}{dt} = -4y + 2xy. \end{cases}$$

Une bonne référence pour les notions ci-dessus est [Dem91] p.147, 273 et suivantes. Consulter également [ZQ95].

6.4 Exercices mathématiques

1. Lire le chapitre "Méthodes de résolution explicite" du Demailly. Refaire les équations proposées en exemple, puis les problèmes du chapitre.
2. Etudier le système du pendule pesant :

$$\begin{cases} x' = y \\ y' = -\sin x \end{cases}$$

L'allure du système s'appelle le *portrait de phase* en mécanique, parce que x représente la position et $y = x'$ la vitesse. Pour l'étude théorique, on adoptera le schéma ci-dessus.
3. Etudier de même le système du pendule pesant avec frottement :

$$\begin{cases} x' = y, \\ y' = -\alpha y - \sin x, \end{cases}$$

avec $\alpha > 0$.

Pour la correction de ces deux exercices, on pourra consulter [Dum] p. 235 et suivantes.

6.5 Exercices de programmation

Exercice 1 On considère le système différentiel suivant :

$$\begin{cases} x'(t) = y(t) - x(t), \\ y'(t) = x(t)y(t) - 1. \end{cases} \tag{29}$$

1. (Math) Etudier ce système différentiel. Déterminer en particulier les points fixes (également appelés points stationnaires, ou points singuliers). Etudiez le système linéarisé au voisinage de chacun des points fixes. Déterminer également l'isocline horizontale (resp. verticale), i.e. l'ensemble des points où le champ de vecteurs est horizontal (resp. vertical).
2. (Programmation) Tracer le champ de vecteurs associé à ce système dans le plan de phase (x, y). On suggère de le faire dans la fenêtre $[-3, 3] \times [-3, 3]$ (en Scilab, fonction `fchamp`, et en Maple, fonction `dfieldplot`).
3. Sur le dessin précédent, rajouter le tracé des isoclines verticales et horizontales.
4. Tracer le portrait de phase (en Maple, `phaseportrait`). En Scilab, commande `ode`.
4bis. (Question subsidiaire en Scilab) : à l'aide de la commande `xclick`, écire un script qui trace sur le dessin de la question 2 la trajectoire passant par le point du dessin sur lequel vous cliquez avec le bouton gauche de la souris.
5. Le comportement observé au voisinage de chacun des points singuliers correspond-il à ce qui est attendu de l'étude du linéarisé ? On pourra tracer le champ de vecteur du linéarisé, pour chacun des points d'équilibre.

Exercice 2 Même questions avec le système différentiel suivant :

$$\begin{cases} x' = x - xy - x^2, \\ y' = -4y + 2xy. \end{cases}$$

Exercice 3 On considère le système différentiel linéaire

$$\begin{cases} x' = ax + by, \\ y' = cx + dy, \end{cases}$$

où

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

est un matrice réelle non singulière. En particulier, ce système admet un unique point fixe $(0, 0)$. Décrivez les différents types de comportement asymptotique possibles au voisinage de ce point fixe.

7 Optimisation

7.1 Problématique et vocabulaire

Un problème d'optimisation, après un travail de modélisation mathématique, met en évidence des variables d'état ou paramètres, des contraintes sur ces paramètres, et un critère à optimiser. Il se présente sous la forme suivante : trouver $\bar{x} \in C$ tel que

$$f(\bar{x}) = \inf_{x \in C} f(x). \tag{30}$$

La fonction $f : X \to \mathbb{R}$ est le critère, ou coût, ou fonction-objectif, ou fonction-coût. X est l'espace de paramètres (typiquement, $\mathbb{R}^n$, plus généralement un espace vectoriel normé) ; la partie C de X est l'ensemble des contraintes.

Les premières questions que l'on se pose sont les suivantes :

- existence de $\bar{x}$: tout d'abord, f est-elle bornée inférieurement sur C (i.e. a-t-on $f(x) \geq C$ pour une constante $C \in \mathbb{R}$) ? Si oui, la borne inférieure est-elle atteinte ?

Si on trouve $\bar{x} \in C$ solution de (30), on dira que

– $\bar{x}$ est un minimiseur (global) de f sur C ;

– $\bar{x}$ est un minimum de f sur C ;

– f atteint en $\bar{x}$ son minimum sur C ;

NB : en optimisation, on peut vouloir maximiser ; pour passer d'un problème de maximisation à un problème de minimisation, on change f en $-f$. Cela marche bien pour certaines propriétés (continuité, différentiabilité), mais pas pour d'autres (convexité).

Vocabulaire : on dit que $\bar{x} \in C$ est un *minimiseur local de f sur C* s'il existe un voisinage V de $\bar{x}$ tel que

$$f(\bar{x}) \leq f(x) \text{ pour tout } x \in C \cap V. \tag{31}$$

De même, on dira que $\bar{x} \in C$ est un *minimiseur local strict de f sur C* s'il existe un voisinage V de $\bar{x}$ tel que

$$f(\bar{x}) < f(x) \text{ pour tout } x \in C \cap V,\ x \neq \bar{x}. \tag{32}$$

Exemples en dimension 1 :

$f(x) = x^2$ un unique minimiseur global ; pas de maximum.

$f(x) = (x^2 - 1)^2$. f admet deux minimiseurs globaux sur $\mathbb{R}$, à savoir ± 1 (qui sont aussi des minimiseurs locaux).

$f(x) = x^4/4 - x^3/3 - x^2$. Alors $f'(x) = x^3 - x^2 - 2x = x(x+1)(x-2)$. Les points critique de f sont -1, 0 et $+2$. Le signe de f' et les valeurs $f(-1) = -1/4$, $f(0) = 0$, $f(2) = -4$ montrent que 2 est un minimiseur global (strict), -1 est un minimiseur local (strict) et 0 un maximiseur local (strict).

$f(x) = x^3 - x$. f n'a pas de minimiseur global sur $\mathbb{R}$. Par contre, f admet un minimiseur local. Idem avec maximiseur (pas de global, 1 local). Notons également un point critique qui n'est ni minimiseur, ni maximiseur.

D'autres questions :

- unicité du minimiseur ?

- conditions nécessaires d'optimalité : du style "si $\bar{x}$ est un minimum local de f sur C, alors P"

- conditions suffisantes d'optimalité : du style, "si P, alors $\bar{x}$ est un minimiseur (local) de f sur C.

- algorithme : procédure permettant de localiser une solution et de l'approcher. Le concepteur d'algorithmes crée une série de règles de construction d'une suite $(x_k)_k$ dont il espère prouver une des conditions suivantes :

- $x_k \to \bar{x}$ une solution de (30) (assez rare)

- $f(x_{k+1}) < f(x_k)$ pour tout k ;

- $f(x_k)$ décroit vers $\bar{f} = \inf_{x \in C} f(x)$ (i.e. (x_k) est une suite minimisante)

- (x_k) est bornée et toute limite de sous-suite convergente vérifie une condition nécessaire de minimalité ;

- il existe une sous-suite de (x_k) dont la limite vérifie une condition nécessaire de minimalité.

Autres aspects importants : vitesse de convergence, complexité des algorithmes.

- aspects qualitatifs : transformation de problème (problème dual) ; sensibilité aux perturbations, robustesse.

7.2 Exemples fondamentaux

Système linéaire

Soit A une matrice rélle de taille N symétrique définie positive et $b \in \mathbb{R}^N$. La solution du système linéaire $Ax = b$ est donnée par le minimiseur suivant

$$\inf_{x \in \mathbb{R}^N} f(x), \quad \text{avec } f(x) = \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle, \tag{33}$$

où $\langle \cdot, \cdot \rangle$ désigne le produit scalaire dans $\mathbb{R}^N$.

Méthode des moindres carrés

Soit $g : \mathbb{R}^N \to \mathbb{R}^M$. Tout solution (s'il en existe) de $g(x) = 0$ est une solution du problème de minimisation

$$\inf_{x \in \mathbb{R}^N} f(x), \text{ avec } f(x) = \|g(x)\|^2. \tag{34}$$

On peut choisir différentes normes sur $\mathbb{R}^M$, mais pour des raisons de différentiabilité, pour les moindres carrés, on choisit la norme euclidienne. Plus précisément, si $g(x) = Bx - c$, avec $B \in M_{M,N}(\mathbb{R})$ et $c \in \mathbb{R}^M$, on appelle solution du système linéaire $Bx = c$ au sens des moindres carrés tout minimiseur de

$$\inf_{x \in \mathbb{R}^N} \|Bx - c\|. \tag{35}$$

Recherche de valeur propre

Soit A une matrice symétrique de taille N. La plus petite valeur propre de A est donnée par

$$\lambda_1 = \inf_{x \in \mathbb{R}^N, \|x\|=1} \langle Ax, x\rangle. \tag{36}$$

Un exemple en gestion

Une entreprise fabrique deux biens, ce qui nécessite l'utilisation de deux ateliers dont les capacités de production exprimées en heures d'usinage sont de 12. Chaque unité du permier bien demande 2 heures d'usinage dans le premier atelier et 1 heure dans le second. Chaque unité du deuxième bien demande 1 heures d'usinage dans le premier atelier et 2 heures dans le second. On veut déterminer les productions x_1 et x_2 des deux biens qui maximisent la marge sur coût variable de l'entreprise si les marges unitaires de deux biens sont de 4 et 3.

Les contraintes de production s'écrivent

$$\begin{aligned} 2x_1 + x_2 &\leq 12 \\ x_1 + 2x_2 &\leq 12 \\ x_1 \geq 0,\ x_2 &\geq 0 \end{aligned}$$

Elles définissent un polygone convexe (appelé "ensemble de production" en microéconomie), de sommets $(0,0)$, $(4,4)$, $(6,0)$ et $(0,6)$.

NB : la marge unitaire, c'est le prix de vente unitaire moins les coûts variables unitaires. La marge sur coût variable (le "bénéfice") est $4x_1 + 3x_2$. On pourrait rajouter les coûts fixes, mais c'est le même problème.

Le problème s'écrit donc

$$\begin{cases} \max 4x_1 + 3x_2 \\ 2x_1 + x_2 \leq 12 \\ x_1 + 2x_2 \leq 12 \\ x_1 \geq 0,\ x_2 \geq 0 \end{cases}$$

Il s'agit d'un problème de minimisation sous contraintes d'inégalité. Comme la fonction-coût et les contraintes sont définies par des fonctions affines (linéaires, par abus de langage), il s'agit d'un *programme linéaire.*

7.3 Une classification (non-exhaustive) des problèmes d'optimisation

- Programmation linéaire : optimisation à données linéaires

$$\inf_{x \in C} f(x), \text{ avec } f(x) = \langle c, x \rangle \text{ et } C := \{x \mid Ax \leq b\}.$$

(NB : C polyèdre fermé, pas nécessairement borné).

Optimisation linéaire-quadratique si

$$f(x) = \frac{1}{2}\langle Ax, x \rangle - \langle b, x \rangle,$$

toujours avec C polyèdre fermé.

- Optimisation convexe : f convexe et C convexe.

- Optimisation différentiable (ou lisse) : toutes les données du problèmes sont des fonctions différentiables (i.e. de classe C^1 au moins).

- Optimisation non-différentiable : cas où certaines données ne sont pas différentiables ; important en pratique ($f(x) = \max(f_1(x), \ldots, f_m(x))$, norme non euclidienne...

- Optimisation en dimension infinie : calcul des variations, contrôle.

- Optimisation combinatoire

NB : Algorithmes génétiques...

7.4 Résultats d'existence et d'unicité

On notera $\langle \cdot, \cdot \rangle$ le produit scalaire usuel dans $\mathbb{R}^n$ et $\| \cdot \|$ la norme associée, i.e.

$$\langle x, y \rangle = \sum_{i=1}^{n} x_i y_i \text{ et } \|x\| = (\sum_{i=1}^{n} x_i^2)^{1/2}.$$

On rappelle qu'une fonction continue sur un compact atteint ses bornes. C'est un résultat fondamental d'existence de minmiseur (dû essentiellement à Weierstrass). On a plus généralement le théorème d'existence suivant :

Théorème 7.1 (Existence) *Soit $f : C \to \mathbb{R}$ avec C fermé non vide de $\mathbb{R}^n$. On suppose que*

(i) f est continue sur C ;

(ii) f est coercive sur C, i.e.

$$\lim_{\|x\| \to +\infty, x \in C} f(x) = +\infty.$$

(pas de condition (ii) à vérifier si C est borné).

Alors f est bornée inférieurement sur C et il existe $\bar{x} \in C$ tel que $f(\bar{x}) = \inf_{x \in C} f(x)$.

Remarque 7.2 *1. On peut remplacer $\mathbb{R}^n$ par tout espace vectoriel de dimension finie.*
2. L'hypothèse (i) peut être affaiblie en "f est semi-continue inférieurement".

Une condition suffisante d'unicité est la stricte convexité de f. On rappelle tout d'abord les définitions suivantes : $C \subset \mathbb{R}^n$ est convexe si pour tout $(x, x') \in C^2$ et pour tout $t \in]0,1[$, $tx + (1-t)x' \in C$ (i.e., le segment $[x, x']$ appartient à C).

Soit C un convexe de $\mathbb{R}^n$. $f : C \to \mathbb{R}$ est dite convexe sur C si pour tout $(x, x') \in C \times C$ et pour tout $t \in]0,1[$ on a

$$f(tx + (1-t)x') \leq tf(x) + (1-t)f(x'). \tag{37}$$

f est dite *strictement convexe* sur C quand l'inégalité (37) est stricte dès que $x \neq x'$, i.e.

$$f(tx + (1-t)x') < tf(x) + (1-t)f(x'). \tag{38}$$

On a

Théorème 7.3 (Unicité) *Si C est convexe et si $f : C \to \mathbb{R}$ est strictement convexe sur C, alors il existe au plus un $\bar{x}$ minimisant f sur C.*

7.5 Rappels sur le calcul différentiel et la convexité

On rappelle qu'une fonction $g : U \subset \mathbb{R}^m \to \mathbb{R}^n$ définie sur un ouvert U de $\mathbb{R}^m$ est différentiable en $x \in U$ s'il existe $l : \mathbb{R}^m \to \mathbb{R}^n$ linéaire telle que

$$f(x+h) = f(x) + l(h) + \|h\|\epsilon(h), \text{ avec } \epsilon(h) \to 0 \text{ quand } h \to 0.$$

On note $l = df(x)$ la différentielle de f en x qui est unique. Elle s'identifie à une matrice à m lignes et n colonnes appelée *matrice jacobienne* de f en x, dont le terme (i,j) est $\frac{\partial f_i(x)}{\partial x_j}$.

Lorsque $n = 1$, l est une forme linéaire, que l'on peut représenter par le vecteur

$$\nabla f(x) = (\partial_{x_1} f(x), \ldots, \partial_{x_m} f(x)).$$

N.B. : la fonction f est différentiable en x ssi chacune des fonctions coordonnée f_i est différentiable en x.

On dit que $f : U \to \mathbb{R}$ est deux fois différentiable lorsque $df(x)$ définie sur U à valeurs dans $M_{m,n}(\mathbb{R})$ est différentiable. La matrice jacobienne de ∇f est appelée *hessienne* de f en x : il s'agit d'une matrice symétrique dont le terme (i,j) est $\frac{\partial^2 f(x)}{\partial x_i \partial x_j}$.

Si f est deux fois différentiable, on a la formule de Taylor-Young d'ordre deux :

$$f(x+h) = f(x) + df(x)(h) + \frac{1}{2} d^2 f(x)(h,h) + \|h\|^2 \epsilon(h),$$

avec $\epsilon(h) \to 0$ lorsque $h \to 0$, $h \neq 0$. On peut l'écrire également

$$f(x+h) = f(x) + \langle \nabla f(x), h \rangle + \frac{1}{2} \langle \nabla^2 f(x) h, h \rangle + \|h\|^2 \epsilon(h).$$

On a rappelé ci-dessus la définition d'une fonction convexe et strictement convexe.

Définition 7.4 *Soit $C \subset \mathbb{R}^n$ un convexe non vide. f est dite α-convexe sur C (avec $\alpha > 0$) si pour tout $(x, x') \in C \times C$, pour tout $t \in]0,1[$,*

$$f(tx + (1-t)x') \leq tf(x) + (1-t)f(x') - \frac{\alpha}{2} t(1-t)\|x - x'\|^2. \tag{39}$$

On dit également fortement convexe, de module de forte convexité α.

On a la

Proposition 7.5 *Soit $f : U \to \mathbb{R}$ différentiable sur un ouvert U de $\mathbb{R}^n$ et C un convexe de U. Alors*

(i) f est convexe sur C ssi

$$f(x) \geq f(x') + \langle \nabla f(x'), x - x' \rangle \quad \forall (x, x') \in C \times C. \tag{40}$$

(ii) f est strictement convexe sur C ssi l'inégalité (40) est stricte dès que $x \neq x'$.

(iii) f est α-convexe ssi

$$f(x) \geq f(x') + \langle \nabla f(x'), x - x' \rangle + \frac{1}{2}\alpha \|x - x'\|^2, \quad \forall (x, x') \in C \times C. \tag{41}$$

Preuve. Commençons par (iii) : on suppose d'abord f α-convexe. Soient $(x, x') \in C^2$. Alors, pour tout $t \in [0, 1]$,

$$f(tx + (1-t)x') \leq tf(x) + (1-t)f(x') - \frac{\alpha}{2}t(1-t)\|x - x'\|^2.$$

En retranchant $f(x')$ des deux côtés, en divisant par $t > 0$, et en faisant $t \to 0^+$, on trouve

$$\langle \nabla f(x'), x - x' \rangle \leq f(x) - f(x') - \frac{\alpha}{2}\|x - x'\|^2,$$

ce qui est le résultat attendu.

Réciproquement, supposons que f vérifie (41). On va montrer d'abord que

$$\langle \nabla f(x) - \nabla f(x'), x - x' \rangle \geq \alpha \|x - x'\|^2, \quad \forall (x, x') \in C^2. \tag{42}$$

Cela se trouve en échangeant les rôles de x et x' dans (41) et en additionnant les deux inégalités obtenues.

Maintenant, soient $(x, x') \in C^2$ et posons $\varphi(t) = f(tx + (1-t)x')$. Alors φ est différentiable sur $[0, 1]$ et

$$\varphi'(t) = \langle \nabla f(tx + (1-t)x'), x - x' \rangle,$$

de sorte que

$$\varphi'(t) - \varphi'(s) = \langle \nabla f(tx + (1-t)x') - \nabla f(sx + (1-s)x'), x - x' \rangle.$$

Si on pose $\tilde{x} = tx + (1-t)x$ et $\tilde{x}' = sx + (1-s)x'$, on a $\tilde{x} - \tilde{x}' = (t-s)(x - x')$, de sorte que d'après (42),

$$(\varphi'(t) - \varphi'(s))(t - s) \geq \alpha(t-s)^2 \|x - x'\|^2, \quad \forall s, t \in [0, 1],$$

ou encore,

$$\varphi'(t) - \varphi'(s) \geq \alpha(t-s)\|x - x'\|^2 \text{ si } 0 \leq s \leq t \leq 1.$$

Soit $\theta \in]0, 1[$. En intégrant cette dernière inégalité de $t = \theta$ à 1 (NB : si $\varphi \in C^1$, on peut intégrer directement ; si φ est seulement différentiable, on calcule la dérivée de la diffénce ci-dessous par rapport à t, on constate qu'elle est ≥ 0, puis on utilise le TAF) puis de $s = 0$ à θ (même remarque) on obtient d'abord

$$\varphi(1) - \varphi(\theta) - (1-\theta)\varphi'(s) \geq \alpha[\frac{(1-s)^2}{2} - \frac{(\theta - s)^2}{2}]\|x - x'\|^2,$$

puis

$$\theta[\varphi(1) - \varphi(\theta)] - (1-\theta)[\varphi(\theta) - \varphi(0)] \geq \alpha \left[\frac{-(1-s)^3}{6} + \frac{(\theta - s)^3}{6}\right]_0^\theta \|x - x'\|^2,$$

i.e.

$$\theta\varphi(1) + (1-\theta)\varphi(0) - \varphi(\theta) \geq \frac{\theta(1-\theta)}{2}\|x - x'\|^2,$$

ce qui est l'égalité demandée.

Pour (i), même démonstration avec $\alpha = 0$. Même chose pour strict (on peut le refaire), au moins le sens (40) strict implique f strictement convexe. Dans l'autre sens, si f est strictement convexe et qu'il existe $x \neq x' \in C$ tels que (40) est une égalité, i.e.

$$f(x) = f(x') + \langle \nabla f(x'), x - x' \rangle,$$

alors on voit sur un dessin que f est affine sur $[x, x']$, une contradiction. Cela se montre avec

$$f(tx + (1-t)x) \geq f(x') + \langle \nabla f(x'), t(x - x') \rangle = tf(x) + (1-t)f(x').$$

■

De même, lorsque l'on peut différentier deux fois, on a :

Proposition 7.6 *Soit f deux fois différentiable sur un ouvert convexe U de $\mathbb{R}^n$. Alors*
(i) f est convexe sur U ssi $\nabla^2 f(x)$ est semi-définie positive pour tout $x \in U$.
(ii) Si $\nabla^2 f(x)$ est définie positive pour tout $x \in U$, alors f est strictement convexe sur U ;
(iii) f est α-convexe sur U ssi la plus petite valeur propre de $\nabla^2 f(x)$ est minorée par α, i.e.

$$\langle \nabla^2 f(x)d, d \rangle \geq \alpha\|d\|^2, \quad \forall x \in U, \quad \forall d \in \mathbb{R}^n.$$

Preuve. Montrons (iii). Supposons f α-convexe. Soit $x \in U$, $d \in \mathbb{R}^n$ et $t \in \mathbb{R}$ assez petit. On pose $x' = x + td$ dans (42) et on fait tendre t vers 0. On obtient le résultat. Dans l'autre sens, soient $(x, x') \in U^2$, et on pose

$$\psi(t) = \langle \nabla f(x) - \nabla f(tx + (1-t)x'), x - x' \rangle.$$

On a

$$\psi'(t) = \langle \nabla^2 f(tx + (1-t)x)(x - x'), x - x' \rangle \geq \alpha\|x - x'\|^2.$$

En intégrant, comme $\psi(0) = 0$, on a

$$\psi(t) \geq t\alpha\|x - x'\|^2,$$

c'est-à-dire (42). On a vu que cela impliquait que f est α-convexe. ■

Exemple : $f \in C^2(\mathbb{R}, \mathbb{R})$ est convexe ssi f' est croissante ssi $f'' \geq 0$. f est α-convexe ssi $f''(x) \geq \alpha$ pour tout x. Si $f'' > 0$, alors f est strictement convexe.

7.6 Minimisation sans contraintes

Théorème 7.7 (Conditions de minimalité du premier ordre) *Soit U un ouvert de $\mathbb{R}^n$ et $f : U \to \mathbb{R}$. Si $\bar{x} \in U$ est un minimum local de f et si f est différentiable en $\bar{x}$, alors $\nabla f(\bar{x}) = 0$.*

Les points $\bar{x} \in U$ vérifiant $\nabla f(\bar{x}) = 0$ s'appellent les points critiques (ou stationnaires) de f. Les valeurs prises par f en des points critiques s'appellent les valeurs critiques de f. Remarquons qu'un point critique n'est pas nécessairement un minimum local. On a une réciproque sous l'hypothèse de convexité.

Preuve. Soit $r > 0$ tel que $B(\bar{x}, r) \subset U$ et $\bar{x}$ minimise f sur $B(\bar{x}, r)$. Soit $h \in \mathbb{R}^n$ et $t \in \mathbb{R}$. Alors, si $|t| < r/\|h\|$, on a $\|th\| < r$ et on peut écrire

$$f(\bar{x} + th) = f(\bar{x}) + \langle \nabla f(\bar{x}), th \rangle + \|th\|\epsilon(th).$$

Comme $f(\bar{x} + h) \geq f(\bar{x})$,

$$t\langle \nabla f(\bar{x}), h \rangle + |t|\|h\|\epsilon(th) \geq 0.$$

En divisant par $t > 0$ et en faisant $t \to 0^+$, on trouve $\langle f(\bar{x}), h \rangle \geq 0$. De même, en divisant par $t < 0$ et en faisant $t \to 0^-$, on trouve $\langle f(\bar{x}), h \rangle \leq 0$. D'où le résultat. ■

On a une réciproque, sous hypothèse de convexité :

Théorème 7.8 ("Réciproque") *Soit $f : U \to \mathbb{R}$ convexe et différentiable , où U est un ouvert convexe de $\mathbb{R}^n$. Alors, pour $\bar{x} \in U$, les conditions suivantes sont équivalentes :*
(i) $\bar{x}$ est un minimiseur (global) de f sur U ;
(ii) $\bar{x}$ est un minimiseur local de f sur U ;
(iii) $\bar{x}$ est un point critique de f sur U, i.e. $\nabla f(\bar{x}) = 0$.

Preuve. (i) $\Rightarrow$ (ii) : vrai pour toute fonction.
(ii) $\Rightarrow$ (iii) : Théorème ci-dessus.
(iii) $\Rightarrow$ (i) : on utilise (40) avec $x' = \bar{x}$. La preuve est complète. ■

En l'absence de convexité, on montrera ci-dessous une condition suffisante de minimalité locale.

Théorème 7.9 (Conditions nécessaires du second ordre) *Soit U un ouvert de $\mathbb{R}^n$ et $f : U \to \mathbb{R}$ deux fois différentiable sur U. Si $\bar{x} \in U$ est un minimum local de f, alors*

$$\nabla f(\bar{x}) = 0 \text{ et } \nabla^2 f(\bar{x}) \text{ est semi-définie positive.}$$

Théorème 7.10 (Conditions suffisantes) *Soit U un ouvert de $\mathbb{R}^n$ et $f : U \to \mathbb{R}$ deux fois différentiable sur U. Si $\bar{x} \in U$ vérifie*

$$\nabla f(\bar{x}) = 0 \text{ et } \nabla^2 f(\bar{x}) \text{ est définie positive,}$$

alors $\bar{x}$ est un minimiseur local strict de f.

Remarque 7.11 *Soit $\bar{x} \in U$ tel que $\nabla f(\bar{x}) = 0$.*
• *si $\nabla^2 f(x)$ n'est ni semi-définie négative, ni semi-définie positive, alors $\bar{x}$ n'est ni un maximum local, ni un minimum local : il s'agit d'un point selle, ou col (preuve par Taylor).*
• *si $\nabla f(\bar{x})$ est semi-définie positive, alors on ne peut pas conclure.*

Preuve. Dans les deux cas, on utilise la formule de Taylor à l'ordre deux.
Deux exemples instructifs

$$f : x = (x_1, x_2) \in \mathbb{R}^2 \mapsto f(x) = 2x_1^4 - 4x_1^2 x_2 + x_2.$$

Alors $(0, 0)$ est un minimum strict le long de toute droite passant par $(0, 0)$. En effet, soit $d = (d_1, d_2) \neq (0, 0)$. Alors

$$f(td_1, td_2) = t^2(3d_1^4 t^2 - 4d_1^2 d_2 t + d_2^2).$$

Soit $p(t)$ le second trinome ; si $d_2 \neq 0$, alors $p(0) = d_2^2 > 0$, et si $d_2 = 0$, alors $p(t) = 3d_1^4 t^2 > 0$ pour $t \neq 0$. Donc $f(td_1, td_2) > 0$ pour $t \neq 0$, t proche de 0.

Cependant, $(0, 0)$ n'est pas un minimum local de f, car

$$f(x_1, x_2) = (x_2 - 3x_1^3)(x_2 - x_1^2).$$

Soit $f : \mathbb{R}^2 \to \mathbb{R}$ définie par

$$f(x_1, x_2) = 2x_1^3 + 3e^{2x_2} - 6x_1 e^{x_2}.$$

Alors $\bar{x} = (1, 0)$ est le seul point critique de f ; $\nabla^2 f(\bar{x})$ est définie positive donc $(0, 0)$ est un minimum local strict de$f$. Cependant, f n'est pas bornée inférieurement ni supérieurement : ainsi, f a un minimum local strict, qui est l'unique point critique de f, et qui n'est pas un minimum global.

7.7 Algorithme de gradient

Soit $f : U \to \mathbb{R}$ de classe C^1, où U est un ouvert de $\mathbb{R}^n$. On veut calculer un minimiseur de f dans U i.e. une solution $\bar{x}$ du problème

$$f(\bar{x}) = \inf_{x \in U} f(x). \tag{43}$$

Un moyen de calculer $\bar{x}$ est donc de résoudre $\nabla f(\bar{x}) = 0$. Pour cela on peut utiliser par exemple l'algorithme de Newton ou un algorithme de point fixe. Mais ces algorithmes ne garantissent pas que l'on converge vers un minimiseur. Pour exploiter que $\bar{x}$ est un minimiseur de f, une idée est de construire des algorithmes de descente, dans lesquels f décroît à chaque étape.

Rappelons que $\nabla f(x)$ indique la direction et le sens dans lequel f croît le plus vite (au premier ordre) au voisinage de x. En effet, d'après le dvt de Taylor, pour $x \in \mathbb{R}^n$, $v \in \mathbb{R}^n$ avec $\|v\| = 1$, et $t \in \mathbb{R}$,

$$f(x + tv) = f(x) + t\langle \nabla f(x), v\rangle + t\varepsilon(t),$$

où $\varepsilon(t) \to 0$ dans $\mathbb{R}^n$ quand $t \to 0$. D'autre part, l'inégalité de Cauchy-Schwarz dit que

$$|\langle \nabla f(x), v\rangle \leq \|\nabla f(x)\| \quad \forall v \in \mathbb{R}^n,\ \|v\| = 1,$$

et qu'il y a égalité ssi $v = \pm \dfrac{\nabla f(x)}{\|\nabla f(x)\|}$ (pour $\nabla f(x) \neq 0$). Donc la meilleure direction v pour faire décroître f le plus vite sur la droite $x + tv$ est (au premier ordre en t), le choix $v = -\dfrac{\nabla f(x)}{\|\nabla f(x)\|}$. C'est le principe de l'algorithme du **gradient à pas fixe**, qui s'écrit

$$x^{k+1} = x^k - \mu \nabla f(x^k), \tag{44}$$

où $\mu > 0$ est un paramètre à fixer appelé *pas*. Comme il ne dépend pas de k, il s'agit d'un pas fixe.

On peut également voir cet algorithme comme le schéma d'Euler appliqué à l'EDO définissant la *plus grande pente* :

$$x'(t) = -\nabla f(x(t)). \tag{45}$$

On pourrait également considèrer l'algorithme de **gradient à pas optimal** : x_0 étant donné, pour $k = 0, 1, \ldots$ faire :

- trouver $\mu^k \geq 0$ qui réalise le minimum de la fonction $\mu \mapsto f(x^k - \mu \nabla f(x^k))$;
- poser $x^{k+1} = x^k - \mu^k \nabla f(x^k)$.

Le théorème suivant donne une condition suffisante sur la fonction f et sur le pas μ pour assurer la convergence de l'algorithme (44) vers un minimiseur de f.

Théorème 7.12 (Convergence de (44)**)** *On suppose que $f \in \mathcal{C}^1(\mathbb{R}^n, \mathbb{R})$ est α-convexe et que $\nabla f(x)$ est L-lipschitzien sur $\mathbb{R}^n$, i.e.*

$$\forall x, y \in \mathbb{R}^n, \quad \|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|.$$

Alors pour tout $0 < \mu < 2\alpha/L^2$, l'algorithme de gradient à pas fixe converge : quel que soit x^0, la suite définie par (44) *converge vers l'unique minimiseur $\bar{x}$ de f dans $\mathbb{R}^n$. De plus, la convergence est géométrique.*

Exemple fondamental : Si $A \in M_n(\mathbb{R})$ est une matrice symétrique définie positive et $b \in \mathbb{R}^n$, alors la fonction

$$f(x) = \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle$$

définie $\forall x \in \mathbb{R}^n$ est α-convexe et son unique minimiseur dans $\mathbb{R}^n$ est également l'unique solution du système linéaire $Ax = b$.

Dans le cas d'un système linéaire, on a en fait le résultat plus précis suivant (cf. cours sur les systèmes linéaires) :

Théorème 7.13 *Soit $A \in M_n(\mathbb{R})$ symétrique définie positive, de valeurs propres $0 < \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$. La méthode du gradient à pas fixe converge si et seulement si $0 < \mu < 2/\lambda_n$, et la vitesse de convergence est optimale lorsque $\mu = 2/[\lambda_1 + \lambda_n]$.*

7.8 Minimisation avec contraintes d'égalité

Nous considérons la situation suivante : $f : \mathbb{R}^n \to \mathbb{R}$ différentiable, et l'ensemble C des contraintes est défini par m égalités

$$C = \{x \in \mathbb{R}^n \mid h_1(x) = 0, \ldots, h_m(x) = 0\}, \tag{46}$$

où les m fonctions $h_i : \mathbb{R}^n \to \mathbb{R}$ sont supposées continûment différentiables sur $\mathbb{R}^n$.

Théorème 7.14 (Conditions nécessaires du premier ordre) *Si $\bar{x} \in C$ est un minimum local de f sur C et si les vecteurs*

$$\nabla h_1(\bar{x}), \ldots, \nabla h_m(\bar{x})$$

sont linéairement indépendants, alors il existe

$$\bar{\lambda} = (\bar{\lambda}_1, \ldots, \bar{\lambda}_m) \in \mathbb{R}^m$$

tels que

$$\nabla f(\bar{x}) + \sum_{i=1}^{m} \bar{\lambda}_i \nabla h_i(\bar{x}) = 0. \tag{47}$$

Les $\bar{\lambda}_i$ sont appelés multiplicateurs de Lagrange. Ils sont uniques. Remarquer qu'on a le même type de condition en un maximum local.

Comment utilise-t-on le théorème 7.14 dans la pratique ? On est confronté à la recherche de $(x, \lambda) \in \mathbb{R}^n \times \mathbb{R}^m$ vérifiant

$$x \in C \text{ et } \nabla f(x) + \sum_{i=1}^{m} \lambda_i \nabla h_i(x) = 0.$$

Il s'agit de résoudre un système (non linéaire en général) à $n+m$ inconnues et $n+m$ équations.

Exemple : On considère l'exemple 3, i.e.

$$\inf_{x \in \mathbb{R}^n, \|x\|=1} \langle Ax, x \rangle,$$

où A est symétrique.

Dans ce cas, $f(x) = \langle Ax, x \rangle$, et $m = 1$ avec $h_1(x) = \|x\|^2 - 1$, de sorte que C est la sphère unité. On a $\nabla h_1(x) = 2x$ et $\nabla f(x) = 2Ax$. Si $\bar{x} \in C$ est un minimiseur de f sur C, alors $h_1(x) \neq 0$ donc il existe $\lambda \in \mathbb{R}$ tel que

$$2Ax + 2\lambda x = 0.$$

Ainsi, x est un vecteur propre de A pour la valeur propre λ.

Attention, l'hypothèse de régularité (vecteurs linéairement indépendants) est importante.
Contre-exemple Soit $f(x_1, x_2) = x_1 + x_2^2$ à minimiser sous la contrainte $h(x_1, x_2) = 0$, avec $h(x_1, x_2) = x_1^3 - x_2^2$. Alors $\bar{x} = (0, 0)$ est le minimum (global) de f sous la contrainte $h(x) = 0$, car dans ce cas

$$x_1 = (x_2^2)^{1/3} \text{ et } f((x_2^2)^{1/3}, x_2) = (x_2^2)^{1/3} + x_2^2 \geq 0,$$

mais on ne peut pas trouver de $\lambda \in \mathbb{R}$ tel que $\nabla f(\bar{x}) + \lambda \nabla h(\bar{x}) = 0$.

Avant de faire la preuve, on va rappeler une définition de géométrie différentielle.

Définition 7.15 *On dit qu'un vecteur d est une direction tangente à C en x si et seulement s'il existe une courbe $c :]-\epsilon, \epsilon[\to \mathbb{R}^n$ différentiable telle que $c(]-\epsilon, \epsilon[) \subset C$, $c(0) = x$ et $c'(0) = d$.*

On a alors le résultat suivant :

Lemme 7.16 *Soit $x \in C$, où C est défini par* (46)*. On suppose que les vecteurs*

$$\nabla h_1(x), \ldots, \nabla h_m(x)$$

sont linéairement indépendants. Alors l'ensemble des vecteurs tangents à C en x est un espace vectoriel, appelé espace tangent à C en x, égal à :

$$T(C, x) = \{d \in \mathbb{R}^n,\ \langle \nabla h_i(x), d \rangle = 0 \text{ pour } i = 1, \ldots, m\}.$$

Il s'agit d'un espace vectoriel de codimension m dans $\mathbb{R}^n$, i.e. de dimension $n - m$.
Preuve du Théorème 7.14. Soit $d \in T(C, \bar{x})$ et soit $c :]-\epsilon, \epsilon[\to \mathbb{R}^n$ différentiable telle que $c(]-\epsilon, \epsilon[) \subset C$, $c(0) = \bar{x}$ et $c'(0) = d$. Alors $t \mapsto f(c(t))$ a un minimum local en $t = 0$, donc sa dérivée $\langle \nabla f(c(t)), c'(t) \rangle$ en $t = 0$ est nulle, i.e. $\langle \nabla f(\bar{x}), d \rangle = 0$. Vrai pour tout $d \in T(C, \bar{x})$. En d'autres termes, $\nabla f(\bar{x}) \in T(C, \bar{x})^\perp$. Or d'après le Lemme 7.16

$$T(C, \bar{x}) = Vect\{\nabla h_1(\bar{x}), \ldots, \nabla h_m(\bar{x})\}^\perp,$$

donc

$$T(C, \bar{x})^\perp = Vect\{\nabla h_1(\bar{x}), \ldots, \nabla h_m(\bar{x})\}.$$

■

Interprétation analytique Soit

$$\mathcal{L}(x, \lambda) \in \mathbb{R}^n \times \mathbb{R}^m \mapsto \mathcal{L}(x, \lambda) := f(x) + \sum_{i=1}^{m} \lambda_i h_i(x).$$

On appelle $\mathcal{L}$ *fonction de Lagrange* ou *Lagrangien* associé au problème de la minimisation de f sur C. Ce qu'exprime (47) est $\nabla_x \mathcal{L}(\bar{x}, \bar{\lambda}) = 0$. Comme d'autre part

$$\nabla_\lambda \mathcal{L}(\bar{x}, \bar{\lambda}) = (h_1(\bar{x}), \ldots, h_m(\bar{x}),$$

ce que dit le Théorème 7.14 est : en un minimum local $\bar{x}$ de f sur C, il existe $\bar{\lambda}$ tel que

$$\nabla \mathcal{L}(\bar{x}, \bar{\lambda}) = (\nabla_x \mathcal{L}(\bar{x}, \bar{\lambda}), \nabla_\lambda \mathcal{L}(\bar{x}, \bar{\lambda})) = 0,$$

i.e. $(\bar{x}, \bar{\lambda})$ est un point critique de $\mathcal{L}$. On passe d'un problème avec contraintes à un problème sans contraintes. Attention, la nature des points n'est pas conservée.

Comme souvent, on a une réciproque sous hypothèse de convexité :

Proposition 7.17 *Supposons $f : U \to \mathbb{R}$ convexe et différentiable sur l'ouvert convexe $U \subset \mathbb{R}^n$, et les h_i affines (i.e. de la forme $x \mapsto h_i(x) = \langle a_i, x\rangle - b_i$). Alors un élément $\bar{x} \in U \cap C$ pour lequel il existe $\bar{\lambda} \in \mathbb{R}^m$ tel que*

$$\nabla f(\bar{x}) + \sum_{i=1}^{m} \lambda_i \nabla h_i(\bar{x}) = 0$$

est un minimum (global) de f sur $U \cap C$.

Preuve. On remarque d'abord que $\langle \nabla f(\bar{x}), x - \bar{x}\rangle = 0$ pour tout $x \in C$. On utilise ensuite la Prop. 7.5, cas (i).

7.9 Exercices mathématiques

1. Etudier (existence, valeur) les extréma globaux de la fonction $f(x, y) = x^4 + y^2$ sur le cercle $x^2 + y^2 = 4$.
2. Etudier de même les extréma globaux de la fonction $f(x, y) = x^2 + y^2$ sur l'hyperbole d'équation $x^2 + 8xy + 7y^2 - 225 = 0$.
3. Soit $n \in \mathbb{N}^\star$. On veut montrer que pour tout $(x_1, \ldots, x_n) \in \mathbb{R}^n_+$,

$$(x_1 x_2 \cdots x_n)^{1/n} \leq \frac{1}{n}(x_1 + x_2 + \cdots + x_n).$$

Pour cela, on considère $f : \mathbb{R}^n_+ \to \mathbb{R}$ définie par $f(x_1, \ldots, x_n) = (x_1 x_2 \cdots x_n)^{1/n}$ et l'ensemble

$$C = \{(x_1, \ldots, x_n) \in \mathbb{R}^n_+ \ : \ x_1 + x_2 + \cdots + x_n = 1\}.$$

a. Montrer que f atteint son maximum sur C en un point $(x_1^\star, \ldots, x_n^\star)$ tel que $x_i^\star > 0$ pour tout i.

b. En déduire le maximum de f sur C et conclure. Quels sont les cas d'égalité ?

c. Redémontrer l'inégalité souhaitée en utilisant la convexité de la fonction $-\ln x$.

4. Soit $n \in \mathbb{N}^\star$. On se donne n réels $\lambda_1 > 0, \ldots, \lambda_n > 0$. On note

$$\Gamma = \{(x_1, \ldots, x_n) \in \mathbb{R}^n \ : \ \sum_{i=1}^{n} \frac{x_i^4}{\lambda_i^4} = 1\}.$$

On définit $f : \mathbb{R}^n \to \mathbb{R}$, $(x_1, \ldots, x_n) \mapsto \sum_{i=1}^n x_i^2$. Déterminer le maximum global de f sur Γ.
5. Soient $a_1, \ldots, a_n > 0$ tels que $a_1 + \cdots + a_n = 1$, avec $n \geq 2$. Démontrer que

$$\Pi_{i=1}^n a_i(1 - a_i) \leq \frac{(n-1)^n}{n^{2n}}.$$

6. Déterminer les triangles d'aire maximum inscrit dans un cercle.

7.10 Exercices informatiques

1. Programmer l'algorithme de gradient à pas fixe pour $f(x) = x^2 + \sin(x)$ sur $\mathbb{R}$. Cette fonction vérifie-t-elle les hypothèses du théorème 7.12 ? Quel pas peut-on choisir ? Vérifiez cela en pratique.

Essayez d'autres exemples : $ch(x)$ sur $\mathbb{R}$ (1-convexe), $x^4/4 - x^3/3 - x^2 + 1$ ou $\sin(x)$ (non convexes), $1/(1 - x^2)$ sur $]-1, 1[$ (domaine non défini partout)
2. Programmer l'algorithme de gradient à pas fixe pour l'exemple fondamental. Représenter

l'erreur en fonction du nombre d'itérations pour différents pas.
3. Même question avec l'algorithme à pas optimal μ^k_{opt}. On montrera que

$$\mu^k_{opt} = \langle d^k, d^k \rangle / \langle Ad^k, d^k \rangle \text{ avec } d^k = Ax^k - b.$$

4. On choisit maintenant un pas $\mu^k = \theta^k \mu^k_{opt}$ où θ^k est choisi aléatoirement dans $[0, 2]$. Comparer la vitesse de convergence de cet algorithme par rapport aux précédents. On pourra choisir par exemple $A = diag([1 : 1 : N])$ avec $N = 100$.

7.11 Références

[HU2, Cia] pour le cours et [HU98] pour des compléments de cours et exercices.

8 La méthode des différences finies sur un problème modèle en dimension 1

On considère le problème modèle en 1d : trouver $u : [0,1] \to \mathbb{R}$ solution de

$$-\frac{d^2u}{dx^2}(x) + c(x)u(x) = f(x), \quad x \in [0,1] \tag{48}$$

$$u(0) = 0, \quad u(1) = 0, \tag{49}$$

où c et f sont deux fonctions données, définies sur [0,1].

8.1 Compléments sur l'étude théorique

Pour les différences finies, on a besoin de plus de régularité qu'en éléments finis. Le théorème que nous utiliserons sera :

Théorème 8.1 (Existence et unicité) *Supposons $c, f \in \mathcal{C}^0([0,1])$ avec $c(x) \geq 0$ pour tout $x \in [0,1]$. Le problème (48)(49) admet une unique solution $u \in \mathcal{C}^2([0,1])$.*

Preuve. Pour le cas général, cf. il faut utiliser le Théorème de Lax-Milgram qui donne existence et unicité d'un solution $u \in H_0^1(]0,1[)$ pour la formulation variationnelle du problème. Ensuite, on utilise que $u''(x) = f(x) - c(x)u(x)$ donc $u'' \in C^0([0,1])$ et on en déduit $u \in C^2([0,1])$ ce type d'argument s'appelle un *bootstrap*.

Dans le cas $c(x) = 0$, on peut donner une démonstration plus élémentaire. Par intégration, $-u''(x) = f(x)$ ssi

$$\exists C_1 \in \mathbb{R},\ u'(x) = -\int_0^x f(s)ds + C_1 \iff \exists C_1, C_2 \in \mathbb{R},\ u(x) = -\int_0^x \int_0^t f(s)dsdt + C_1 x + C_2.$$

On a de plus (49) ssi $u(0) = 0 \Rightarrow C_2 = 0$ et $C_1 = \int_0^1 \int_0^t f(s)dsdt$, d'où unicité et existence. Une autre façon d'écrire, en intégrant par parties, est :

$$u(x) = [-t\int_0^t f(s)ds]_0^x + \int_0^x tf(t)dt + C_1 x = -x\int_0^x f(s)ds + \int_0^x sf(s)ds + C_1 x,$$

avec $C_1 = \int_0^1 f(s)ds - \int_0^1 sf(s)ds$, et donc

$$u(x) = x\int_0^1 f(s)(1-s)ds - \int_0^x f(s)(x-s)ds = \int_0^1 G(x,s)f(s)ds,$$

avec

$$G(x,s) = x(1-s) \text{ si } s \geq x, s(1-x) \text{ si } s \leq x.$$

La fonction G est appelée le *noyau de Green* (ou solution élémentaire) de l'équation. Il y a des généralisations en dimension n sur la boule unité où l'on a explicitement cette fonction de Green. C'est une des techniques classiques dans l'analyse des problèmes elliptiques. ■

Un problème est dit bien posé s'il admet une unique solution et si cette solution dépend continûment des données (pour des normes à expliciter). C'est le cas ici :

Corollaire 8.2 (Continuité de la solution en fonction des données) *Si $c \in \mathcal{C}^0([0,1])$ et $c \geq 0$ sur $[0,1]$, il existe une constante $C > 0$ telle que*

$$\forall f \in \mathcal{C}^0([0,1]), \quad ||u||_{\mathcal{C}^2([0,1])} \leq C||f||_{\mathcal{C}^0([0,1])},$$

où u désigne l'unique solution de (48)(49).

Preuve. C'est le théorème de l'application ouverte appliquée aux espaces de Banach $E = \{f \in \mathcal{C}^2([0,1]), f(0) = f(1) = 0\}$, $F = \mathcal{C}^0([0,1])$ et l'application $T : E \ni u \to -u'' + cu \in F$. L'application T est linéaire continue bijective, donc T^{-1} est continu de F dans E.
Remarque : on peut inclure les conditions aux bords dans le théorème et son corollaire, en prenant $E = \mathcal{C}^2$ et $F = \mathcal{C}^0 \times \mathbb{R}^2$, $\tilde{T}(u) = (Tu, u(0), u(1))$. ■

La propriété suivante est appelée principe du maximum. Elle traduit le fait physique que si on pousse le fil vers le haut, alors le déplacement se fait vers le haut. Elle permet également de comparer des solutions correspondant à des données différentes.

Proposition 8.3 (Principe du maximum) *Supposons $f, c \in \mathcal{C}^0([0,1])$, $c \geq 0$ sur $[0,1]$, et soit $u \in \mathcal{C}^2([0,1]$ la solution de (48)(49). Si $f \geq 0$ sur $[0,1]$, alors $u \geq 0$ sur $[0,1]$.*

Preuve. Si $c \equiv 0$, la formule intégrale le montre. Dans le cas général, on va montrer que c'est vrai si on remplace f par $f + \varepsilon$, puis on va faire tendre ε vers 0, en utilisant le corollaire ci-dessus. Soit donc $\varepsilon > 0$, et $u_\varepsilon \in \mathcal{C}^2([0,1])$ l'unique solution de

$$-u'' + cu = f + \varepsilon, \qquad u_\varepsilon(0) = u_\varepsilon(1) = 0. \tag{50}$$

Si u_ε n'est pas positive sur $[0,1]$, alors u_ε atteint son minimum < 0 en (au moins) un point $x_0 \in]0,1[$. Par formule de Taylor, $u''(x_0) \geq 0$; or en remplaçant dans (50) au point x_0, on voit que le membre de gauche est ≤ 0, tandis que le membre de droite est $\geq \varepsilon > 0$; contradiction. Donc $u_\varepsilon \geq 0$ sur $[0,1]$. Quand $\varepsilon \to 0$, $u_\varepsilon \to u$ en norme $\mathcal{C}^2([0,1])$ d'après le corollaire ci-dessus. Donc $u \geq 0$ sur $[0,1]$. ■

NB : on peut faire un démonstration plus élémentaire, sans utiliser la continuité.

8.2 Exemples de différences finies

L'idée est d'approcher la dérivée de la fonction u en un point x par des quotients aux différences finies (d'où le terme). Par exemple

$$\frac{du}{dx}(x) = \lim_{h \to 0, h>0} \frac{u(x+h) - u(x)}{h}.$$

Quand h est petit, le quotient $[u(x+h) - u(x)]/h$ constitue une bonne approximation de $du/dx(x)$. Plus précisément, en supposant u de classe $\mathcal{C}^2$ au voisinage de x :

$$u(x+h) = u(x) + hu'(x) + \frac{h^2}{2}u''(\xi) \text{ (formule de Taylor)},$$

où $\xi \in]x, x+h[$. Pour un $h_0 > 0$ assez petit, en notant $C = \sup_{y \in [x, x+h_0]} |u''(y)|/2$, on a

$$\forall h \in]0, h_0], \ \left| \frac{u(x+h) - u(x)}{h} - u'(x) \right| \leq Ch.$$

On a défini une approximation consistante d'ordre 1 de u' au point x. Plus généralement, on a une approximation consistante d'ordre p, $(p > 0)$ s'il existe une constant $C > 0$, indépendante de h, telle que cette erreur soit majorée par Ch^p.
Exemples :

$$\frac{u(x) - u(x-h)}{h}$$

est une approximation consistante d'ordre 1 de $u'(x)$, pour u de classe $\mathcal{C}^2$ au voisinage de x (même démo)

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

est une approximation consistante d'ordre 2 de $u'(x)$, pour u de classe $\mathcal{C}^3$ au voisinage de x (exercice). Remarquer que l'approximation n'est que d'ordre 1 si u n'est que de classe $\mathcal{C}^2$: la précision de l'approximation dépend de la régularité de u (général en différences finies).

Pour notre problème, nous avons besoin de la dérivée seconde. Son approximation la plus célèbre par différences finies est donnée par le résultat suivant.

Lemme 8.4 *Supposons u de classe $\mathcal{C}^4$ sur un intervalle $[x-h, x+h]$ avec $h>0$. Alors il existe $\xi \in]x-h, x+h[$ tel que*

$$\frac{u(x+h)-2u(x)+u(x-h)}{h^2} = \frac{d^2u}{dx^2}(x) + \frac{h^2}{12}u^{(4)}(\xi). \tag{51}$$

En particulier, si $u \in C^4([x-h_0, x+h_0])$ (avec $h_0>0$) on a pour tout $h \in]0, h_0]$,

$$\left|\frac{u(x+h)-2u(x)+u(x-h)}{h^2} - \frac{d^2u}{dx^2}(x)\right| \leq Ch^2,$$

avec

$$C = \sup_{y\in[x-h_0,x+h_0]} |u^{(4)}(y)|/12.$$

En d'autres termes, le quotient différentiel $[u(x+h)-2u(x)+u(x-h)]/h^2$ est une approximation consistante d'ordre deux de la dérivée seconde de u au point x.

Preuve. On utilise, comme toujours, la formule de Taylor : l'ordre 4 est donné dans l'énoncé :

$$\begin{aligned} u(x+h) &= u(x) + hu'(x) + \frac{h^2}{2}u''(x) + \frac{h^3}{6}u^{(3)}(x) + \frac{h^4}{24}u^{(4)}(\xi^+), \\ u(x-h) &= u(x) - hu'(x) + \frac{h^2}{2}u''(x) - \frac{h^3}{6}u^{(3)}(x) + \frac{h^4}{24}u^{(4)}(\xi^-), \end{aligned}$$

avec $\xi^+ \in]x, x+h[$, $\xi^- \in]x_h, x[$. Avec le théorème des valeurs intermédiaires, on en déduit (51). ■

Remarquer que si u n'était que de classe $\mathcal{C}^3$, on aurait une erreur d'ordre h et pas mieux. Noter aussi que notre quotient différentiel est égal à $D_h^+ D_h^- = D_h^- D_h^+$, où D_h^+ et D_h^- sont les opérateurs discrets décentrés d'ordre 1 utilisés ci-dessus.

8.3 Approximation du problème par différences finies

Soit $u \in \mathcal{C}^2([0,1])$ solution du pb modèle (c et f continues, $c \geq 0$) :

$$-\frac{d^2u}{dx^2}(x) + c(x)u(x) = f(x), \quad x \in [0,1] \tag{52}$$

$$u(0) = 0, \quad u(1) = 0. \tag{53}$$

On se donne une subdivision uniforme de l'intervalle $[0,1]$ de pas $h = 1/(N+1)$, i.e. $x_i = ih$, $i \in \{0, \ldots, N+1\}$. On cherche, en chacun de ces points, une valeur approchée notée u_i de $u(x_i)$. On prend $u_0 = 0$ et $u_{N+1} = 0$ aux extrémités de l'intervalle (condition (53)). Pour les points internes, on écrit l'équation (52) en chacun des sommets internes x_i, $i \in \{1, \ldots, N\}$ du maillage,

en approchant la dérivée seconde en ce point par le quotient différentiel ci-dessus. On obtient le système suivant :

$$-\frac{u_{i+1}-2u_i+u_{i-1}}{h^2}+c(x_i)u_i=f(x_i),\quad i\in\{1,\dots,N\} \tag{54}$$

$$u_0=0,\ u_{N+1}=0. \tag{55}$$

On dit qu'on a discrétisé le problème par une méthode de différences finies utilisant le "schéma à 3 points" de la dérivée seconde. Il s'agit d'un schéma "centré" : pour évaluer la valeur de u au point x_i, on utilise les valeurs en 3 points centrés autour de x_i.

On est en présence d'un système linéaire d'inconnues $u_1,\dots,u_N$ (ca n'entre pas dans le cadre des suites récurrentes, de même que le problème aux limites n'est pas un pb de Cauchy). Matriciellement, le pb s'écrit (étape intermédiaire, écrire le système ligne par ligne) :

$$A_h u_h = b_h, \tag{56}$$

avec $u_h=(u_1,\dots,u_N)$, $A_h=A_h^{(0)}+diag(c(x_1),\dots,c(x_N))$,

$$A_h^{(0)}=1/h^2[diag(2,N)+diag(-1,1)+diag(-1,-1)],$$

et $b_h=(f(x_1),\dots,f(x_N))$. Pour déterminer la solution discrète u_h, il suffit de résoudre le système linéaire (56).
Question 1 : ce système admet-il une solution ? La matrice A_h est-elle inversible ? On a mieux :

Proposition 8.5 *Supposons $c(x)\geq 0\ \forall x\in[0,1]$. La matrice A_h est symétrique définie positive.*

Preuve. La matrice A_h est clairement symétrique. Montrons qu'elle est définie positive. Soit $X=(x_1,\dots,x_N)\in\mathbb{R}^N$ et calculons ${}^tXA_hX$:

$${}^tXA_hX={}^tXA_h^{(0)}X+\sum_{i=1^N}c(x_i)x_i^2\geq{}^tXA_h^{(0)}X,$$

de sorte qu'il suffit de montrer que $A_h^{(0)}$ est définie positive. On a

$$h^2\,{}^tXA_h^{(0)}X=\sum_{i=1}^{N}(x_{i+1}-x_i)^2,$$

où on a introduit $x_0=x_{N+1}=0$. On en conclut que $A_h^{(0)}$ est positive, et que de plus si ${}^tXA_h^{(0)}X=0$, $x_0=x_1=\ldots=x_N=x_{N+1}=0$. q.e.d. ■

On va également montrer une propriété de A_h qui nous sera utile dans la suite. On commence par la

Définition 8.6 *On dit qu'un vecteur X est positif (noté $X\geq 0$) si toutes ses composantes x_i sont positives.*

Proposition 8.7 (Principe du maximum discret) *On suppose $c(x)\geq 0\ \forall x\geq 0$. Si $u_h\in\mathbb{R}^N$ vérifie $A_hu_h\geq 0$, alors $u_h\geq 0$.*

Preuve. Par l'absurde (ou par contraposée) on suppose que $u_h=(u_1,\dots,u_N)$ n'est pas positive, i.e. que le minimum des u_i est <0. Notons i_0 le plus grand indice tel que $u_{i_0}=\min_{1\leq i\leq N}u_i<0$. On a en particulier $u_{i_0+1}>u_{i_0}$ (on utilise $u_0=0$ et $u_{N+1}=1$). En écrivant (54) au point i_0,

$$-(u_{i_0+1}-u_{i_0})-(u_{i_0-1}-u_{i_0})+h^2c(x_{i_0})u_{i_0}=h^2f(x_{i_0})$$

Le membre de gauche est la somme d'un terme <0, et de deux termes ≤ 0, donc <0. En contradiction avec le membre de droite ≥ 0. ■

Remarque : c'est une propriété importante et qu'on ne retrouve pas (en général) avec l'approche éléments finis.

Corollaire 8.8 *Les coefficients de la matrice A_h^{-1} sont tous positifs ou nuls.*

Preuve. Soit $A_h^{-1} = (\alpha_{ij})_{ij}$ et prenons $b_h = (0, \dots, 0, 1, 0 \dots, 0)$, où le 1 est en $j$ème position. Comme $b_h \geq 0$, alors $A_h^{-1} b_h \geq 0$ d'après la proposition ci-dessus. Or ce vecteur est égal à la $j$ème colonne de $A_h^{-1} b_h$. Ceci est vrai pour chaque colonne, et le corollaire est démontré. ■

Question 2 : convergence de la méthode ? a-t-on par exemple, $u_i - u(x_i) \to 0$ en chacun des sommets du maillage (uniformément), quand $h \to 0$?
Question 2 bis : Précision de la méthode ?

On va répondre à ces deux questions en démontrant le théorème suivant :

Théorème 8.9 *Supposons $u \in \mathcal{C}^4([0,1])$. Alors*

$$\max_{1 \leq i \leq N} |u_i - u(x_i)| \leq \frac{h^2}{96} \max_{x \in [0,1]} |u^{(4)}(x)|.$$

En particulier, le schéma (54)(55) *est convergent, dans le sens où*

$$\lim_{h \to 0} \max_{1 \leq i \leq N} |u_i - u(x_i)| = 0.$$

En reprenant la démo du lemme 8.4, comme $u \in \mathcal{C}^4([0,1])$:

$$\frac{u(x_{i+1}) - 2u(x_i) + u(x_{i-1})}{h^2} = \frac{d^2u}{dx^2}(x_i) + \frac{h^2}{12} u^{(4)}(\xi),$$

avec $\xi \in [x_{i-1}, x_{i+1}]$. Donc

$$-\frac{u(x_{i+1}) - 2u(x_i) + u(x_{i-1})}{h^2} + c(x_i)u(x_i) = f(x_i) - \frac{h^2}{12} u^{(4)}(\xi). \tag{57}$$

Par différence et linéarité, l'erreur numérique $e_i = u_i - u(x_i)$ (liée à l'*erreur de consistance*) satisfait

$$-\frac{e_{i+1} - 2e_i + e_{i-1}}{h^2} + c(x_i)e_i = \frac{h^2}{12} u^{(4)}(\xi).$$

Nous allons considérer ces égalités matriciellement : $A_h e_h = \epsilon_h$, avec $e_h = (e_1, \dots, e_N)$ et $\epsilon_h = h^2/12(u^{(4)}(x_1), \dots, u^{(4)}(x_N)) = (\epsilon_1, \dots, \epsilon_N)$. On a donc $e_h = A_h^{-1} \epsilon_h$. En notant $A_h^{-1} = (\alpha_{ij})_{1 \leq i,j \leq N}$, on a $\forall i \in \{1, \dots, N\}$,

$$|e_i| = |\sum_{j=1}^{N} \alpha_{ij} \epsilon_j| \leq \left(\sum_{j=1}^{N} |\alpha_{ij}| \right) \cdot \max_{1 \leq j \leq N} |\epsilon_j| \leq \frac{1}{8} \cdot \frac{h^2}{12} \sup_{x \in [0,1]} |u^{(4)}(x)|,$$

grâce au corollaire 8.8 et au lemme 8.10.

Lemme 8.10 *Supposons $c \geq 0$ sur $[0,1]$ et notons $A_h^{-1} = (\alpha_{ij})_{1 \leq i,j \leq N}$. Alors,*

$$\forall i \in \{1, \dots, N\}, \ \sum_{j=1}^{N} \alpha_{ij} \leq \frac{1}{8}.$$

Preuve. Rappelons que $A_h^{(0)}$ est la matrice A_h correspondant au cas $c \equiv 0$. On remarque que

$$A_h^{-1} - (A_h^{(0)})^{-1} = A_h^{-1}[A_h^{(0)} - A_h](A_h^{(0)})^{-1}.$$

Comme $A_h^{-1} \geq 0$, $(A_h^{(0)})^{-1} \geq 0$ d'après le corollaire 8.8, et $[A_h^{(0)} - A_h] = -diag(c(x_1), \ldots, c(x_N))$, on a $0 \leq A_h^{-1} \leq (A_h^{(0)})^{-1}$. Il suffit donc de montrer le résultat pour la matrice $(A_h^{(0)})^{-1}$.

L'expression à majorer est le coefficient de la $i$ème ligne du vecteur $(A_h^{(0)})^{-1} \cdot (1, 1, \ldots, 1)$, i.e. la solution du problème (54)(55) correspondant à $f \equiv 1$. La solution exacte du problème continu dans ce cas est donnée par $u(x) = x(1-x)/2$. D'autre part, l'expression (57) concernant l'erreur de consistance montre que si $u^{(4)} \equiv 0$, la solution du problème discret et les valeurs de la solution exacte au points du maillage coïncident. Donc ici, $u_h = 1/2(x_1(1-x_1), \ldots, x_N(1-x_N)) \geq 0$ et $\max_{1 \leq i \leq N} u_i \leq \max_{x \in [0,1]} x(1-x)/2 = 1/8$. CQFD. ■

Remarque : en terme de norme matricielle, on a montré que $||A_h^{(-1)}||_\infty \leq 1/8$.

Les étapes de la démonstration, que l'on retrouvera toujours, sont : l'erreur de consistance, la linéarité, et la stabilité. Ces notions peuvent se généraliser pour d'autres problèmes proches et en dimension supérieure.

8.4 Un peu de vocabulaire

Les étapes utilisées pour la démonstration de la convergence sont similaires à celles qu'on a pour la résolution numérique des EDO.

Définition 8.11 (Erreur de consistance) *L'erreur de consistance du schéma $A_h u_h = b_h$ pour la résolution de* (52)(53) *est le vecteur de $\mathbb{R}^N$*

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

et u est la solution exacte de (52)(53).

Ainsi, $\pi_h u$ est la projection de la solution exacte sur le maillage. Remarquer que cette définition de la consistance est très proche dans l'esprit de celle utilisée pour les EDO, car c'est l"'erreur pour le le schéma appliqué à la solution exacte". On a également les relations

$$\epsilon_h = (A_h \pi_h u - b_h) - (A_h u_h - b_h) = A_h \pi_h u - A_h u_h = A_h(\pi_h u - u_h).$$

Définition 8.12 (Consistance) *Le schéma $A_h u_h = b_h$ est consistant si* $\lim_{h \to 0} \|\epsilon_h\| = 0$.

On a comme pour les EDO la

Définition 8.13 (Ordre) *On dit que le schéma $A_h u_h = b_h$ est d'ordre au moins p s'il existe une constante indépendante de h et $h_0 > 0$ tels que*

$$\forall N \geq 1, \quad \|\epsilon_h\| \leq C h^p.$$

L'ordre p est lié à la vitesse de convergence.

En utilisant $f(x_i) = -u''(x_i) + c(x_i)u(x_i)$, on avait montré que pour chaque composante ϵ_i de ϵ_h (c'est le fameux lemme 8.4 sur l'approximation de la dérivée seconde) :

$$\epsilon_i = \frac{h^2}{12} u^{(4)}(\xi_i),$$

avec $\xi_i \in]x_i - h, x_i + h[$. On en déduit pour $h_0 = 1/2$ par exemple,

Proposition 8.14 *On suppose que la solution u est de classe $\mathcal{C}^4$ sur $[0,1]$. Alors*

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

En d'autres termes, le schéma $A_h u_h = b_h$ est consistant d'ordre 2 pour la norme $\|\cdot\|_\infty$.

On rappelle que

$$\|A\|_\infty := \max_{\|x\|_\infty=1} \|Ax\|_\infty = \max_{i=1,\dots,m} \left(\sum_{j=1}^{n} |a_{ij}| \right).$$

En termes de norme matricielle, le lemme 8.10 et le corollaire 8.8 se traduisent ainsi :

Proposition 8.15 *Supposons que $c \geq 0$. On a l'estimation suivante :*

$$\|A_h^{-1}\|_\infty \leq \frac{1}{8}.$$

C'est un résultat de **stabilité**.

L'erreur de convergence est par définition $\|u_h - \pi_h u\|_\infty$. Comme $u_h - \pi_h u = A_h^{-1}\epsilon_h$ d'après ce qui précède, on a montré

Théorème 8.16 (Convergence) *Supposons $u \in \mathcal{C}^4([0,1])$. Alors*

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

Ainsi, la démonstration de la convergence fait encore intervenir les deux notions de consistance et de stabilité.

8.5 Références

Toute cette section est dans [Luc]. Voir également les nombreux exercices proposés (et corrigés !) et dans le cours, un problème avec conditions aux bord de type Neumann : trouver $u : [0,1] \to \mathbb{R}$ solution de

$$\begin{aligned} -u''(x) + c(x)u(x) &= f(x) \quad x \in]0,1[\\ u'(0) = g_0,\ u'(1) &= g_1, \end{aligned}$$

où $c, f \in \mathcal{C}^0([0,1])$.

9 Introduction à la FFT

FFT= Fast Fourier Transform (en Français, on dit plutôt TFR, pour Transformée de Fourier Rapide).

Comme référence, consulter par exemple [Sch].

9.1 Algorithme de Cooley et Tuckey

La transformée de Fourier discrète (DFT en anglais, pour "Discrete Fourier Transform") est un outil mathématique de traitement du signal numérique et de résolution d'équations aux dérivées partielles. C'est un équivalent discret de la transformée de Fourier continue.

Définition 9.1 *Soit $N \in \mathbb{N}^\star$ et $X = (x_0, x_1, \ldots, x_{N-1}) \in \mathbb{C}^N$. La transformée de Fourier discrète de X est le vecteur $\hat{X} = (\hat{x}_0, \hat{x}_1, \ldots, \hat{x}_N) \in \mathbb{C}^N$ défini par*

$$\hat{x}_k = \sum_{j=0}^{N-1} x_j e^{-\frac{2i\pi}{N}jk}, \quad k = 0, 1, \ldots, N-1.$$

On voit que le calcul ci-dessus est équivalent au produit matrice-vecteur $\hat{X} = F_N X$, où $F_N \in M_N(\mathbb{C})$ est la matrice (carrée, de taille N) de terme général

$$(F_N)_{kj} = e^{-\frac{2i\pi}{N}jk}, \quad 0 \leq k, j \leq N-1.$$

On a la **formule d'inversion :**

$$x_j = \frac{1}{N} \sum_{k=0}^{N-1} \hat{x}_k e^{\frac{2i\pi}{N}jk}, \quad j = 0, 1, \ldots, N-1. \tag{58}$$

Sous forme matricielle, cela équivaut à dire que

$$X = \frac{1}{N} \bar{F}_N \hat{X}.$$

De plus, la matrice $\frac{1}{\sqrt{N}} F_N$ est unitaire.

En 1966, Cooley et Tuckey on trouvé un algorithme permettant de calculer efficacement le produit $F_N x$ lorsque N est une puissance de 2, i.e. $N = 2^m$. Le nombre de multiplications se trouve réduit à $mN/2$, i.e. $Nlog_2 N/2$, au lieu de ce qu'il devrait être a priori, i.e. N^2.

Détaillons cet algorithme. On suppose donc que $N = 2^m$ et on pose $M = 2^{m-1}$ de sorte que $N = 2M$. On définit

$$X_I = \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{M-1} \end{pmatrix}, \; X_{II} = \begin{pmatrix} x_M \\ x_{M+1} \\ \vdots \\ x_{2M-1} \end{pmatrix}, \; \hat{X}_p = \begin{pmatrix} \hat{x}_0 \\ \hat{x}_2 \\ \vdots \\ \hat{x}_{2M-2} \end{pmatrix}, \; \hat{X}_i = \begin{pmatrix} \hat{x}_1 \\ \hat{x}_3 \\ \vdots \\ \hat{x}_{2M-1} \end{pmatrix}.$$

NB : indice p pour "pair" et i pour "impair".

Calcul de $\hat{X}_p$: en utilisant $N = 2M$, on obtient, pour $k = 0, 1, \ldots, M-1$,

$$\begin{aligned}\hat{x}_{2k} &= \sum_{j=0}^{2M-1} x_j e^{-\frac{2i\pi}{2M}2jk}, \\ &= \sum_{j=0}^{M-1} x_j e^{-\frac{2i\pi}{M}jk} + \sum_{j=M}^{2M-1} x_j e^{-\frac{2i\pi}{M}jk}, \\ &= \sum_{j=0}^{M-1} x_j e^{-\frac{2i\pi}{M}jk} + \sum_{j=0}^{M-1} x_{j+M} e^{-\frac{2i\pi}{M}jk},\end{aligned}$$

où l'on a utilisé que $e^{-2i\pi k} = 1$ dans la dernière ligne. Ainsi,

$$\hat{X}_p = F_M X_I + F_M X_{II} = F_M(X_I + X_{II}).$$

Calcul de $\hat{X}_i$: on a de même, pour $k = 0, 1, \ldots, M-1$,

$$\begin{aligned}\hat{x}_{2k+1} &= \sum_{j=0}^{2M-1} x_j e^{-\frac{2i\pi}{2M}j(2k+1)}, \\ &= \sum_{j=0}^{M-1} x_j e^{-\frac{i\pi j}{N}} e^{-\frac{2i\pi}{M}jk} + \sum_{j=M}^{2M-1} x_j e^{-\frac{i\pi j}{N}} e^{-\frac{2i\pi}{M}jk}, \\ &= \sum_{j=0}^{M-1} x_j e^{-\frac{i\pi j}{N}} e^{-\frac{2i\pi}{M}jk} - \sum_{j=0}^{M-1} x_{j+M} e^{-\frac{2i\pi}{M}jk},\end{aligned}$$

où l'on a utilisé que $e^{-i\pi} = -1$. Ainsi,

$$\hat{X}_i = F_M(P_M X_I) - F_M(P_M X_{II}) = F_M(P_M(X_I - X_{II})),$$

où P_M est la matrice diagonale de diagonale $(e^{-\frac{i\pi j}{M}})_{0 \le j \le M-1}$.

Pour récupérer les composantes de $\hat{X}$ dans le bon ordre, il faut ensuite effectuer une permutation sur les lignes.

Soit R_N le nombre de multiplications (complexes) pour le calcul de la FFT avec $N = 2^m$. Le calcul de $\hat{X}_p$ demande $R_{N/2}$ multiplications, celui de $\hat{X}_i$ $R_{N/2} + N/2$. Ainsi,

$$R_N = 2R_{N/2} + N/2,$$

avec $R_1 = 0$. On vérifie par récurrence que $R_N = (N/2)\log_2(N)$, ce qui est le coût annoncé.

9.2 Exemples de programmes récursifs

Une première manière de programmer la FFT est d'utiliser un programme récursif, i.e. faisant appel à lui-même. Nous donnons ici quelques exemples simples de programmes récursifs ; la programmation récursive de la FFT est laissée en exercice.

Les définitions par récurrence sont assez courantes en mathématiques. Prenons le cas du calcul de $n!$ pour n entier, qui est la suite définie par

$$\begin{cases} u_0 = 1, \\ u_n = n u_{n-1}, \quad \text{pour } n \ge 1. \end{cases} \tag{59}$$

Pour calculer $u_n = n!$, la manière "classique" (ou itérative) de programmer est la suivante (fonction "fact2.sci") :

```
function f=fact2(n)
//calcul n! pour n entier >0
f=1;
for k=1:n
    f=k*f;
end
endfunction
```

La programmation récursive permet de traduire "littéralement" la définition par récurrence. La programmation récursive de (59) s'écrira ainsi (fonction "fact3.sci")

```
function f=fact3(n)
// calcul de n! en recursif
   if n<=1 then
      f=1;
   else
      f=n*fact3(n-1);
   end
endfunction
```

La programmation récursive, qui est assez naturelle du point de vue de la définition mathématique, peut parfois est très gourmande en calcul. Il faut la manier avec précaution ; il faut également bien définir les conditions d'arrêt. Prenons l'exemple de la suite de Fibonacci, définie par

$$\begin{cases} u_0 = u_1 = 1, \\ u_n = u_{n-1} + u_{n-2} \quad \text{pour } n \geq 2. \end{cases}$$

Les premiers termes de la suite sont donc 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89,.... La programmation itérative de cette suite est

```
function f=fib2(n)
 // calcul iteratif du n ieme terme de la suite de Fibonnaci
   // fib(0) = 1, fib(1) = 1, fib(n+2) = fib(n+1) + fib(n)
   if n<=1 then
      f=1;
   else
      f=1;g=1;
      for k=2:n
          h=g+f
          f=g;
          g=h;
     end
     f=h;
   end
endfunction
```

La programmation récursive est :

```
function f=fib(n)
   // calcul recursif du n ieme terme de la suite de Fibonnaci
```

```
   // fib(0) = 1, fib(1) = 1, fib(n+2) = fib(n+1) + fib(n)
   if n <= 1 then
      f = 1
   else
      f = fib(n-1) + fib(n-2)
   end
endfunction
```

Comptons le nombre d'appels à la fonction "fib" pour le calcul de fib(n). Un appel consiste à évaluer l'argument, puis à se lancer dans l'exécution de la fonction avec la valeur de l'argument. Pour $fib(4)$, on obtient 9 appels. Plus généralement, si R_n est le nombre d'appels récursifs, on a $R_0 = R_1 = 1$, et $R_n = 1 + R_{n-1} + R_{n-2}$ pour $n \geq 2$. En posant $R'_n = R_n + 1$, on en déduit que

$$\begin{cases} R'_0 = R'_1 = 2, \\ R'_n = R'_{n-1} + R'_{n-2} \quad \text{pour } n \geq 1, \end{cases}$$

d'où $R'_n = 2Fib(n)$. Le nombre d'appels récursifs R_n vaut ainsi $2fib(n) - 1$: il croît exponentiellement en fonction de n.

9.3 Exemple d'utilisation de la FFT

Supposons que nous souhaitons résoudre le problème suivant : étant donné $(f_0, \ldots, f_{N-1}) \in \mathbb{R}^N$, trouver $(u_0, u_1, \ldots, u_{N-1}) \in \mathbb{R}^N$ tel que

$$\frac{-u_{j+1} + 2u_j - u_{j-1}}{h^2} + u_j = f_j \quad j = 0, 1, \ldots, N-1, \tag{60}$$

avec la convention $u_N = u_0$ et $u_{-1} = u_{N-1}$ et $h = 1/N$.

Il s'agit d'une discrétisation par différences finies du problème : trouver $u : [0,1] \to \mathbb{R}$ solution de

$$-u''(x) + u(x) = f(x) \text{ pour } x \in (0,1), \quad u(0) = u(1) \text{ et } u'(0) = u'(1).$$

Le problème continu se résout facilement par utilisation des coefficients de Fourier (et le fait que u est périodique).

Le problème discret va pouvoir être résolu facilement avec l'utilisation de la FFT. En effet, si l'on multiplie chaque équation (60) par $e^{-2i\pi jk/N}$ et que l'on somme sur $j \in \{0, \ldots, N-1\}$, on obtient

$$-\frac{e^{2i\pi k/N}\hat{u}_k + 2\hat{u}_k - e^{-2i\pi k/N}\hat{u}_k}{h^2} + \hat{u}_k = \hat{f}_k \quad k = 0, 1, \ldots, N-1.$$

En simplifiant, on trouve

$$[4N^2 \sin^2(k\pi/N) + 1]\hat{u}_k = \hat{f}_k \quad \forall k.$$

On en déduit $\hat{u}_k$, puis u par la transformée de Fourier inverse.

Cette résolution du système linéaire initial demande N divisions et $N/2\log_2(N)$ multiplications, à comparer aux $O(N)$ multiplications et additions nécessaires pour résoudre un système tridiagonal. On ne gagne pas vraiment en temps de calcul en dimension 1.

Cependant, le problème et la méthode de résolution peuvent s'étendre en dimension 2 sur un carré et en dimension 3 sur un cube. Le gain en temps de calcul est alors foudroyant.

9.4 Exercices de programmation

1. Programmer les fonctions fib2.sci et fib.sci. Comparer leur temps d'exécution (tic,toc). On pourra également afficher les calculs intermédiaires (disp). Démontrer également l'affirmation "R_n croît exponentiellement" en fonction de n.
2. Programmer l'algorithme de Cooley et Tuckey de manière récursive, sous la forme d'une fonction `hx=myFFT(x)` prenant en entrée un vecteur x de taille $N = 2^m$ et donnant en sortie le vecteur `hx`$= \hat{x}$. Comparer votre résultat aux fonctions existantes `dft` et `fft`, pour des petites valeurs de N.
3. Quelle est le N maximal pouvant être pris en compte par votre algorithme FFourierT et par `dft`, `fft` ? Comparer les vitesses d'exécution de ces trois algorithmes pour N grand.
4. Démontrer la formule d'inversion (58) et l'affirmation que $\frac{1}{\sqrt{N}}F_N$ est unitaire.
5. Pour calculer le coût d'un algorithme, on prend la convention suivante : 1 addition complexe= 2 additions réelles et 1 multiplication complexe = 2 additions réelles et quatre multiplications réelles.
6. Montrer que le produit $F_N X$ est de N^2 multiplications complexes et $N(N-1)$ additions complexes.
7. Montrer que la FFT pour un vecteur de taille $N = 2^m$ demande au plus $5N \log_2 N$ opérations (addition ou multiplication) réelles.
8. Résoudre le problème (60) par la FFT, comme indiqué, sur un cas-test que l'on choisira (par exemple, $u(x) = 0.5 * \sin(2\pi x) + \cos(4\pi x)$).

Références

[Cia] Ciarlet. *Introduction à l'analyse numérique et à l'optimisation.*

[CM84] Michel Crouzeix and Alain L. Mignot. *Analyse numérique des équations différentielles.* Collection Mathématiques Appliquées pour la Maîtrise. [Collection of Applied Mathematics for the Master's Degree]. Masson, Paris, 1984.

[Dem91] J.-P. Demailly. *Analyse numérique et équations différentielles.* Presses Universitaires de Grenoble, 1991.

[Dum] L. Dumas. *Modélisation à l'épreuve de l'agrégation.*

[Fer] J.-M. Ferrard. *Maths et Maple.* Dunod.

[Hai] E. Hairer. Introduction à l'analyse numérique (non publié). http ://www.unige.ch/∼hairer/polycop.html.

[HU98] Jean-Baptiste Hiriart-Urruty. *Optimisation et analyse convexe.* Mathématiques. [Mathematics]. Presses Universitaires de France, Paris, 1998.

[HU2]

[Luc] B. Lucquin. *Equations aux dérivées partielles et leurs approximations.*

[Pom] A. Pommelet. *Agrégation de mathématiques : cours d'analyse.* Ellipses.

[PR] J. Picasso and M. Rappaz. *Introduction à l'analyse numérique.* Presses polytechniques et universitaires romandes.

[Sch] M Schatzmann. *Analyse numérique : une approche numérique.* Dunod.

[TL] P. Théodore and R. Lascaux. *Analyse numérique matricielle appliquée à l'art de l'ingénieur.* Dunod.

[ZQ95] C. Zuily and H. Quéffelec. *Eléments d'analyse pour l'agrégation.* Masson, 1995.