Chapitre II

Interpolation et Approximation

Problème de l'interpolation : on recherche des fonctions "simples" (polynômes, polynômes par morceaux, polynômes trigonométriques) passant par (ou proche) des points donnés

$$(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n), \tag{0.1}$$

c.-à-d., on cherche $p(x)$ avec $p(x_i) = y_i$ pour $i = 0, 1, \ldots, n$. Si les valeurs de y_i satisfont $y_i = f(x_i)$ où $f(x)$ est une fonction donnée, il est alors intéressant d'étudier l'erreur de l'*approximation*

$$f(x) - p(x) = \ ? \tag{0.2}$$

Bibliographie de ce chapitre

J.H. Ahlberg, E.N. Nilson & J.L. Walsh (1967): *The Theory of Splines and Their Applications*. Academic Press, New York. [MA 65/4]

C. de Boor (1978): *A Practical Guide to Splines*. Springer-Verlag. [MA 65/141]

G.D. Knott (2000): *Interpolating Cubic Splines.* Birkhäuser. [MA 65/431]

H.J. Nussbaumer (1981): *Fast Fourier Transform and Convolution Algorithms*. Springer-Verlag.

H. Späth (1995): *One Dimensional Spline Interpolation.* AK Peters. [MA 65/362]

II.1 Différences divisées et formule de Newton

> ''. . . tho' I will not undertake to prove it to others."
>
> (Newton, letter to Collins, Nov. 8, 1676 ; publ. Cotes 1711, p. 38)

Problème (Newton 1676). Étant donnés les $n + 1$ points (0.1), chercher un polynôme

$$p(x) = ax^n + bx^{n-1} + cx^{n-2} + \ldots \tag{1.1}$$

de degré n qui satisfasse

$$p(x_i) = y_i \qquad \text{pour} \qquad i = 0, 1, \ldots, n. \tag{1.2}$$

Pour un exemple voir la fig. II.1.

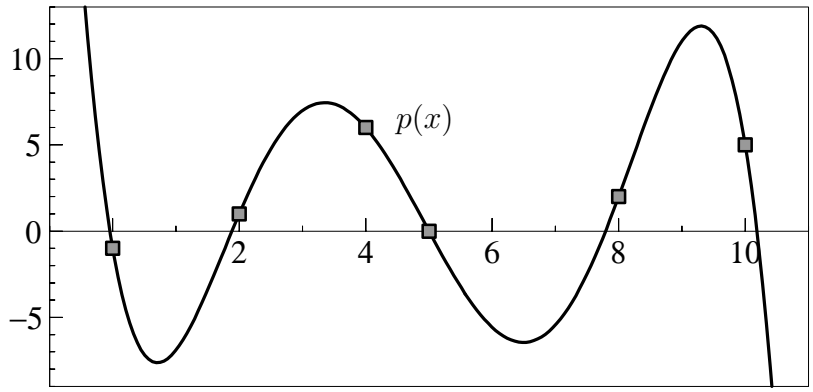


FIG. II.1: Polynôme d'interpolation de degré 5

Solution. En insérant les conditions (1.2) dans (1.1), le problème se transforme en un système linéaire (à matrice du type Vandermonde ; ici écrit pour $n = 2$)

$$\begin{aligned} c + bx_0 + ax_0^2 &= y_0 \\ c + bx_1 + ax_1^2 &= y_1 \\ c + bx_2 + ax_2^2 &= y_2 \end{aligned} \qquad \text{soustraire et diviser} \qquad \begin{aligned} b + a(x_1 + x_0) &= \frac{y_1 - y_0}{x_1 - x_0} \\ b + a(x_2 + x_1) &= \frac{y_2 - y_1}{x_2 - x_1} \end{aligned} \tag{1.3}$$

et, si on soustrait et divise une deuxième fois, on trouve

$$a = \frac{1}{x_2 - x_0}\Big(\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}\Big). \tag{1.4}$$

Le même calcul a été effectué pour $n = 4$ dans un manuscript de Newton datant de 1676 ; comme à l'accoutumée, Newton refusa de le publier (voir citation). Cotes le publia comme dernier chapitre *Methodus differentialis* du livre *Analysis per quantitatum series, fluxiones, ac differentias*, Londini 1711 (voir fac-similé en figure II.2 [1]).

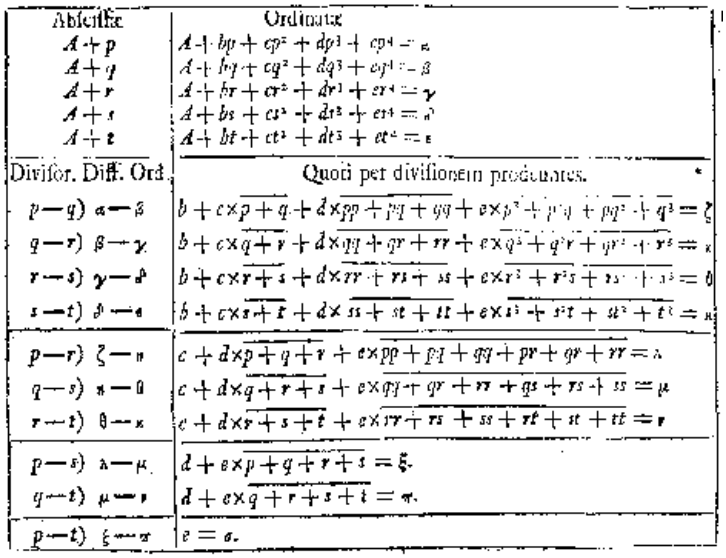


FIG. II.2: Fac-similé du calcul de Newton pour le problème de l'interpolation

Dans tous ces calculs apparaissent les "différences divisées" :

[1] On peut observer que Newton maîtrise les éliminations de variables dans un système linéaire avec brio ; plus tard, toute la gloire pour cette méthode reviendra à Gauss.

Définition 1.1 (différences divisées) Pour (x_i, y_i) donnés (x_i distincts) on définit

$$\begin{aligned} y[x_i] &:= y_i \\ \delta y[x_i, x_j] &:= \frac{y[x_j] - y[x_i]}{x_j - x_i} \\ \delta^2 y[x_i, x_j, x_k] &:= \frac{\delta y[x_j, x_k] - \delta y[x_i, x_j]}{x_k - x_i} \\ \delta^3 y[x_i, x_j, x_k, x_l] &:= \frac{\delta^2 y[x_j, x_k, x_l] - \delta^2 y[x_i, x_j, x_k]}{x_l - x_i} \quad \text{etc.} \end{aligned}$$

Ensuite Newton, "charmant" comme toujours, donne la formule suivante sans la moindre démonstration (voir citation) :

Théorème 1.2 (formule de Newton) *Le polynôme d'interpolation de degré n qui passe par les $n+1$ points $(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)$, où les x_i sont distincts, est unique et donné par*

$$\begin{aligned} p(x) = y[x_0] + (x - x_0)\,\delta y[x_0, x_1] + (x - x_0)(x - x_1)\,\delta^2 y[x_0, x_1, x_2] \\ + \ldots + (x - x_0)(x - x_1) \cdot \ldots \cdot (x - x_{n-1})\,\delta^n y[x_0, x_1, \ldots, x_n]. \end{aligned} \tag{1.5}$$

Démonstration. Nous utilisons deux idées :

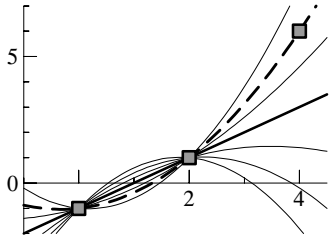
1. On procède par récurrence. Pour $n = 1$, et en tenant compte des premiers deux points, nous avons

$$p(x) = y_0 + (x - x_0)\,\frac{y_1 - y_0}{x_1 - x_0}. \tag{1.6}$$

Il s'agit d'une formule bien connue des Géomètres (voir $\Gamma\epsilon\omega\mu\epsilon\tau\rho\acute{\iota}\alpha$, figure II.1.8).

Puis, pour $n = 2$, en rajoutant le point (x_2, y_2), on essaie de bâtir là dessus un polynôme de degré 2, qui ne change plus les valeurs de y_0 et de y_1. Il est donc de la forme

$$p(x) = y_0 + (x - x_0)\,\frac{y_1 - y_0}{x_1 - x_0} + a \cdot (x - x_0)(x - x_1) \tag{1.7}$$



où le coefficient a est à déterminer. Mais il s'agit du coefficient de x^2 de $p(x)$: nous savons déjà (voir (1.4)) que celui-ci est la deuxième différence divisée $\delta^2 y[x_0, x_1, x_2]$.

Pour démontrer le cas général, nous supposons que

$$p_1(x) = y[x_0] + (x - x_0)\,\delta y[x_0, x_1] + \ldots + (x - x_0) \cdot \ldots \cdot (x - x_{n-2})\,\delta^{n-1} y[x_0, x_1, \ldots, x_{n-1}]$$

soit le polynôme unique de degré $n-1$ qui passe par (x_i, y_i) pour $i = 0, 1, \ldots, n-1$. Alors, comme auparavant, le polynôme $p(x)$ a nécessairement la forme

$$p(x) = p_1(x) + a \cdot (x - x_0)(x - x_1) \cdot \ldots \cdot (x - x_{n-1}),$$

où a est déterminé par $p(x_n) = y_n$.

2. L'idée de Aitken-Neville. Pour montrer que $a = \delta^n y[x_0, x_1, \ldots, x_n]$, ce qui achève la démonstration, nous considérons également le polynôme de degré $n-1$

$$p_2(x) = y[x_1] + (x - x_1)\,\delta y[x_1, x_2] + \ldots + (x - x_1) \cdot \ldots \cdot (x - x_{n-1})\,\delta^{n-1} y[x_1, x_2, \ldots, x_n],$$

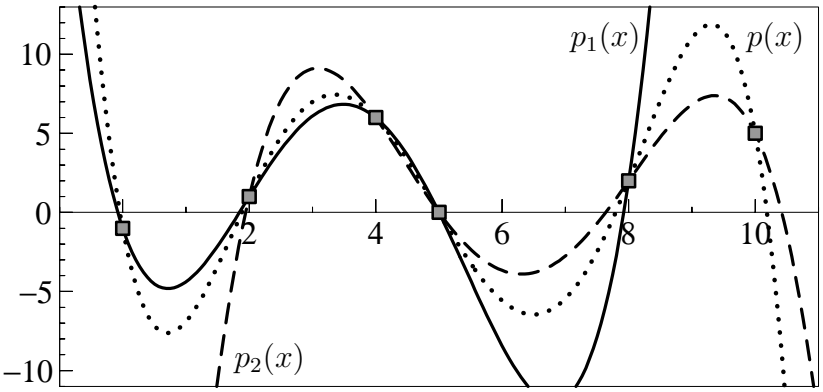


FIG. II.3: Les polynômes $p_1(t)$, $p_2(t)$ et $p(t)$ de l'algorithme d'Aitken-Neville

qui passe par (x_i, y_i) pour $i = 1, \ldots, n$ (voir figure II.3). Ensuite, on pose (Aitken - Neville, 1929, 1932 [2])

$$p(x) = \frac{1}{x_n - x_0}\Big((x_n - x)p_1(x) + (x - x_0)p_2(x)\Big). \tag{1.8}$$

Il s'agit d'un polynôme de degré n, qui satisfait la condition (1.2) pour le point x_0 (ici, le facteur $(x - x_0)$ est nul), pour le point x_n (ici, le facteur $(x - x_n)$ est nul), et pour les points $x_1, \ldots, x_{n-1}$ (ici, les deux polynômes p_1 et p_2 sont égaux à y_i). Le polynôme désiré est donc trouvé.

En considérant le coefficient de x^n dans (1.8), nous obtenons

$$a = \frac{1}{x_n - x_0}\Big(\delta^{n-1}y[x_1, \ldots, x_n] - \delta^{n-1}y[x_0, \ldots, x_{n-1}]\Big) = \delta^n y[x_0, \ldots, x_n],$$

ce qui démontre la formule (1.5). □

TAB. II.1: Différences divisées pour les données de la fig. II.1

x_i	y_i	δy	$\delta^2 y$	$\delta^3 y$	$\delta^4 y$	$\delta^5 y$
0	−1					
		1				
2	1		3/8			
		5/2		−77/120		
4	6		−17/6		167/960	
		−6		3/4		−287/9600
5	0		5/3		−1/8	
		2/3		−1/4		
8	2		1/6			
		3/2				
10	5					

Exemple 1.3 Pour les données de la fig. II.1, les différences divisées sont présentées dans le tableau II.1. Le polynôme d'interpolation est alors donné par

$$p(x) = -1 + x + x(x-2)\frac{3}{8} - x(x-2)(x-4)\frac{77}{120} + x(x-2)(x-4)(x-5)\frac{167}{960} - x(x-2)(x-4)(x-5)(x-8)\frac{287}{9600}.$$

ou mieux encore pour la programmation (ou le calcul à la main)

$$p(x) = -1 + x\Big(1 + (x-2)\Big(\frac{3}{8} + (x-4)\Big(-\frac{77}{120} + (x-5)\Big(\frac{167}{960} - (x-8)\frac{287}{9600}\Big)\Big)\Big)\Big).$$

[2] Il fallait plus de deux siècles pour avoir cette idée !...

Remarque. L'ordre des $\{x_i\}$ n'a aucune importance pour la formule de Newton (1.5). Si l'on permute les données (x_i, y_i), on obtient évidemment le même polynôme. Pour l'exemple ci-dessus et pour les $\{x_i\}$ choisis dans l'ordre $\{4, 5, 2, 8, 0, 10\}$, on obtient ainsi

$$p(x) = 6 + (x-4)\Big(-6 + (x-5)\Big(-\frac{17}{6} + (x-2)\Big(\frac{3}{4} + (x-8)\Big(\frac{167}{960} - x\frac{287}{9600}\Big)\Big)\Big)\Big).$$

En observant que $\delta^n y[x_{i_0}, \ldots, x_{i_n}]$ est une fonction symétrique de ses arguments (par exemple, $\delta^2 y[x_2, x_3, x_1] = \delta^2 y[x_1, x_2, x_3]$, voir exercices), on peut utiliser les valeurs calculées dans le tableau II.1.

Si x est entre 4 et 5, les deux facteurs $x-4$ et $x-5$ dans la formule précédente sont relativement petits, ce qui favorise la diminution des erreurs d'arrondi.

II.2 Erreur de l'interpolation

Supposons que les points (x_i, y_i) soient sur le graphe d'une fonction $f : [a, b] \to \mathbb{R}$, c.-à-d.,

$$y_i = f(x_i), \qquad i = 0, 1, \ldots, n, \tag{2.1}$$

étudions alors l'erreur $f(x) - p(x)$ du polynôme d'interpolation $p(x)$. Deux exemples sont donnés dans la fig. II.4. A gauche, on voit un polynôme d'interpolation pour la fonction $f(x) = \sin x$, et à droite pour la fonction $1/(1+x^2)$. Pour mieux rendre visible l'erreur, on a dessiné la fonction $f(x)$ en une courbe pointillée.

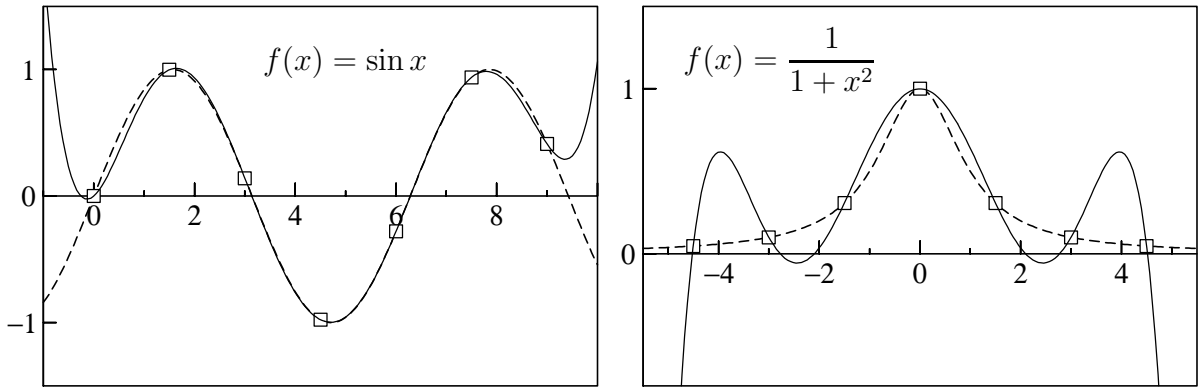


FIG. II.4: Polynôme d'interpolation pour $\sin x$ (gauche) et pour $1/(1+x^2)$ (droite)

Les résultats suivants sont dus à Cauchy (1840, *Sur les fonctions interpolaires*, C.R. XI, p. 775-789, *Oeuvres* ser. 1, vol. V, p. 409-424). Commençons par une relation intéressante entre les différences divisées pour (2.1) et les dérivées de la fonction $f(x)$.

Lemme 2.1 *Soit $f(x)$ n-fois différentiable et $y_i = f(x_i)$ pour $i = 0, 1, \ldots, n$ (x_i distincts). Alors, il existe un $\xi \in (\min x_i, \max x_i)$ tel que*

$$\delta^n y[x_0, x_1, \ldots, x_n] = \frac{f^{(n)}(\xi)}{n!}. \tag{2.2}$$

Démonstration. Soit $p(x)$ le polynôme d'interpolation de degré n passant par (x_i, y_i) et notons $d(x) = f(x) - p(x)$. Par définition de $p(x)$, la différence $d(x)$ s'annule en $n+1$ points distincts :

$$d(x_i) = 0 \qquad \text{pour} \qquad i = 0, 1, \ldots, n.$$

Comme $d(x)$ est différentiable, on peut appliquer n fois le théorème de Rolle (voir le cours d'Analyse I) et on en déduit que

$$d'(x) \qquad \text{a } n \text{ zéros distincts dans } (\min x_i, \max x_i).$$

Le même argument appliqué à $d'(x)$ donne

$$d''(x) \qquad \text{a } n-1 \text{ zéros distincts dans } (\min_i x_i, \max_i x_i),$$

et finalement encore

$$d^{(n)}(x) \qquad \text{a 1 zéro dans } (\min_i x_i, \max_i x_i).$$

Notons ce zéro de $d^{(n)}(x)$ par ξ. Alors, on a

$$f^{(n)}(\xi) = p^{(n)}(\xi) = n! \cdot \delta^n y[x_0, x_1, \ldots, x_n]. \tag{2.3}$$

La deuxième identité dans (2.3) résulte du fait que $\delta^n y[x_0, x_1, \ldots, x_n]$ est le coefficient de x^n dans $p(x)$. □

Théorème 2.2 *Soit $f : [a, b] \to \mathbb{R}$ $(n+1)$-fois différentiable et soit $p(x)$ le polynôme d'interpolation de degré n qui passe par $(x_i, f(x_i))$ pour $i = 0, 1, \ldots, n$. Alors, pour $x \in [a, b]$, il existe un $\xi \in (\min(x_i, x), \max(x_i, x))$ tel que*

$$f(x) - p(x) = (x - x_0) \cdot \ldots \cdot (x - x_n) \cdot \frac{f^{(n+1)}(\xi)}{(n+1)!}. \tag{2.4}$$

Démonstration. Si $x = x_i$ pour un indice $i \in \{0, 1, \ldots, n\}$, la formule (2.4) est vérifiée car $p(x_i) = f(x_i)$. Fixons alors un $\bar{x}$ dans $[a, b]$ qui soit différent de x_i et montrons la formule (2.4) pour $x = \bar{x}$.

L'idée est de considérer le polynôme $\bar{p}(x)$ de degré $n+1$ qui passe par $(x_i, f(x_i))$ pour $i = 0, 1, \ldots, n$ et par $(\bar{x}, f(\bar{x}))$. La formule de Newton donne

$$\bar{p}(x) = p(x) + (x - x_0) \cdot \ldots \cdot (x - x_n) \cdot \delta^{n+1} y[x_0, \ldots, x_n, \bar{x}]. \tag{2.5}$$

Si l'on remplace la différence divisée dans (2.5) par $f^{(n+1)}(\xi)/(n+1)!$ (voir le lemme précédent) et si l'on pose $x = \bar{x}$, on obtient le résultat (2.4) pour $x = \bar{x}$ car $\bar{p}(\bar{x}) = f(\bar{x})$. Comme $\bar{x}$ est arbitraire, la formule (2.4) est vérifiée pour tout x. □

Exemple 2.3 Dans la situation de la fig. II.4, on a $n + 1 = 7$. Comme la $7^{\text{ème}}$ dérivée de $\sin x$ est bornée par 1, on a que

$$|p(x) - \sin x| \leq |x(x - 1.5)(x - 3)(x - 4.5)(x - 6)(x - 7.5)(x - 9)| \cdot \frac{1}{7!},$$

par exemple

$$|p(4) - \sin 4| \leq 0.035 \qquad \text{ou} \qquad |p(1) - \sin 1| \leq 0.181.$$

Pour le deuxième exemple, $f(x) = 1/(1 + x^2)$, la $7^{\text{ème}}$ dérivée est donnée par

$$f^{(7)}(x) = -8! \cdot \frac{(x+1)(x-1)x(x^2 - 2x - 1)(x^2 + 2x - 1)}{(1 + x^2)^8},$$

qui est maximale pour $x \approx \pm 0.17632698$. On obtient ainsi

$$\Big|p(x) - \frac{1}{1 + x^2}\Big| \leq |(x^2 - 20.25)(x^2 - 9)(x^2 - 2.25)x| \cdot \frac{4392}{7!}.$$

Alors, l'erreur peut être 4392 fois plus grande que pour l'interpolation de $\sin x$.

Convergence de l'interpolation.

- Une grande surprise en mathématiques fut la découverte, d'abord par Riemann (1854), puis par Weierstrass (1872), de l'incroyable complexité qu'ont certaines fonctions continues, p. ex. , de n'être nulle part différentiables ;
- puis la deuxième grande surprise : toutes ces fonctions, aussi compliquées qu'elles puissent être, peuvent être approchées, aussi près qu'on le veut et uniformément, par les fonctions les plus simples qui existent, des polynômes (Weierstrass 1885 ; voir [HW96], §III.9) ;
- personne ne pensait alors que les polynômes d'interpolation, si on prend seulement les points suffisamment proches les uns des autres, ne convergeaient pas vers la fonction donnée. La découverte que cela n'est même pas assuré pour les fonctions rationnelles, les deuxièmes fonctions les plus simples, (voir dessin de figure II.5), a choqué énormément les mathématiciens vers 1900 (en particulier E. Borel).

Carl David Tolmé Runge (1856-1927), premier prof de maths appliquées de l'histoire et, en tant qu'élève de Weierstrass, ayant aussi une solide formation en maths pures, fut certes l'homme idéal pour expliquer ce phénomène de manière claire et élégante (1901, *Zeitschr. Math. u. Physik* vol. 46).

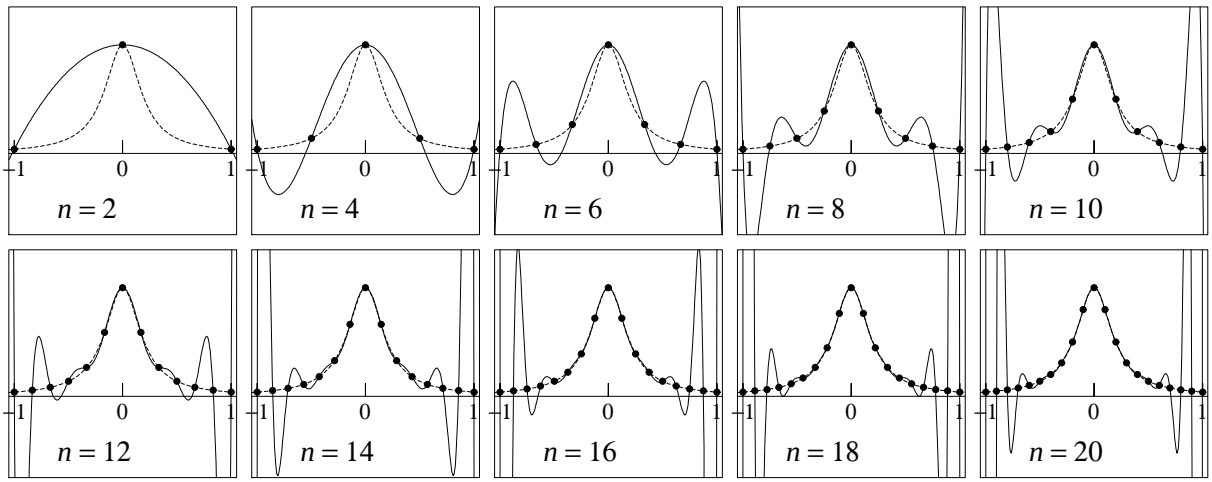


FIG. II.5: Le phénomène de Runge pour $f(x) = 1/(1 + 25x^2)$

II.3 Polynômes de Chebyshev

La formule (2.4) montre que l'erreur de l'interpolation est un produit de la $(n+1)^{\text{ème}}$ dérivée de $f(x)$, évaluée à un point inconnu, avec l'expression $(x - x_0) \cdot \ldots \cdot (x - x_n)$ qui ne dépend que de la division $\{x_0, \ldots, x_n\}$. Nous arrivons à la question suivante :

Chercher, pour un n donné, la division de $[a, b]$ pour laquelle

$$\max_{x\in[a,b]} |(x - x_0) \cdot \ldots \cdot (x - x_n)| \qquad \text{est minimal.} \tag{3.1}$$

Nous considérons l'intervalle $[-1, 1]$ et avons le problème :

Problème. Chercher un polynôme (avec coefficient principal $= 1$)

$$\tau(x) = x^n + a_{n-1}x^{n-1} + \ldots + a_0 \quad \text{tel que} \quad L = \max_{x\in[-1,1]} |\tau(x)| \quad \text{est minimal.} \tag{3.2}$$

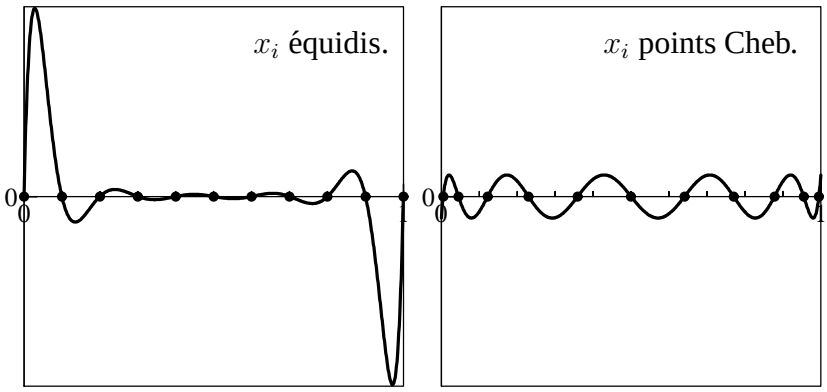


FIG. II.6: Le produit $(x - x_0) \cdot (x - x_1) \ldots \cdot (x - x_n)$ pour $n = 10$ et x_i équidistants (gauche), x_i poits de Chebyshev (droite).

On trouve la réponse à cette question dans un travail de P.L. Chebyshev (transcription française : "Tchebychef", 1854, *Œuvres* I, p. 109) sur la conception optimale des tiges des pistons de locomotives à vapeur . . . . . Entre-temps, les locomotives à vapeur sont au musée, et les polynômes de Chebyshev sont encore et toujours des outils importants en mathématiques.

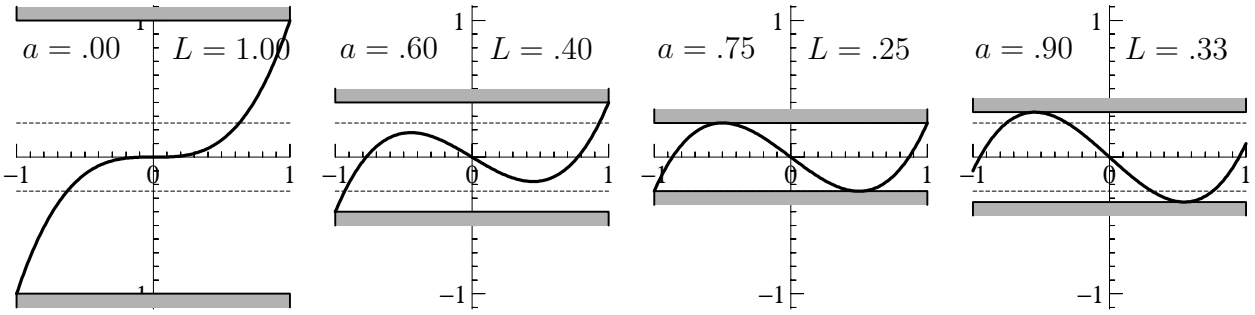


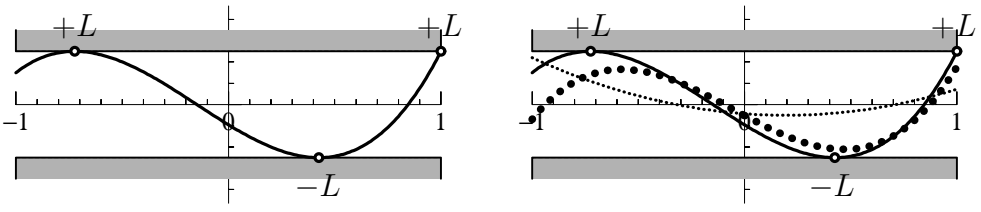
FIG. II.7: Valeurs maximales de $\tau(x) = x^3 - ax$

Le cas $n = 3$**.** Cherchons à résoudre cette question pour $n = 3$, i.e., posons (symétrie oblige !...) $\tau(x) = x^3 - ax$ où a est un paramètre à trouver. Nous voyons en figure II.7 que, pour $a = 0$, nous avons $L = 1$; puis cette valeur diminue quand a croît, jusqu'au moment où la courbe touche la borne $+L$ et $-L$ en même temps (pour $a = 3/4$) ; après, L recommence de grandir. La solution optimale est donc

$$\tau_3(x) = x^3 - \frac{3}{4}x \; . \tag{3.3}$$

Chebyshev a vu que pour un n quelconque, le polynône optimal possède, comme on dit aujourd'hui, une *alternante de Chebyshev* :

Théorème 3.1 *Le polynôme (3.2), minimisant L sur $[-1, 1]$, prend alternativement les valeurs $+L$ et $-L$ exactement $n + 1$ fois.*



Démonstration. La preuve est simple : supposons, par exemple pour $n = 3$, que le polynôme $\tau_3(x) = x^3 + \dots$ prenne seulement *trois* fois les valeurs maximales, disons, $+L$, puis $-L$, puis $+L$ (voir dessin précédent). Il existe alors un polynôme $q_2(x)$ de degré 2, qui est > 0, < 0, et > 0 à ces trois points. Le polynôme $\tau_3(x) - \epsilon q_2(x)$ va donc, pour $\epsilon > 0$, diminuer en valeur absolue *à tous les trois points*. Par conséquent, le polynôme n'était pas optimal. □

Par contre, pour n quelconque, il est beaucoup plus difficile de trouver des formules explicites pour ces polynômes. L'idée suivante est due à Zolotarev :

Idée. Multiplions $\tau_3(x)$ de (3.3) par 4 et posons

$$T_3(x) = 4x^3 - 3x \qquad \text{ce qui ressemble à} \qquad \cos(3\varphi) = 4\cos^3\varphi - 3\cos\varphi \tag{3.4}$$

(cf. l'équation pour la trisection de l'angle ; $\Gamma\epsilon\omega\mu\epsilon\tau\rho\acute{\iota}\alpha$, p. 79, ou [HW97], p. 7). On voit bien en figure II.8, que cela représente la projection d'une guirlande sur un tambour $x = \cos\varphi, y = \sin\varphi, z = \cos(3\varphi)$ sur le plan (x, z). Il est maintenant facile d'étendre cette idée au cas général :

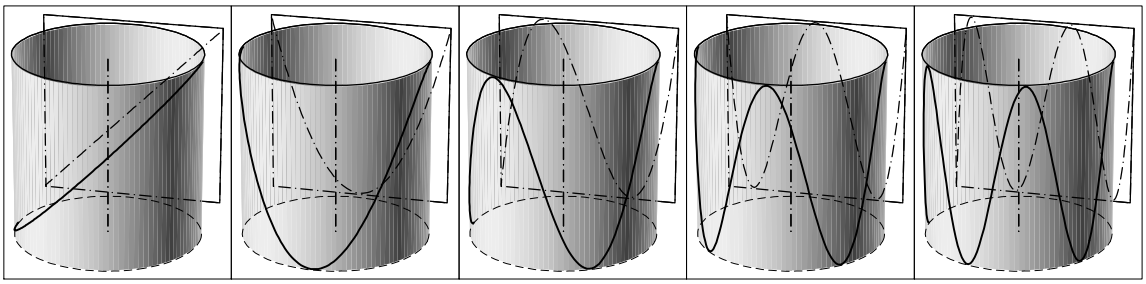


FIG. II.8: Les guirlandes $z = \cos(n\varphi)$ autour d'un tambour $x = \cos\varphi, y = \sin\varphi$ et leurs projections, les polynômes de Chebyshev, sur le plan (x, z).

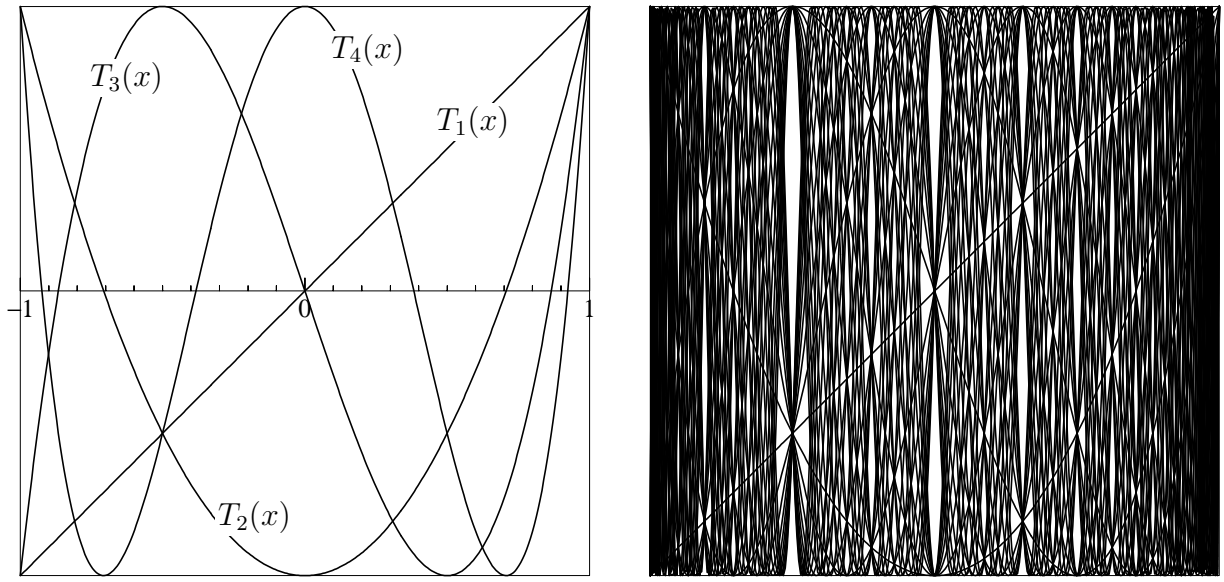


FIG. II.9: Les premiers 4 (à gauche) respectivement 30 (à droite) polynômes de Chebyshev

Définition 3.2 (Polynômes de Chebyshev) *Pour $n = 0, 1, 2, \dots$ et pour $x \in [-1, 1]$, on définit*

$$T_n(x) = \cos(n\varphi) \quad \textit{où} \qquad x = \cos\varphi\,. \tag{3.5}$$

Par les formules de Moivre (cf. [HW97], p. 45), on peut voir que, malgré cette définition étrange, $T_n(x)$ est un polynôme en x. Mais il y a mieux : la formule

$$\cos((n+1)\varphi) + \cos((n-1)\varphi) = 2\cos\varphi \cdot \cos(n\varphi),$$

en posant $\cos\varphi = x$, devient

$$T_0(x) = 1, \qquad T_1(x) = x, \qquad T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x). \tag{3.6}$$

Par conséquent, $T_n(x)$ est un polynôme de degré n dont le coefficient de x^n est 2^{n-1}, c.-à-d., $T_n(x) = 2^{n-1}x^n + \ldots$. Les premiers sont $T_0(x) = 1,\ T_1(x) = x$,

$$T_2(x) = x^2{-}1, \quad T_3(x) = 4x^3{-}3x, \quad T_4(x) = 8x^4{-}8x^2{+}1, \quad T_5(x) = 16x^5{-}20x^3{+}5x\,. \tag{3.7}$$

Pour des dessins voir la figure II.9[3]. Les polynômes de Chebyshev s'annulent en

$$T_n\Big(\cos\Big(\frac{(2k+1)\pi}{2n}\Big)\Big) = 0 \qquad \text{pour } k = 0, 1, \ldots, n-1. \tag{3.8}$$

Revenons maintenant à la question de trouver une division satisfaisant (3.1).
Car le coefficient principal de $T_{n+1}(x)$ est 2^n, nous voyons que

$$\max_{x\in[-1,1]} |(x-x_0)\cdot\ldots\cdot(x-x_n)| \qquad \text{est minimal}$$

si et seulement si $(x-x_0)\cdot\ldots\cdot(x-x_n) = 2^{-n}T_{n+1}(x)$, c.-à-d., par (3.8), si

$$x_k = \cos\Big(\frac{(2k+1)\pi}{2n+2}\Big), \qquad k = 0, 1, \ldots, n \tag{3.9}$$

(points de Chebyshev). Pour répondre à la question (3.1), il faut encore utiliser la translation $x \mapsto \frac{a+b}{2} + \frac{b-a}{2}x$, qui envoie l'intervalle $[-1, 1]$ sur $[a, b]$. On obtient alors

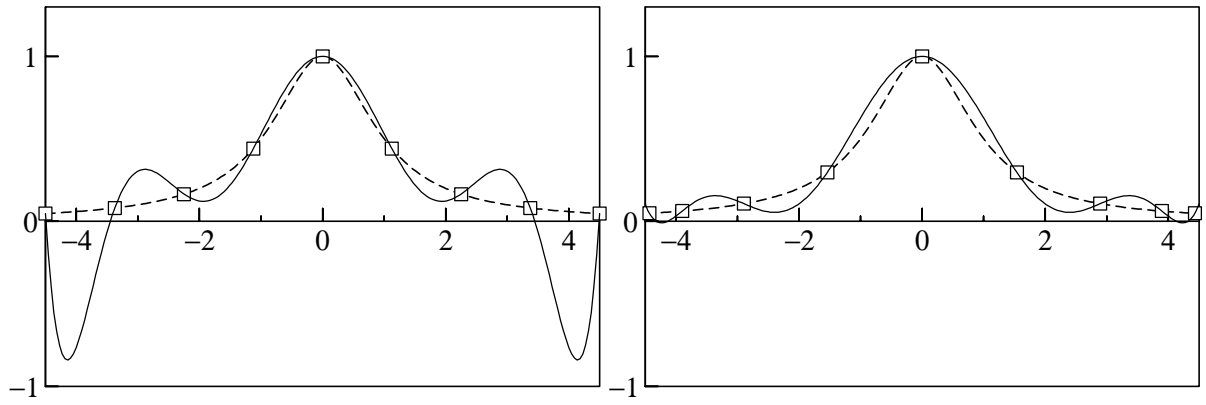


FIG. II.10: Interpolation avec des points équidistants (à gauche) et les points de Chebyshev (à droite)

Théorème 3.3 *L'expression (3.1) est minimale parmi toutes les divisions $\{x_0, x_1, \ldots, x_n\}$ si et seulement si*

$$x_k = \frac{a+b}{2} + \frac{b-a}{2} \cdot \cos\Big(\frac{(2k+1)\pi}{2n+2}\Big), \qquad k = 0, 1, \ldots, n. \tag{3.10}$$

Exemple 3.4 Comme dans la fig. II.4, nous considérons la fonction $f(x) = 1/(1{+}x^2)$ sur l'intervalle $[-4.5, 4.5]$. Dans la fig. II.10, on compare le polynôme d'interpolation basé sur des points équidistants avec celui basé sur les points de Chebyshev. Tout commentaire est superflu !...

[3]Pour une étude des "courbes blanches" dans la fig. II.9 (à droite) voir la page 209 du livre: Th.J. Rivlin, *Chebyshev Polynomials.* 2nd ed., John Wiley & Sons, 1990 [MA 41/36]

II.4 Influence des erreurs d'arrondi sur l'interpolation

Chaque année, les ordinateurs doublent leur performance et, aujourd'hui, en un clin d'oeil, ils exécutent des milliards d'opérations arithmétiques. La *précision* des calculs n'a toutefois pas augmenté ! Par conséquent, en un clin d'oeil, un ordinateur fait aussi *des milliards d'erreurs d'arrondi* !... Il est alors primordial de savoir si, après toutes ces erreurs d'arrondi, les résultats obtenus ont encore la moindre fiabilité. Nous appelons ce sujet *l'étude de la stabilité numérique*.

Encore une mauvaise surprise !... Équipé du merveilleux Théorème de Runge, choisissons la fonction $f(x) = \sin x$ sur l'intervalle $[0, 5]$. Cette fonction n'a aucun pôle fini, donc *la convergence du polynôme d'interpolation est assurée pour tout x !* Et alors, que se passe-t-il en figure II.11 ? Pour expliquer ce phénomène, intéressons nous aux *erreurs d'arrondi*.

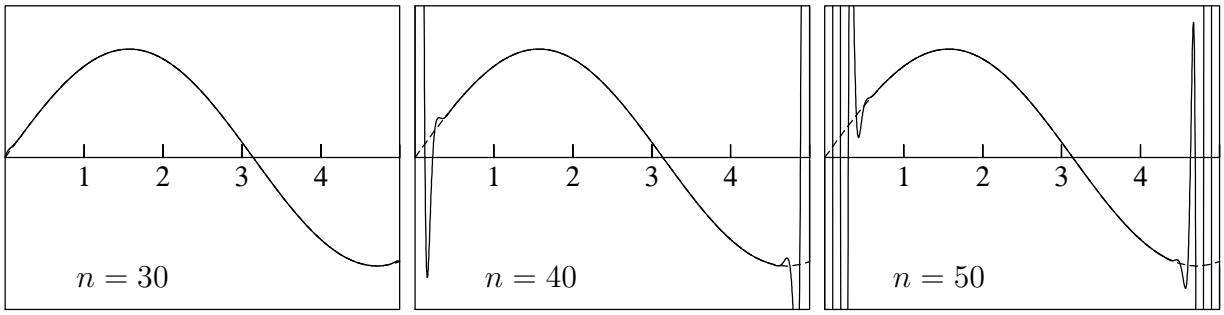


FIG. II.11: Polynôme d'interpolation pour $f(x) = \sin x$ sur $[0, 5]$ à points équidistants

Représentation en virgule flottante. Depuis 30 à 40 ans, la représentation d'un nombre réel sur ordinateur a été standarisée sur 32 bits binaires (64 en double précision). Prenons un lap-top dernier cri, acheté en 2005, et donnons lui les nombres $\frac{1}{3}$, $\frac{2}{3}$, $\frac{4}{3}$, $\frac{8}{3}$ à digérer. Le résultat est donné en Fig. II.12. Rappelons, qu'en base 2 la division $1 : 11$ donne $0.0101010101\ldots$, la division $10 : 11 = 0.101010101\ldots$, la division $100 : 11 = 1.0101010101\ldots$, etc. Nous constatons que le premier chiffre non-nul subit une translation vers la virgule (*virgule flottante*); l'*exposant* de 2 correspondant est stocké dans les bits 3–9. Le bit 2 indique les puissances positives, le bit 1 le signe du nombre. Puis, dans les bits 10–32 suit la *mantisse*, de façon à ce que le premier bit, 1, est supprimé pour gagner une place. Enfin, nous voyons que la suite régulière des bits $1, 0, 1, 0, 1, 0...$ est interrompue au bit 32, à cause d'un arrondissement (correcte) des bits 33,34,35,... qui seraient $1, 0, 1, \ldots$.

Estimation de l'erreur d'arrondi. Pour un exposant p donné, le nombre correspondant x est

$$|x| \geq 2^p \cdot \underset{9}{1} \,.\, \underset{10}{0}\ \underset{11}{0}\ \ldots$$

tandis que l'erreur d'arrondi est

$$|err| \leq 2^p \cdot \underset{9}{0} \,.\, \underset{10}{0}\ \ldots\ \underset{32}{0}\ \underset{33}{0}\ \underset{34}{1}\ \underset{35}{1}\ \underset{36}{1}\ \ldots = 2^p \cdot \underset{9}{0} \,.\, \underset{10}{0}\ \ldots\ \underset{32}{0}\ \underset{33}{1}$$

Estimation importante :

$$\boxed{|err| \leq |x| \cdot 2^{-24} = |x| \cdot eps.} \tag{4.1}$$

En *double précision*, nous disposons de $2 \cdot 32 = 64$ bits, dont 12 sont utilisés pour l'exposant. En résumé

$$\begin{aligned} &\text{REAL} * 4, && eps = 2^{-24} \approx 5.96 \cdot 10^{-8} \\ &\text{REAL} * 8, && eps = 2^{-53} \approx 1.11 \cdot 10^{-16} \\ &\text{REAL} * 16, && eps = 2^{-113} \approx 9.63 \cdot 10^{-35}. \end{aligned}$$

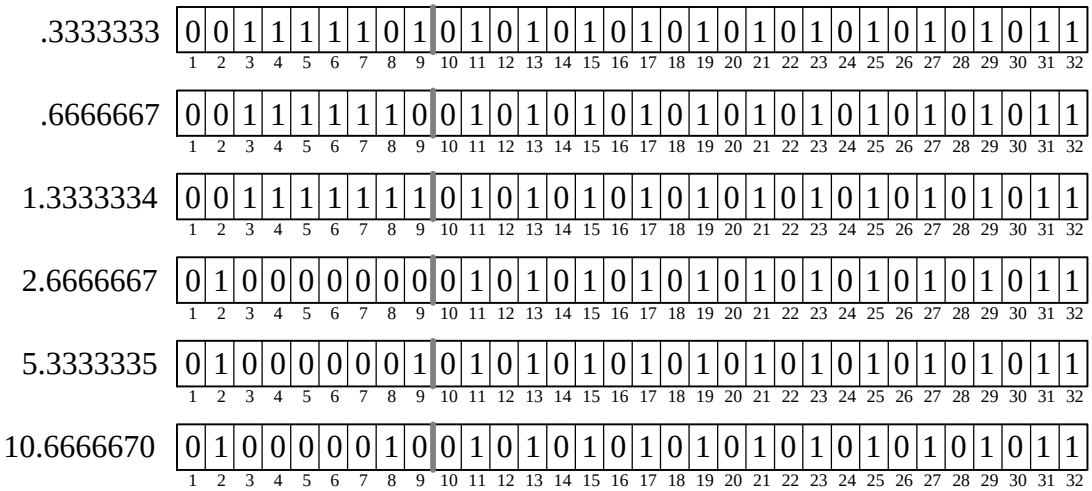


FIG. II.12: Les nombres $\frac{1}{3}$, $\frac{2}{3}$, $\frac{4}{3}$, ... en virgule flottante en REAL*4

Théorème 4.1 *Si on dénote le nombre arrondi d'un* $x \neq 0$ *par* arr (x)*, alors*

$$\frac{|\text{arr}\,(x) - x|}{|x|} \leq eps\,, \tag{4.2}$$

ou

$$arr\,(x) = x(1+\epsilon) \qquad où \qquad |\epsilon| \leq eps\,. \tag{4.3}$$

Ces formules sont la base de toute étude d'erreurs d'arrondi.

Influence des erreurs dans y_i sur le polynôme d'interpolation. Supposons que les données y_i, pour lesquelles on devrait calculer le polynôme d'interpolation $p(x)$, soient erronées,

$$\widehat{y}_i = y_i(1+\epsilon_i) \qquad \text{où} \qquad |\epsilon_i| \leq eps, \tag{4.4}$$

et qu'on calcule en fait avec ces valeurs un polynôme $\widehat{p}(x)$.

La différence de ces polynômes est

$$\widehat{p}(x) - p(x) = \sum_{i=0}^{n} \epsilon_i y_i \ell_i(x)\,,$$

où les $\ell_i(x)$ sont les polynômes de Lagrange (voir Chap. I, formule (1.4) et Fig. II.13). On obtient

$$|\widehat{p}(x) - p(x)| \leq eps \cdot \max_{i=0,\ldots,n} |y_i| \cdot \sum_{i=0}^{n} |\ell_i(x)|. \tag{4.5}$$

La fonction $\sum_{i=0}^{n} |\ell_i(x)|$ décrit l'amplification de l'erreur dans les données (voir figure II.14).

TAB. II.2: Constantes de Lebesgue pour noeuds équidistants

n	10	20	30	40	50	60	80	100
Λ_n	$3.0 \cdot 10^1$	$1.1 \cdot 10^4$	$6.6 \cdot 10^6$	$4.7 \cdot 10^9$	$3.6 \cdot 10^{12}$	$3.0 \cdot 10^{15}$	$2.2 \cdot 10^{21}$	$1.8 \cdot 10^{27}$

Sa valeur maximale

$$\Lambda_n := \max_{x \in [a,b]} \sum_{i=0}^{n} |\ell_i(x)| \tag{4.6}$$

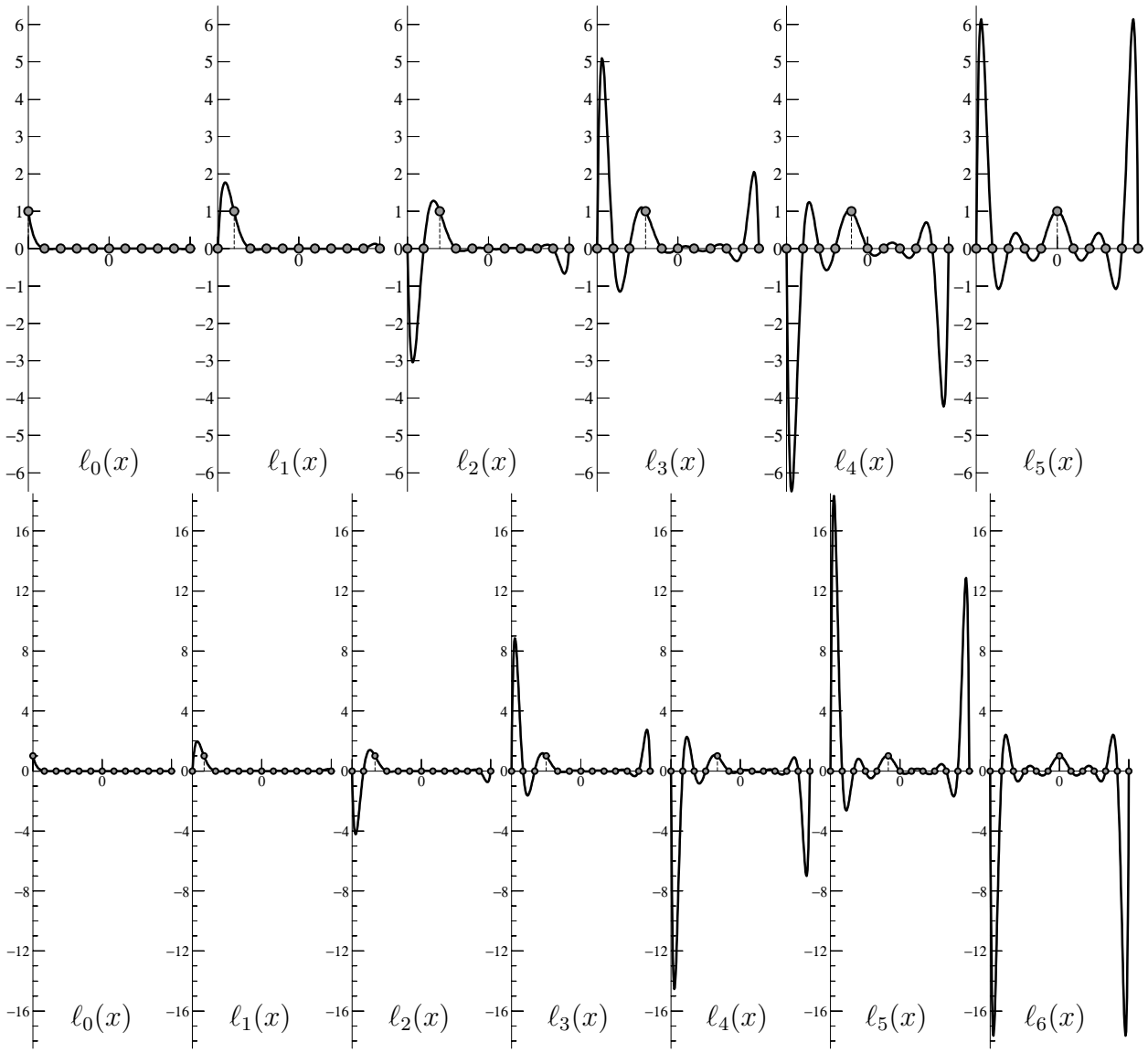


FIG. II.13: Polynômes de Lagrange à points équidistants pour $n = 10$ et $n = 12$

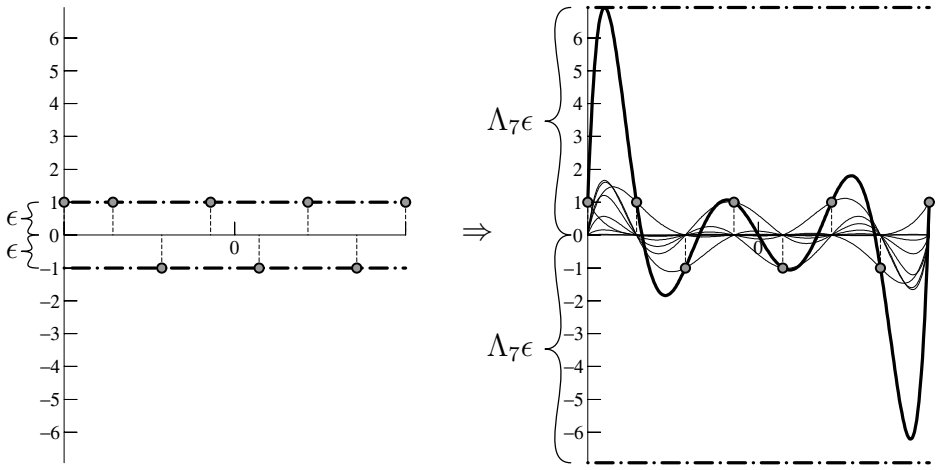


FIG. II.14: Constante de Lebesgue pour $n = 7$

s'appelle la *constante de Lebesgue* [4] associée aux points $x_0, x_1, \ldots, x_n$ et à l'intervalle $[a, b]$. Cette

[4]Pour une raison obscure ; on ne trouve dans les oeuvres de Lebesgue qu'un article général sur l'approximation

constante peut être calculée numériquement (voir Table II.2).

Calcul asymptotique pour points équidistants. Nous voyons, en figure II.13, que les ℓ_i *dont les* x_i *se trouvent au centre de l'intervalle* sont dangereux *vers le bord de l'intervalle*. Essayons donc de trouver une valeur asymptotique pour le premier maximum de $\ell_m(x)$ où $n = 2m$:

$$\ell_m(x) = \frac{x\ (x-1)\ (x-2)}{m(m-1)(m-2)} \cdots \frac{(x-(m-1))}{1} \cdot \frac{(x-(m+1))}{1} \cdots \frac{(x-(n-2))(x-(n-1))(x-n)}{(m-2)\qquad (m-1)\qquad m} \tag{4.7}$$

Nous avons

$$\ell'_m(x) = \ell_m(x)\Big(\frac{1}{x} + \frac{1}{x-1} + \frac{1}{x-2} + \ldots + \frac{1}{x-m+1} + \frac{1}{x-m-1} + \ldots + \frac{1}{x-n}\Big)\,.$$

Donc $\ell'_m(x) = 0$ si

$$\frac{1}{x} = \frac{1}{1-x} + \frac{1}{2-x} + \ldots + \frac{1}{m-1-x} + \frac{1}{m+1-x} + \ldots + \frac{1}{n-x}\,.$$

Car la série harmonique diverge, on voit que $\frac{1}{x} \to \infty$ et $x \to 0$. Ainsi, asymptotiquement,

$$\frac{1}{x} \approx \frac{1}{1} + \frac{1}{2} + \ldots + \frac{1}{m-1} + \frac{1}{m+1} + \ldots + \frac{1}{n} \approx \log n\,.$$

Cela inséré dans (4.7) donne

$$\max|\ell_m(x)| \approx \frac{1}{\log n} \cdot \frac{1\cdot 2\cdot 3\cdots(m-1)m(m+1)\cdots n}{m(m!)^2} \approx 2^n \cdot \frac{2^{3/2}}{\log n \cdot \sqrt{\pi}\cdot n^{3/2}} \tag{4.8}$$

où la célèbre formule asymptotique de Stirling (voir [HW97], §II.10)

$$n! \approx \sqrt{2\pi n}\,\frac{n^n}{e^n}$$

a été utilisée. Le facteur 2^n dans (4.8) montre clairement que chaque augmentation de n par 1 *fait perdre 1 bit de précision* !... Ainsi, avec les quelque 30 bits de mantisse en REAL*4, la limite est atteinte vers $n = 30$ (comme on l'observe en figure II.11).

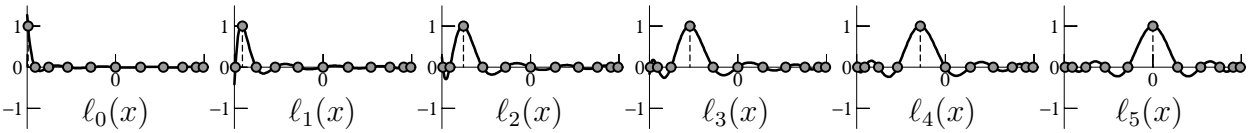


FIG. II.15: Polynômes de Lagrange à points de Chebyshev pour $n = 10$

Points de Chebyshev. Est-il encore nécessaire de mentionner que, pour les points de Chebyshev $x_i = \frac{a+b}{2} + \frac{b-a}{2}\cos(\frac{(2i+1)\pi}{2n+2})$, on n'a pas cette horrible instabilité. Voir Fig. II.15.

(Oeuvres 3, p. 256), où une note en bas de page discute le phénomène de Runge, qui fit fureur à l'époque. Le même phénomène a été indépendamment découvert par E. Borel (1905, *Leçons sur les fonctions de variables réelles et les développements en séries de polynomes*, Chap. IV, p. 74-82), MA 26/49.

II.5 Transformation de Fourier discrète

Commençons par un rappel de la **Transformation CONTINUE de Fourier.**
Origines :

- Problème de la *corde vibrante* (Taylor 1713, Joh. Bernoulli 1727, Euler 1739) ;
- *Théorie du son* (Lagrange 1759) ;
- Théorie *calculi sinuum* d'Euler (1754, *Opera 14*, p. 542), Euler 1777 ;
- *Théorie de la chaleur* (Fourier 1807, 1822).

Une *série trigonométrique* est une série du type

$$y = \frac{a_0}{2} + \sum_{k=1}^{\infty} \Big(a_k \cos kx + b_k \sin kx \Big) \tag{5.1}$$

(voir Figure II.16). Elle représente une fonction $f(x)$ *périodique de période* 2π

$$f(x + 2k\pi) = f(x) \qquad k \in \mathbb{Z} \; . \tag{5.2}$$

Une telle fonction est déterminée par ses valeurs dans un *intervalle de référence,* soit $[0, 2\pi]$, soit $[-\pi, \pi]$. Les sommes partielles s'appellent des *polynômes trigonométriques*.

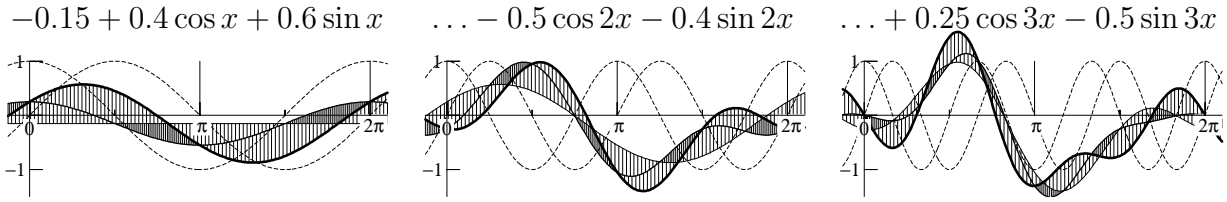


FIG. II.16: Premiers termes d'une série trigonométrique

Un des grand problèmes des mathématiques, élucidé seulement après un lutte d'un bon siècle, était la question si une "quelconque" fonction puisse être décomposée ainsi en "fréquences basses" et "fréquences hautes" (harmoniques ; voir Cours d'Abalyse II).

Représentation complexe. Les formules les plus simples s'obtiennent en passant aux complexes. Grâce à

$$e^{\pm ix} = \cos x \pm i \sin x \; , \qquad \cos x = \frac{e^{ix} + e^{-ix}}{2} \; , \qquad \sin x = \frac{e^{ix} - e^{-ix}}{2i} \tag{5.3}$$

la série (5.1) devient

$$y = \sum_{k=-\infty}^{\infty} c_k e^{ikx} \qquad \text{où} \quad \begin{array}{l} c_k = \frac{1}{2}(a_k - ib_k), \\ c_{-k} = \frac{1}{2}(a_k + ib_k) \end{array} \quad (k \geq 0) \quad \text{ou} \quad \begin{array}{l} a_k = c_k + c_{-k} \\ b_k = i(c_k - c_{-k}). \end{array} \tag{5.4}$$

Orthogonalité. La clé fondamentale permettant le calcul des séries trigonométriques a été découverte par Euler (1777, *Opera* vol. 16, Pars 1, p. 333) :

$$\int_0^{2\pi} e^{-i\ell x} \cdot e^{ikx} \, dx = \int_0^{2\pi} e^{i(k-\ell)x} \, dx = \frac{1}{i(k-\ell)} \, e^{i(k-\ell)x} \Big|_0^{2\pi} = \begin{cases} 0 & \text{si } \ell \neq k \\ 2\pi & \text{si } \ell = k \; . \end{cases} \tag{5.5}$$

Grâce à cette formule, il suffit de multiplier la série (5.4) par $e^{-i\ell x}$ et intégrer terme par terme de 0 à 2π. [5] Tous les termes, sauf un, disparaissent et on obtient

$$c_k = \frac{1}{2\pi}\int_0^{2\pi} f(x)e^{-ikx}\,dx, \qquad (k \in \mathbb{Z}) \tag{5.6}$$

et, par (5.4),

$$a_k = \frac{1}{\pi}\int_0^{2\pi} f(x)\cos kx\,dx, \qquad b_k = \frac{1}{\pi}\int_0^{2\pi} f(x)\sin kx\,dx \qquad (k \geq 0). \tag{5.7}$$

Amplitude et phase. En écrivant $c_k = \frac{1}{2}\,r_k \cdot e^{i\varphi_k}$ on a encore

$$a_k\cos kx + b_k \sin kx = c_k e^{ikx} + \overline{c}_k e^{-ikx} = r_k\,\mathrm{Re}\,e^{ikx+i\varphi_k} = r_k\cos(kx+\varphi_k), \tag{5.8}$$

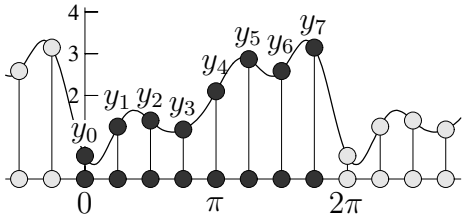
i.e. les deux termes $\cos kx$ et $\sin kx$ se confondent en un terme $\cos kx$ d'*amplitude* $r_k = 2|c_k|$ avec une *phase* φ_k. Ces amplitudes sont normalement représentées dans un *spectrogramme*.

Transformation DISCRÈTE de Fourier.
Origines :

- Interpolation de fonctions périodiques en *astronomie* (Clairaut 1754, Euler 1753, *Opera 14*, p. 463) ;
- Traitement de données périodiques en *météorologie* (Bessel 1828).

Données discrètes.
Supposons que la fonction 2π-périodique $f(x)$ soit seulement connue pour les x de la division équidistante

$$x_j = \frac{2\pi j}{N}, \qquad j = 0, 1, \ldots, N-1$$



et posons $y_j = f(x_j)$. Si nécessaire, on peut prolonger $\{y_j\}$ à une suite N-périodique en posant $y_{j+N} = y_j$ pour tout entier j ($N = 8$ dans la petite figure).
Problème. Trouver des coefficients z_k ($k = 0, 1, 2, \ldots, N-1$) tels que

$$y_n = \sum_{k=0}^{N-1} z_k\, e^{ikx_n} = \sum_{k=0}^{N-1} z_k\, e^{\frac{2i\pi kn}{N}} = \sum_{k=0}^{N-1} z_k\,\omega^{kn} \quad \text{avec} \quad \omega = e^{\frac{2i\pi}{N}} \qquad n = 0, 1, 2, \ldots, N-1\,. \tag{5.9}$$

Interpolation trigonométrique. Comme

$$\omega^N = 1 \qquad \Rightarrow \qquad e^{i(N-k)x_n} = e^{-ikx_n}$$

les termes pour $k = N-1, k = N-2, \ldots$ correspondent aux termes $k = -1, -2 \ldots$ (voir figure II.17). Le problème (5.9) est alors équivalent à rechercher un polynôme trigonométrique

$$p(x) = \frac{1}{2}\Big(z_{-N/2}e^{-iNx/2} + z_{N/2}e^{iNx/2}\Big) + \sum_{|k|<N/2} z_k e^{ikx} =: \sum_{k=-N/2}^{N/2\,\prime} z_k e^{ikx} \tag{5.10}$$

[5] ce qui peut causer des problèmes de légitimité ..., voir le cours d'Analyse IIA ou le cours *Math. pour Info.*, Chap. VI.

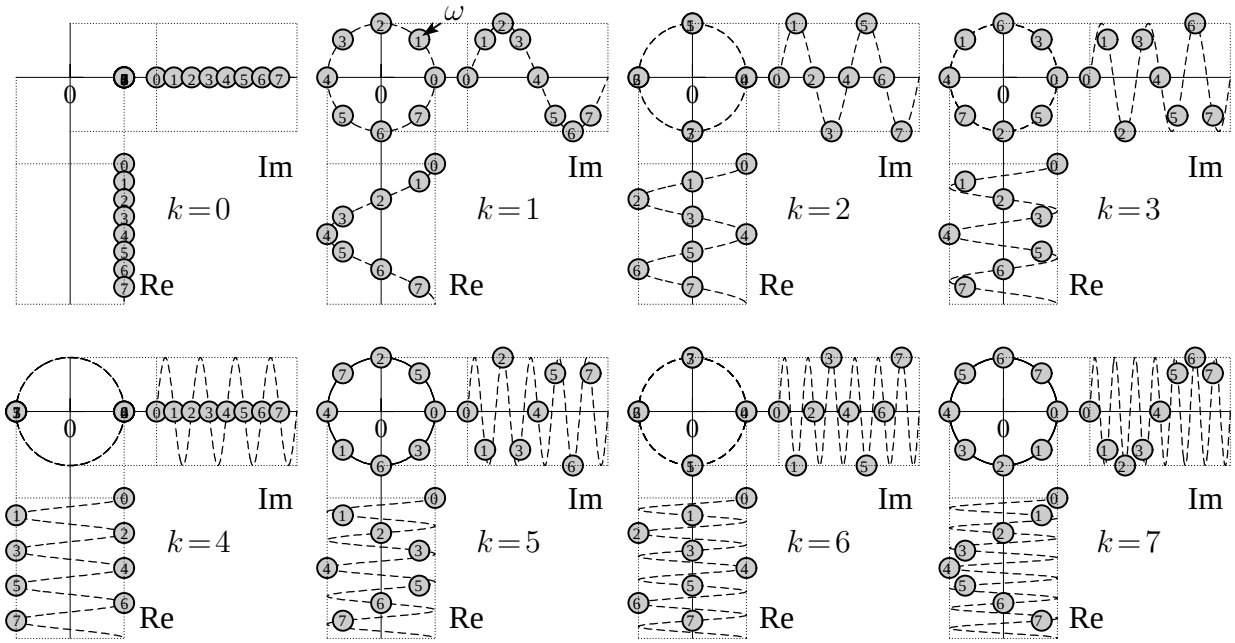


FIG. II.17: Fonctions de base pour la transformation discrète de Fourier

satisfaisant $p(x_n) = y_n$ pour $n = 0, 1, \ldots, N-1$. De plus, si les données y_n sont réelles, nous aurons $z_{N-k} = z_{-k} = \overline{z_k}$. Ainsi, notre polynôme devient avec (5.4)

$$p(x) = \frac{a_0}{2} + \sum_{k=1}^{N/2-1} \Big(a_k \cos kx + b_k \sin kx \Big) + a_{N/2} \cos(N/2)x \qquad \begin{array}{l} a_k = 2\,\mathrm{Re} z_k, \quad a_{N/2} = \mathrm{Re} z_{N/2}, \\ b_k = -2\,\mathrm{Im} z_k \end{array} \tag{5.11}$$

et nous voyons clairement la similitude avec la transformée *continue* de Fourier.

Orthogonalité. Pour la solution du problème, i.e., le calcul des coefficients z_k, nous attendons un miracle semblable à celui de la condition d'orthogonalité ci-dessus (5.5). En fait, nous avons

$$\sum_{n=0}^{N-1} e^{-i\ell x_n} \cdot e^{ikx_n} = \sum_{n=0}^{N-1} \omega^{(k-\ell)n} = \begin{cases} \frac{1-(\omega^{k-\ell})^N}{1-\omega^{k-\ell}} = \frac{1-(\omega^N)^{k-\ell}}{1-\omega^{k-\ell}} = \frac{1-1^{k-\ell}}{1-\omega^{k-\ell}} = 0 & \text{si } \ell \neq k \\ 1+1+\ldots+1 = N & \text{si } \ell = k\,. \end{cases} \tag{5.12}$$

En parfaite analogie avec la preuve ci-dessus pour (5.6), on multiplie la somme (5.9) par $e^{-i\ell x_n}$ et on additionne termes à termes de 0 à $N-1$.[6] Tous les termes, sauf un, disparaissent et on obtient :

Théorème 5.1

$$\begin{array}{ccc} z_k = \dfrac{1}{N} \displaystyle\sum_{n=0}^{N-1} y_n\, e^{-ikx_n} = \frac{1}{N} \sum_{n=0}^{N-1} y_n \omega^{-kn} & \Leftrightarrow & y_n = \displaystyle\sum_{k=0}^{N-1} z_k\, \omega^{nk} \\ z = \mathcal{F}_N(y) & & y = \mathcal{F}_N^{-1}(z) = N \cdot \overline{\mathcal{F}}_N(z)\,. \end{array} \tag{5.13}$$

Relation avec l'algèbre linéaire. Les deux formules (5.13) représentent un système linéaire et son inverse. La matrice est *orthogonale* (ou *hermitienne* comme on dit dans le cas complexe) et jouit de toutes les belles propriétés qu'on connaît en algèbre et en géométrie. En particulier, ces systèmes sont *bien conditionnés* (détails au chapitre IV) ; ici, il *n'y a pas* de méchants phénomènes, tels que celui de Runge ou l'explosion des erreurs d'arrondi (de la section précédente).

[6] sans problème de légitimité cette fois-ci.

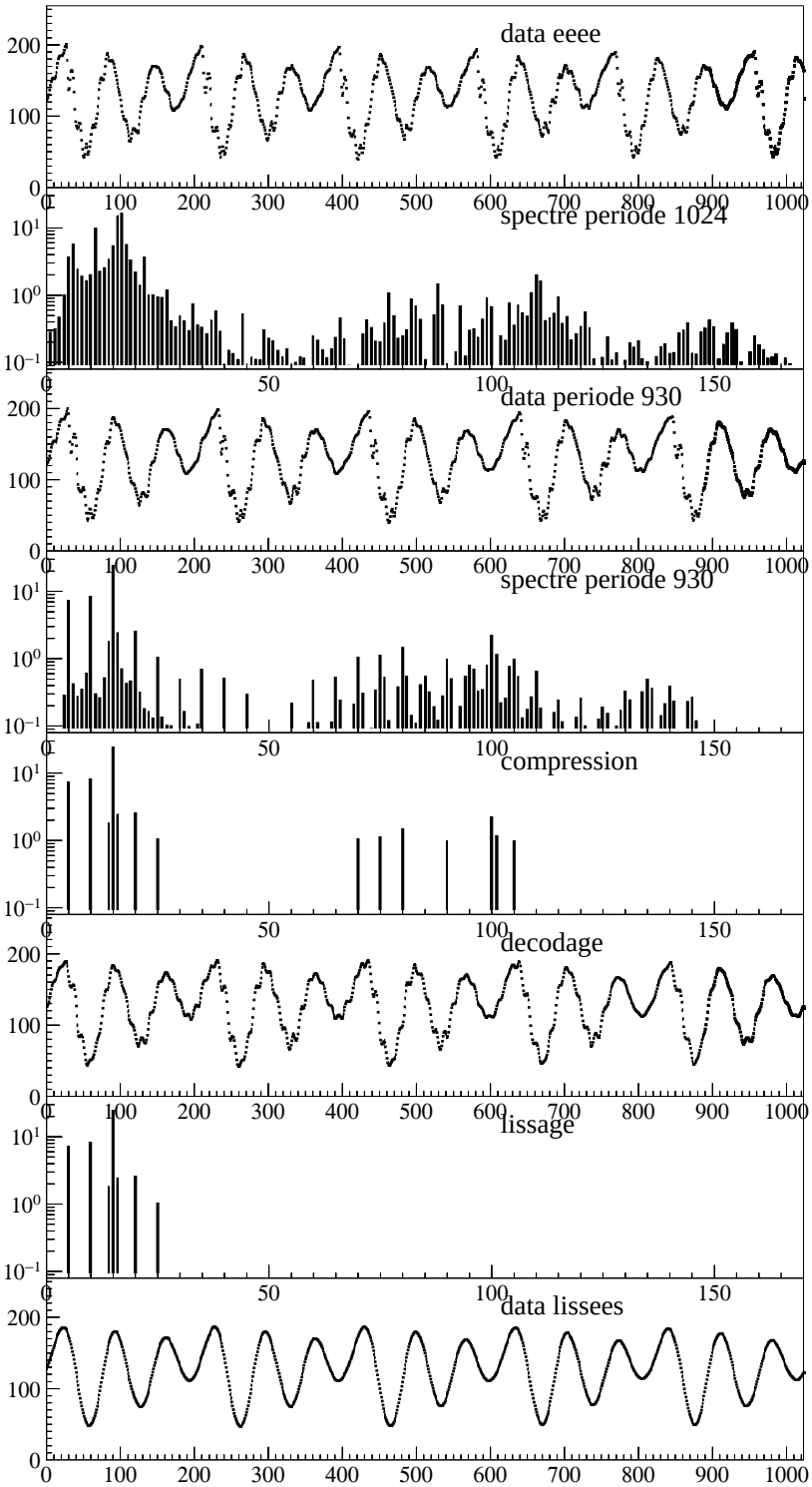


FIG. II.18: Le spectrogramme pour un son ‘eee’ à 118 Hz

Exemple. Dans la première image de la fig. II.18, on voit la digitalisation d'un son "eeee" prononcé par un célèbre mathématicien [7]. Nous partons d'une fenêtre de $N = 1024$ données pour un signal enregistré à 22000 impulsions par seconde ; nous montrons en deuxième image le spectre $|z_k|$. Celui-ci est chaotique. Point de miracle : on applique une théorie pour données périodiques à des données qui ne le sont apparemment *pas*, cela étant dû à la coupure arbitraire après 1024

[7] les historiens de la science ne sont pas unanimes quant à son identité ; il s'agit soit de Martin Hairer, soit de G. Wanner. Une chose est certaine : tous deux étaient présents lors de l'enregistrement...

points. On constate que $y_{N+n} - y_n$ est plutôt minimale près de $N = 930$. On calcule donc 1024 nouvelles valeurs du signal sur une période de 930 (par interpolation linéaire) ; leur transformée de Fourier discrète dans la troisième image est nettement plus claire et montre une fréquence principale ($5 * 22000/930 \approx 118$Hz), ainsi que des harmoniques (multiples de la fréquence principale).

Compression des données. L'idée est de supprimer tous les termes de la série de Fourier dont les coefficients sont en dessous d'un certain seuil (p.ex. 3% du coefficient maximal). Ainsi, la *vraie* information contenue dans le signal ne contient que 14 nombres (au lieu de 1024 ; 4ème image de la figure II.18) et peut toujours être *décompressée* par $\mathcal{F}^{-1}$ (5ème image).

Autres applications. La transformation de Fourier permet de nombreuses autres opérations sur les signaux (filtering, denoising, repair, crossing ...) et un excellent code "Amadeus II" est vivement conseillé (`http://www.hairersoft.com/`).

II.6 Transformation de Fourier rapide (FFT)

"10% d'inspiration et 90% de transpiration."
(Un grand scientifique (Einstein?) sur le secret de ses inspirations)

En 1965, John Tukey, célèbre statisticien à Princeton et aux Bell Labs, demande au jeune programmeur J.W. Cooley d'interrompre son travail pendant deux semaines pour l'aider à programmer une petite idée et à tester son utilité pratique. Les "deux semaines" sont devenues 20 années et la "petite idée" un des algorithmes les plus importants du 20ème siècle. Comme souvent dans pareil cas, on a trouvé plus tard des références antérieures dans la littérature, en particulier un exposé clair de la méthode dans les cours de Runge (voir *Runge-König*, 1924, §66).

Idée. Cherchons à calculer $\mathcal{F}_N^{-1}(z)$ par (5.13). Commençons par la "transpiration", i.e., écrivons cette formule dans tout ses détails pour $N = 4$:

$$\begin{pmatrix} \omega^0 & \omega^0 & \omega^0 & \omega^0 \\ \omega^0 & \omega^1 & \omega^2 & \omega^3 \\ \omega^0 & \omega^2 & \omega^0 & \omega^2 \\ \omega^0 & \omega^3 & \omega^2 & \omega^1 \end{pmatrix} \begin{pmatrix} z_0 \\ z_1 \\ z_2 \\ z_3 \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{pmatrix}$$

Un calcul "brut" de ce produit nécessiterait N^2 multiplications et additions. Essayons d'être plus élégants : l'idée est de permuter les données z (et les colonnes de la matrice), en prenant d'abord les indices *pairs*, puis les indices *impairs* :

$$\left(\begin{array}{cc|cc} \omega^0 & \omega^0 & \omega^0 & \omega^0 \\ \omega^0 & \omega^2 & \omega^1 & \omega^3 \\ \hline \omega^0 & \omega^0 & \omega^2 & \omega^2 \\ \omega^0 & \omega^2 & \omega^3 & \omega^1 \end{array}\right) \left(\begin{array}{c} z_0 \\ z_2 \\ \hline z_1 \\ z_3 \end{array}\right) = \left(\begin{array}{c} y_0 \\ y_1 \\ \hline y_2 \\ y_3 \end{array}\right)$$

Nous voyons que la matrice se décompose en quatre blocs, dont les deux blocs de gauche sont la matrice de la transformée de Fourier pour $\omega \mapsto \omega^2$, i.e., celle avec $N \mapsto N/2$. Comme $\omega^2 = -1$, nous avons encore (en écrivant u_ℓ pour les données paires et v_ℓ pour les données impaires)

$$\left(\begin{array}{cc|cc} \omega^0 & \omega^0 & \omega^0 & \omega^0 \\ \omega^0 & \omega^2 & \omega^1 & \omega^3 \\ \hline \omega^0 & \omega^0 & -\omega^0 & -\omega^0 \\ \omega^0 & \omega^2 & -\omega^1 & -\omega^3 \end{array}\right) \left(\begin{array}{c} u_0 \\ u_1 \\ \hline v_0 \\ v_1 \end{array}\right) = \left(\begin{array}{c} y_0 \\ y_1 \\ \hline y_2 \\ y_3 \end{array}\right)$$

Maintenant, la structure est parfaitement claire et notre “inspiration” montre que

$$\begin{aligned} y_k &= (\mathcal{F}_N^{-1} z)_k = (\mathcal{F}_{N/2}^{-1} u)_k + \omega^k \, (\mathcal{F}_{N/2}^{-1} v)_k \\ y_{k+N/2} &= (\mathcal{F}_N^{-1} z)_{k+N/2} = (\mathcal{F}_{N/2}^{-1} u)_k - \omega^k \, (\mathcal{F}_{N/2}^{-1} v)_k \end{aligned} \tag{6.1}$$

($k = 0, 1, \ldots N/2-1$). Pour la transformée de Fourier *directe*, nous devons tenir compte du facteur $1/N$ et remplacer ω par ω^{-1}. Ainsi, nous arrivons au résultat :

$$\begin{aligned} (\mathcal{F}_N y)_k &= \frac{1}{2} \left((\mathcal{F}_{N/2} u)_k + \omega^{-k} \, (\mathcal{F}_{N/2} v)_k \right) \\ (\mathcal{F}_N y)_{k+N/2} &= \frac{1}{2} \left((\mathcal{F}_{N/2} u)_k - \omega^{-k} \, (\mathcal{F}_{N/2} v)_k \right) . \end{aligned} \tag{6.2}$$

C’est formidable : le même procédé peut être appliqué récursivement aux suites u et v, aussi longtemps que leurs longueurs restent paires. Si l’on suppose que $N = 2^m$, l’algorithme développe toute sa puissance : on peut pousser les simplifications jusqu’au bout. Par exemple, pour $N = 8 = 2^3$ l’algorithme est présenté dans le schéma suivant

$$\begin{array}{l} N \cdot \mathcal{F}_N \begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{pmatrix} \swarrow \\ \qquad\qquad (N/2) \cdot \mathcal{F}_{N/2} \begin{pmatrix} y_0 \\ y_2 \\ y_4 \\ y_6 \end{pmatrix} \swarrow \\ \qquad\qquad\qquad (N/4) \cdot \mathcal{F}_{N/4} \begin{pmatrix} y_0 \\ y_4 \end{pmatrix} \swarrow \\ \qquad\qquad\qquad\qquad \mathcal{F}_{N/8} y_0 = y_0 \\ \qquad\qquad\qquad\qquad \mathcal{F}_{N/8} y_4 = y_4 \\ \qquad\qquad\qquad (N/4) \cdot \mathcal{F}_{N/4} \begin{pmatrix} y_2 \\ y_6 \end{pmatrix} \searrow \\ \qquad\qquad\qquad\qquad \mathcal{F}_{N/8} y_2 = y_2 \\ \qquad\qquad\qquad\qquad \mathcal{F}_{N/8} y_6 = y_6 \\ \qquad\qquad (N/2) \cdot \mathcal{F}_{N/2} \begin{pmatrix} y_1 \\ y_3 \\ y_5 \\ y_7 \end{pmatrix} \swarrow \\ \qquad\qquad\qquad (N/4) \cdot \mathcal{F}_{N/4} \begin{pmatrix} y_1 \\ y_5 \end{pmatrix} \swarrow \\ \qquad\qquad\qquad\qquad \mathcal{F}_{N/8} y_1 = y_1 \\ \qquad\qquad\qquad\qquad \mathcal{F}_{N/8} y_5 = y_5 \\ \qquad\qquad\qquad (N/4) \cdot \mathcal{F}_{N/4} \begin{pmatrix} y_3 \\ y_7 \end{pmatrix} \searrow \\ \qquad\qquad\qquad\qquad \mathcal{F}_{N/8} y_3 = y_3 \\ \qquad\qquad\qquad\qquad \mathcal{F}_{N/8} y_7 = y_7 \end{array} \tag{6.3}$$

La programmation de cet algorithme se fait en deux étapes. D’abord, on met les $\{y_i\}$ dans l’ordre exigé par l’algorithme (6.3), c.-à-d. qu’il faut inverser les bits dans la représentation binaire des indices:

$$\begin{array}{ll} 0 \doteq (0,0,0) & 0 \doteq (0,0,0) \\ 1 \doteq (0,0,1) & 4 \doteq (1,0,0) \\ 2 \doteq (0,1,0) & 2 \doteq (0,1,0) \\ 3 \doteq (0,1,1) & 6 \doteq (1,1,0) \\ 4 \doteq (1,0,0) & 1 \doteq (0,0,1) \\ 5 \doteq (1,0,1) & 5 \doteq (1,0,1) \\ 6 \doteq (1,1,0) & 3 \doteq (0,1,1) \\ 7 \doteq (1,1,1) & 7 \doteq (1,1,1) \end{array} \qquad \longleftrightarrow$$

Après, on effectue les opérations de (6.2) comme indiquées dans le schéma (6.3). Une explication détaillée du code en FORTRAN “ancien” est donnée dans le livre “Numerical Recipies”.[8] Les programmeurs “modernes” profiteront de la récursivité (voir TP).

[8] W.H. Press, B.R. Flannery, S.A. Teukolsky & W.T. Vetterling (1989): *Numerical Recipies. The Art of Scientific Computing* (FORTRAN Version). Cambridge University Press.

TAB. II.3: Comparaison de nombres d'opérations

N	N^2	$N \log_2 N$	quotient
$2^5 = 32$	$\approx 10^3$	160	≈ 6.4
$2^{10} \approx 10^3$	$\approx 10^6$	$\approx 10^4$	≈ 100
$2^{20} \approx 10^6$	$\approx 10^{12}$	$\approx 2 \cdot 10^7$	$\approx 5 \cdot 10^4$

Pour passer d'une colonne à une autre (dans le schéma (6.3)), on a besoin de $N/2$ multiplications complexes et de N additions (ou soustractions). Comme $m = \log_2 N$ passages sont nécessaires, on a

Théorème 6.1 *Pour $N = 2^m$, le calcul de $\mathcal{F}_N y$ peut être effectué en $\frac{N}{2} \log_2 N$ multiplications complexes et $N \log_2 N$ additions complexes.* □

Pour mieux illustrer l'importance de cet algorithme, comparons dans le tableau II.3 le nombre d'opérations nécessaires pour le calcul de $\mathcal{F}_N y$ – avec ou sans FFT.

II.7 Transformée cosinus discrète (DCT) et JPEG

L'algorithme du paragraphe précédent marche très bien si les données ont une certaine périodicité. Si elles ne sont pas périodiques (par exemple pour la compression d'images), on utilise souvent une variante de la transformée de Fourier discrète. Cette variante a en plus l'avantage d'éviter le calcul avec des nombres complexes.

Transformée de Fourier en cosinus. Soit $f(x)$ une fonction continue, définie sur l'intervalle $[0, \pi]$. On la prolonge en une fonction paire par $f(-x) = f(x)$ et en une fonction 2π-périodique par $f(x + 2\pi) = f(x)$. La série de Fourier (5.1) d'une telle fonction contient seulement les termes en cosinus et s'écrit

$$f(x) \;=\; \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos kx. \tag{7.1}$$

Sur l'intervalle $[0, \pi]$, les fonctions $\cos kx$ (pour $k \geq 0$) vérifient la *relation d'orthogonalité*

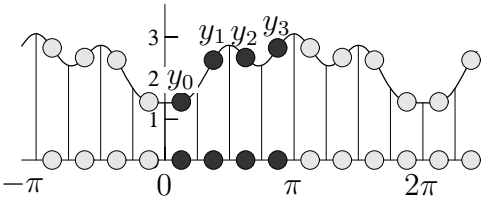
$$\int_0^\pi \cos \ell x \cdot \cos kx \, dx = \frac{1}{2} \int_0^\pi \Big(\cos(\ell + k)x + \cos(\ell - k)x \Big) \, dx = \begin{cases} 0 & \text{si } \; k \neq \ell \\ \pi/2 & \text{si } \; k = \ell \neq 0 \\ \pi & \text{si } \; k = \ell = 0. \end{cases} \tag{7.2}$$

La démarche habituelle (multiplier (7.1) par $\cos \ell x$ et intégrer terme par terme de 0 à π) nous donne

$$a_k = \frac{2}{\pi} \int_0^\pi f(x) \cos kx \, dx. \tag{7.3}$$

Transformée cosinus discrète (DCT). Comme $f(0) \neq f(\pi)$ en général, nous considérons les N points au milieu des sous-intervalles, c.-à-d. les points

$$x_j = \frac{(2j+1)\pi}{2N}, \qquad j = 0, 1, \ldots, N-1$$



et nous posons $y_j = f(x_j)$ ($N = 4$ dans la petite figure). Par analogie avec (7.1), nous exprimons cette suite par (voir la fig. II.19 pour les fonctions de base $\cos kx_j$)

$$y_j \;=\; \frac{z_0}{2} + \sum_{k=1}^{N-1} z_k \cos kx_j. \tag{7.4}$$

Avec la *relation d'orthogonalité discrète* (pour $0 \le k, \ell \le N-1$)

$$\sum_{j=0}^{N-1} \cos \ell x_j \cdot \cos kx_j = \frac{1}{2} \sum_{j=0}^{N-1} \Big(\cos(\ell+k)x_j + \cos(\ell-k)x_j \Big) = \begin{cases} 0 & \text{si } \; k \neq \ell \\ N/2 & \text{si } \; k = \ell \neq 0 \\ N & \text{si } \; k = \ell = 0. \end{cases} \tag{7.5}$$

nous trouvons (multiplier l'équation (7.4) par $\cos \ell x_j$ et additionner de $j = 0$ à $j = N-1$) la *transformée cosinus discrète* (DCT)

$$z_k = \frac{2}{N} \sum_{j=0}^{N-1} y_j \cos kx_j. \tag{7.6}$$

La valeur z_k de (7.6) est le résultat de la règle du point milieu appliquée à l'intégrale dans (7.3).

Transformée cosinus discrète en dimension 2. Une image digitale est donnée par un tableau $\{Y_{i,j}\}$, où i parcourt les pixels verticalement et j horizontalement. La valeur de $Y_{i,j}$ représente le niveau de gris (ou la couleur) du pixel (i, j). Motivé par l'algorithme précédent, nous essayons d'exprimer ces données par

$$Y_{i,j} = \sum_{k=0}^{N-1} \sum_{\ell=0}^{N-1} \widetilde{Z}_{k,\ell} \cos kx_i \, \cos \ell x_j \tag{7.7}$$

où $x_j = (2j+1)\pi/2N$ comme pour la transformée cosinus discrète. Pour compenser le facteur $1/2$ dans (7.4), nous utilisons la notation $\widetilde{Z}_{0,0} = Z_{0,0}/4$, $\widetilde{Z}_{k,0} = Z_{k,0}/2$, $\widetilde{Z}_{0,\ell} = Z_{0,\ell}/2$ et $\widetilde{Z}_{k,\ell} = Z_{k,\ell}$. La relation d'orthogonalité (7.5) nous permet de calculer les $Z_{k,\ell}$ par la formule

$$Z_{k,\ell} = \frac{4}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} Y_{i,j} \cos kx_i \, \cos \ell x_j. \tag{7.8}$$

Les fonctions de base $\cos kx_i \, \cos \ell x_j$ ($k, \ell = 0, \ldots, N-1$) sont illustrées dans la fig. II.20.

JPEG ("Joint Photographic Experts Group"). L'utilisation de cette base dans la compression d'images est due à Ahmed, Natarajan et Rao (IEEE 1974) [9]. On décompose l'image entière en blocs de 8×8 pixels et on calcule pour chaque bloc la transformée cosinus discrète (7.8). Les coefficients $Z_{k,\ell}$ sont alors quantifiés et ceux qui sont en-dessous d'un seuil sont remplacés par zéro. Pour des images contenant des parties uniformes (par exemple : ciel bleu), seulement deux ou trois coefficients vont rester ce qui donne un important facteur de compression.

[9] Pour des explications plus détaillées et références voir *JPEG still image data compression standard* par W. B. Pennebaker et J. L. Mitchell, New York, 1992. (Merci à Kyle Granger pour ces renseignements).

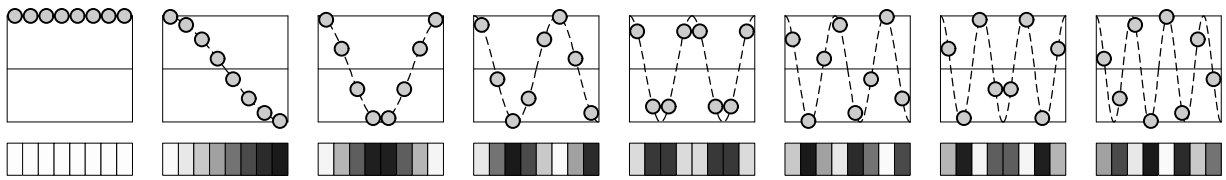


FIG. II.19: Fonctions de base pour la transformée cosinus discrète ($N = 8$)

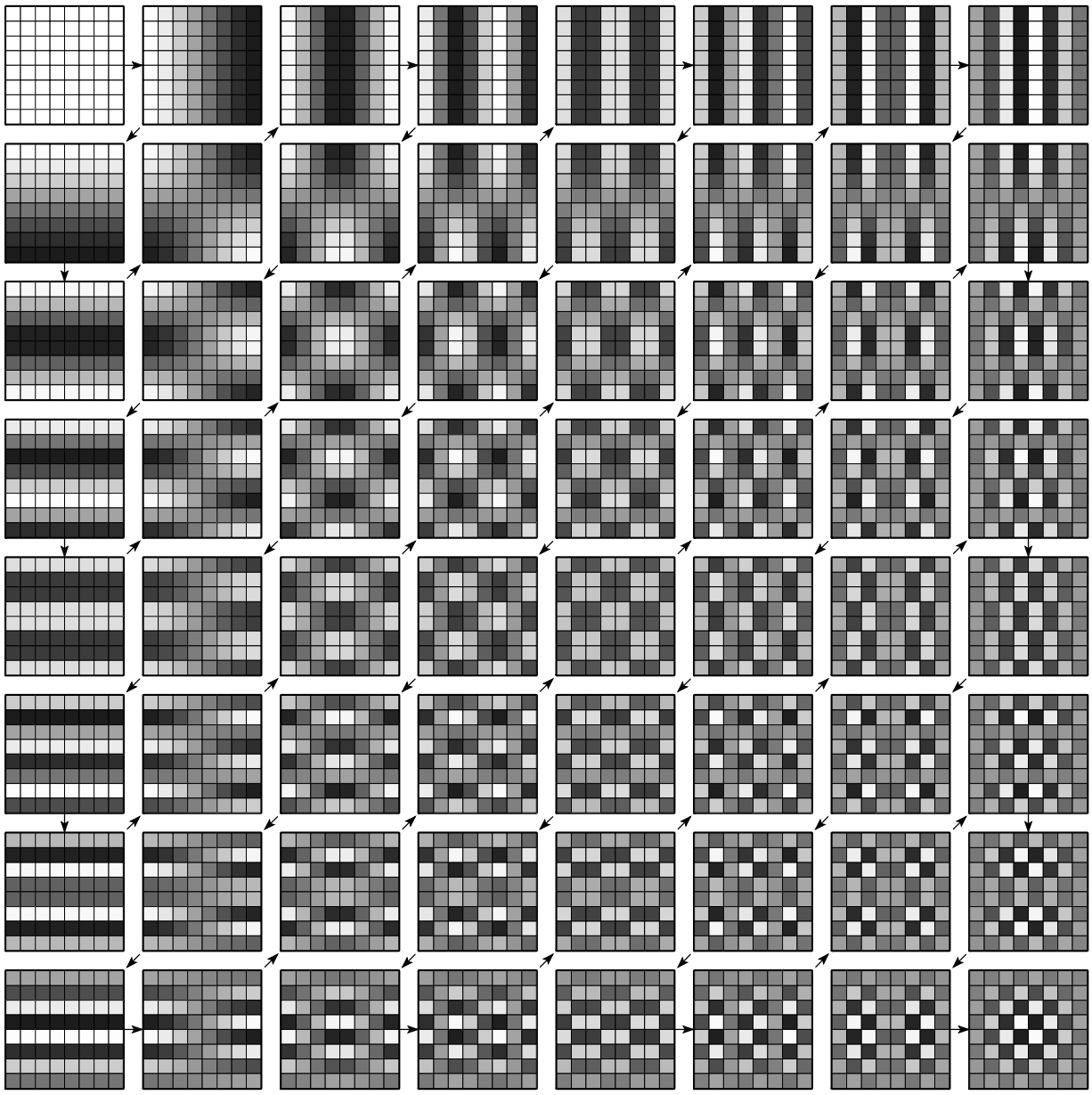
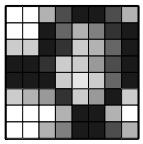


FIG. II.20: Fonctions de base pour la transformée cosinus discrète en 2 dimensions (8×8 pixels)

La reconstruction de l'image à droite (un conglomérat entre Astérix, Mickey Mouse et Donald Duck, dessiné par un grand artiste) est illustrée dans la fig. II.21. Le premier dessin (en haut à gauche) correspond au premier terme de la somme (7.7). En rajoutant un terme après l'autre, et en suivant un chemin astucieux en zig-zag, la reconstruction est démontrée.



II.8 Interpolation par fonctions spline

Le mot "spline" (anglais) signifie "languette élastique". On s'intéresse à la courbe décrite par une languette forcée de passer par un nombre fini de points donnés (disons par (x_i, y_i) pour $i = 0, 1, \ldots, n$). La fig. II.22 montre le spline passant par les mêmes données que pour la fig. II.1 (pour pouvoir le comparer avec le polynôme d'interpolation).

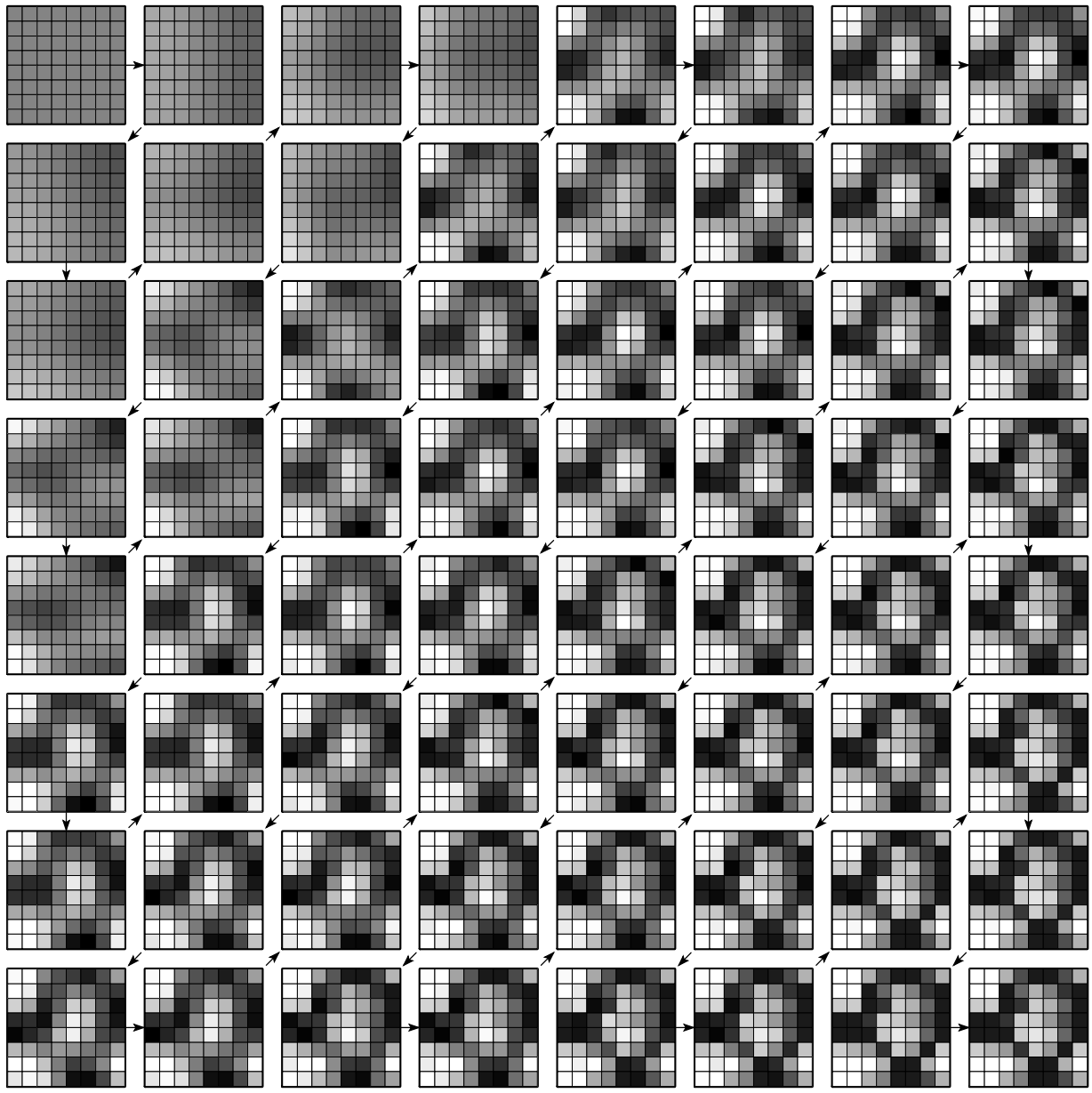


FIG. II.21: Une image toujours mieux reconstituée en zig-zag par JPEG

La théorie des splines a été développée dans les années 1950 et '60 par I.J. Schoenberg pour servir au calcul scientifique (approximations, intégrales, équations différentielles) ; de nos jours elle est costamment appliquée pour la représentation de courbes (voir figure II.23) et surfaces en Computer Graphics.

Mathématiquement, ce problème peut être formulé comme suit: on cherche une fonction $s : [a, b] \to I\!R$ ($a = x_0$, $b = x_n$) satisfaisant

(S1) $s(x_i) = y_i$ pour $i = 0, 1, \ldots, n$;

(S2) $s(x)$ est 2 fois continûment différentiable;

(S3) $\displaystyle\int_a^b (s''(x))^2 \, dx \to \min.$

L'intégrale dans (S3) représente l'énergie de la languette déformée qui, par le principe de Maupertius, est supposée minimale.

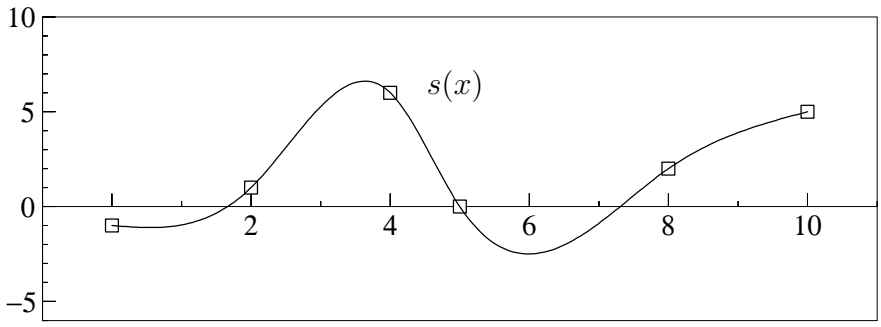


FIG. II.22: Spline cubique (à comparer avec fig. II.1)

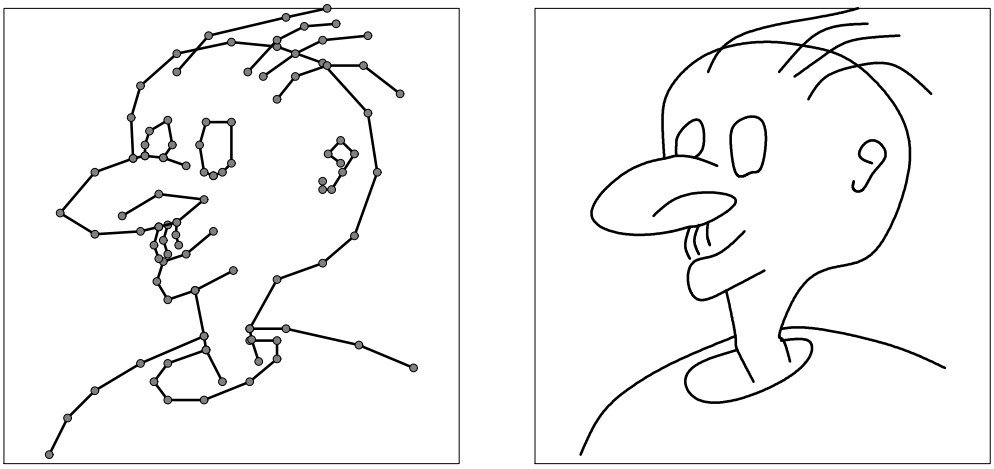


FIG. II.23: Un dessin en zig-zag (à gauche) et en splines (à droite)

Théorème 8.1 *Soit $a = x_0 < x_1 < \ldots < x_n = b$ une division donnée, $s : [a,b] \to I\!R$ et $f : [a,b] \to I\!R$ deux fonctions qui vérifient (S1) et (S2). Supposons que*

$$s''(b)\Big(f'(b) - s'(b)\Big) = s''(a)\Big(f'(a) - s'(a)\Big) \tag{8.1}$$

et que $s(x)$ soit un polynôme de degré 3 sur chaque sous-intervalle $[x_{i-1}, x_i]$. Alors, on a

$$\int_a^b (s''(x))^2\,dx \le \int_a^b (f''(x))^2\,dx. \tag{8.2}$$

Démonstration. Cherchons des conditions suffisantes pour $s(x)$ afin de satisfaire (S3). Pour ceci nous considérons des fonctions voisines $f(x) = s(x) + \epsilon h(x)$ où $\epsilon \in I\!R$ et $h(x)$ est de classe $\mathcal{C}^2$ et vérifie

$$h(x_i) = 0 \qquad \text{pour} \quad i = 0, 1, \ldots, n. \tag{8.3}$$

Chaque fonction $f(x)$ satisfaisant (S1) et (S2) peut être obtenue de cette manière. La condition (S3) s'écrit alors

$$\begin{aligned}\int_a^b (s''(x))^2\,dx &\le \int_a^b (s''(x) + \epsilon h''(x))^2\,dx \\ &= \int_a^b (s''(x))^2\,dx + 2\epsilon \int_a^b s''(x)h''(x)\,dx + \epsilon^2 \int_a^b (h''(x))^2\,dx\end{aligned}$$

(pour tout $\epsilon \in I\!R$), ce qui est équivalent à

$$\int_a^b s''(x)h''(x)\,dx = 0 \tag{8.4}$$

pout tout $h \in \mathcal{C}^2$ vérifiant (8.3). En supposant $s(x)$ 3 fois différentiable sur chaque sous-intervalle de la division, on obtient par intégration par parties que

$$s''(x)h'(x)\Big|_a^b - \int_a^b s'''(x)h'(x)\,dx = 0. \tag{8.5}$$

L'hypothèse (8.1) implique que la première expression de (8.5) est nulle. Comme $s'''(x)$ est constant sur (x_{i-1}, x_i), disons égal à α_i, la deuxième expression de (8.5) devient

$$\int_a^b s'''(x)h'(x)\,dx = \sum_{i=1}^{n} \alpha_i \int_{x_{i-1}}^{x_i} h'(x)\,dx = \sum_{i=0}^{n} \alpha_i (h(x_i) - h(x_{i-1})) = 0$$

par (8.3). Ainsi, (8.4) et par conséquent (8.2) aussi sont vérifiés. □

Définition 8.2 *Soit $a = x_0 < x_1 < \ldots < x_n = b$ une division de $[a, b]$. Une fonction $s \in \mathcal{C}^2[a, b]$ s'appelle spline (cubique) si, sur chaque intervalle $[x_{i-1}, x_i]$, elle est un polynôme de degré 3.*

Pour satisfaire la condition (8.1), on les possibilités suivantes sont les plus utiles:

- *spline naturel:* on suppose que

$$s''(a) = 0 \qquad \text{et} \qquad s''(b) = 0. \tag{8.6}$$

- *spline périodique:* on suppose que

$$s'(a) = s'(b) \qquad \text{et} \qquad s''(a) = s''(b). \tag{8.7}$$

Calcul du spline interpolant.

Nous supposons les points x_i équidistants de distance 1. Pour le cas plus général, les idées sont les mêmes, mais les calculs sont plus compliqués. Nous utilisons ici une idée de Schoenberg (1946) et L.L. Schumaker (1969). Une autre approche (utilisant l'intérpolation d'Hermite) sera l'objet d'un exercice.

1ère idée: Si nous developpons un spline $s(x)$ dans le voisinage de x_i en puissances de $(x - x_i)$:

$$s(x) = a + b(x - x_i) + c(x - x_i)^2 + d(x - x_i)^3,$$

les coefficients a, b, c représentent les dérivées d'ordre $0, 1,$ et 2; elles sont donc les mêmes à droite et à gauche de x_i. La différence d'un spline à droite et à gauche de x_i est donc un multiple de $(x - x_i)^3$. Si nous introduisons la fonction

$$(x - x_i)_+^3 = \begin{cases} (x - x_i)^3 & \text{si } x \geq x_i, \\ 0 & \text{si } x \leq x_i; \end{cases} \tag{8.8}$$

alors un spline aura comme base les fonctions (voir première colonne de Fig. II.24)

$$(x - x_0)_+^3, \;\; (x - x_1)_+^3, \;\; (x - x_2)_+^3, \;\; \ldots$$

Inutile de dire que cette base serait très mal conditionnée.

Idée des B-splines. Il vaut donc mieux *de prendre les différences* de ces fonctions. Et puisque les 4èmes différences d'un polynôme de degré 3 sont nulles, nous avons la bonne surprise que les Δ^4

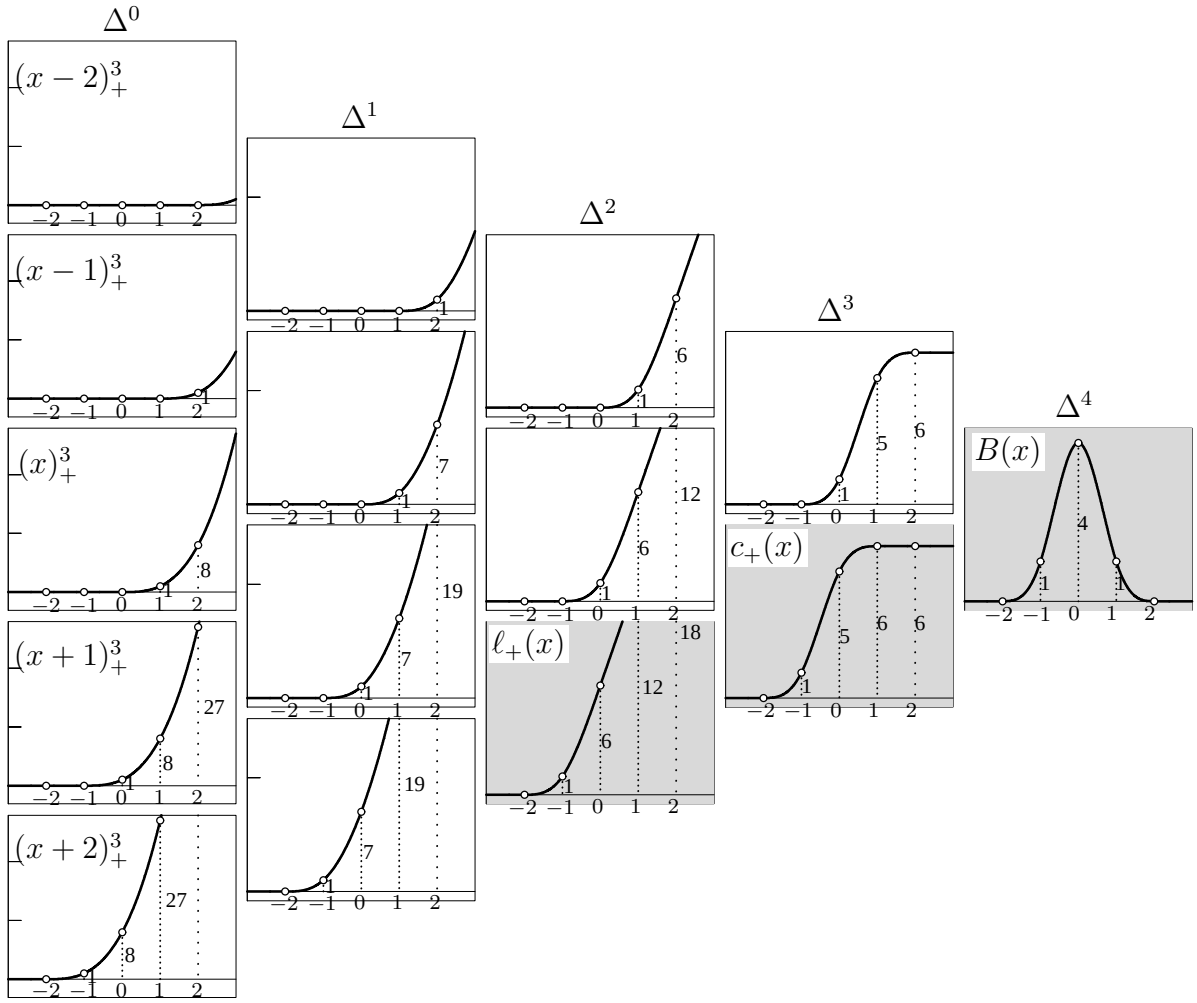


FIG. II.24: Differences successives des fonctions $(x - x_i)_+^3$.

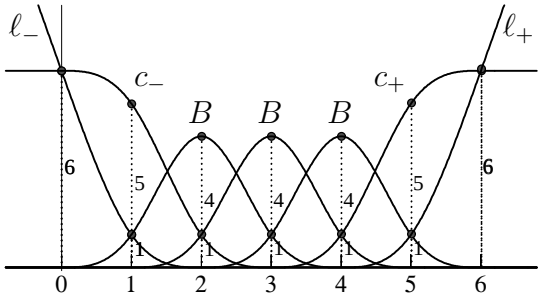
de $(x-x_i)_+^3$ *sont à support compact* (voir Fig. II.24). Notre base est donc composé des translations de

$$B(x) = \Delta^4 (x-2)_+^3 = \begin{cases} (x+2)^3 & \text{si } -2 \le x \le -1, \\ (x+2)^3 - 4(x+1)^3 & \text{si } -1 \le x \le 0, \\ (x+2)^3 - 4(x+1)^3 + 6x^3 & \text{si } 0 \le x \le 1, \\ (x+2)^3 - 4(x+1)^3 + 6x^3 - 4(x-1)^3 & \text{si } 1 \le x \le 2, \\ 0 & \text{sinon} \end{cases} \tag{8.9}$$

Petite difficulté. Les B-splines de (8.9) ne reproduisent pas, pour les splines naturels, les polynômes de degré 1 en dehors des points $0, 1, \ldots, n$. Nous allons donc rajouter à notre base quelques deuxièmes et troisièmes différences (en gris dans la Fig. II.24; "ℓ" veut dire "linéaire", "c" veut

dire “constant” en dehors des x_i ; $c_-(x) = c_+(-x)$ et $\ell_-(x) = \ell_+(-x)$ s’obtiennent par symétrie)

$$s(x) = \alpha_0\ell_-(x) + \alpha_1 c_-(x-1) + \alpha_2 B(x-2) + \ldots$$
$$+\alpha_{n-2}B(x-(n-2)) + \alpha_{n-1}c_+(x-(n-1)) + \alpha_n\ell_+(x-n)$$



(8.10)

Calcul du spline intérpolant. Les conditions $s(i) = y_i$, en introduisant les valeurs des B, c, et ℓ, deviennent des équations linéaires pour les coefficients α_i, que voici (pour $n = 6$)

$$\begin{pmatrix} 6 & 6 & & & & & \\ 1 & 5 & 1 & & & & \\ & 1 & 4 & 1 & & & \\ & & 1 & 4 & 1 & & \\ & & & 1 & 4 & 1 & \\ & & & & 1 & 5 & 1 \\ & & & & & 6 & 6 \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{pmatrix} \tag{8.11}$$

Remarque. Pour le *spline périodique*, il n’y a que des $B(x)$ tout au tour et la matrice dans (8.11) n’a que des $1, 4, 1$ avec deux 1 supplémentaires dans les coins $n, 0$ et $0, n$.

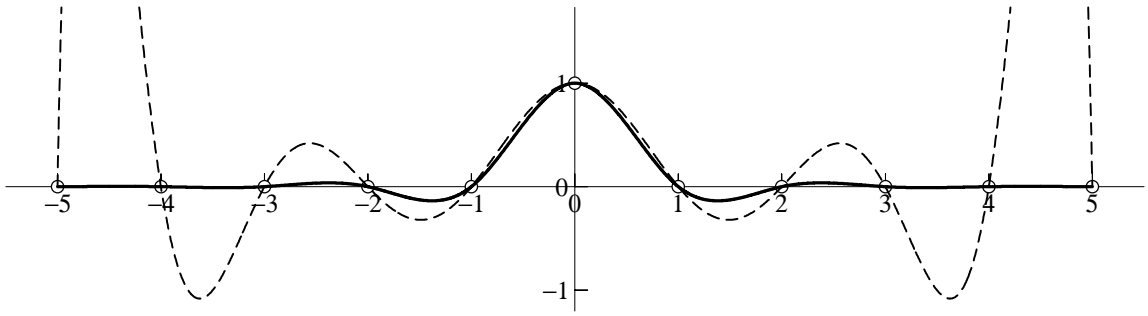


FIG. II.25: Spline “de Lagrange” comparé au polynôme de Lagrange.

Phénomène de Runge, erreurs d’arrondi, splines “de Lagrange”. Nous allons procéder à l’élimination des éléments sous la diagonale du système (8.11):

$$\begin{pmatrix} 6 & 6 \\ 1 & 5 \end{pmatrix} \mapsto \begin{pmatrix} 6 & 6 \\ 0 & 5-\frac{6}{6} \end{pmatrix} \quad \begin{pmatrix} 4 & 1 \\ 1 & 4 \end{pmatrix} \mapsto \begin{pmatrix} 4 & 1 \\ 0 & 4-\frac{1}{4} \end{pmatrix} \quad \begin{pmatrix} 3.75 & 1 \\ 1 & 4 \end{pmatrix} \mapsto \begin{pmatrix} 3.75 & 1 \\ 0 & 4-\frac{1}{3.75} \end{pmatrix}$$

Nous voyons que les éléments de la diagonale convergent vite vers une valeur κ qui satisfait

$$\kappa = 4 - \frac{1}{\kappa} \qquad \text{donc} \qquad \kappa = 2 + \sqrt{3} = 3.73205...$$

Si maintenant les premiers y_i sont toutes nulles, nous obténons pour les α_i approximativement

$$\kappa\alpha_i + \alpha_{i+1} = 0 \qquad \text{ou} \qquad \alpha_i = -\frac{1}{\kappa}\,\alpha_{i+1} = -0.268\,\alpha_{i+1}$$

i.e., en s'éloignant de la position où $y_j = 1$, les α_i tendent vers zéro comme une suite géométrique à rapport -0.268 (une découverte de M. Powell 1966 ; voir illustration en Fig. II.25). Pour les splines, *il n'y a donc ni Phénomène de Runge, ni influence catastrophique des erreurs d'arrondi*.

II.9 Ondelettes ('One Lecture on Wavelets')

Origines.

- Base de Haar (Math. Annalen vol. 69, 1910) ;
- Morlet 1982/83 (Analyse d'ondes sismiques) ;
- Morlet–Grossmann 1984, Y. Meyer 1985 (Wavelets de Meyer ; "1984" : l'*année de naissance* des ondelettes . . .) ;
- Mallat 1987 (Analyse multirésolution ; "1987" : l'*annus mirabilis*) ;
- Daubechies 1987 (Wavelets à support compact) ;
- Lemarié–Battle 1987 (Wavelets spline) ;
- Holschneider 1988, Daubechies 1992 (Fast Wavelet Transform).

Littérature.

1. I. Daubechies, *Ten Lectures on Wavelets* 1992 ;
2. Y. Meyer, *Ondelettes* 1990 ;
3. P.G. Lemarié, *Les Ondelettes en 1989*, (Lecture Notes 1438, de nombreuses applications) ;
4. Louis–Maass–Rieder, *Wavelets* (en allemand) 1994.

Motivation. L'analyse de Fourier est basée sur les fonctions

e^{ikx} =

Elle détermine les fréquences, mais ne *localise pas* bien les sons. Prenons pour exemple trois notes de Beethoven (sol - sol - mibémol) en figure II.26. L'analyse de Fourier montre bien les deux fréquences dominantes (en rapport 5 sur 4), mais la *localisation* de ces notes est brouillée dans le chaos des hautes fréquences.

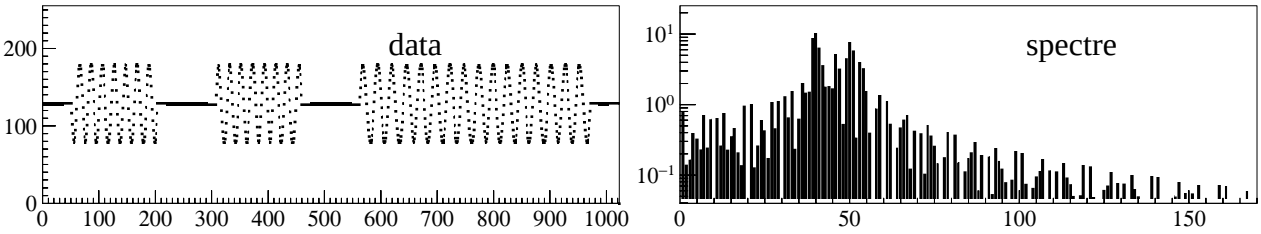


FIG. II.26: Trois notes de la 5ème de Beethoven, la transformée de Fourier et le résultat d'une trop forte compression.

Exemple : la base de Haar. Après le 'désastre' des théorèmes de convergence pour séries de Fourier et fonctions continues (fausse preuve de Cauchy, correction de Dirichlet, contre-exemples

de Dubois-Reymond, H.A. Schwarz, Lebesgue et Fejér, voir cours d'Analyse II), Hilbert pose à son étudiant A. Haar le problème suivant : trouver (enfin) une base de fonctions orthogonales, où la convergence est *assurée* pour toute fonction continue. Le résultat de ces recherches est la "base de Haar" (1910, voir figure II.27) ; la convergence (même uniforme) n'est pas trop difficile à démontrer (voir exercices).

Mais : la base de Haar est composée de fonctions *discontinues* et la convergence est lente. Donc, nous ne sommes pas au bout de nos peines. Avant de progresser, nous constatons :

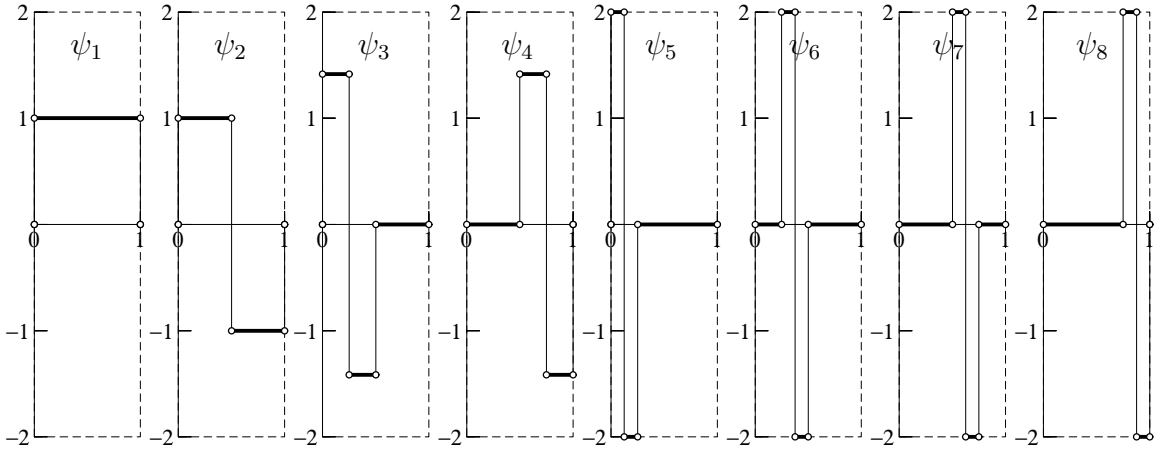


FIG. II.27: La base de Haar

Lemme 9.1 *Soit* $\psi(x) = 1$ *pour* $0 < x < \frac{1}{2}$ *et* $= -1$ *pour* $\frac{1}{2} < x < 1$ *la fonction* ψ_2 *de la base de Haar, alors* **toutes** *les autres fonctions de cette base (à l'exception de* ψ_1*) sont de la forme*

$$\psi_{m,n}(x) = 2^{m/2} \cdot \psi(2^m x - n) \qquad (m = 1, 2, 3, \ldots, \quad n = 0, \pm 1, \pm 2, \ldots). \tag{9.1}$$

Une telle fonction s'appelle "ondelette mère". Les valeurs de n *représentent des translations en* x *et servent à localiser un événement, les valeurs de* m *correspondent aux fréquences comme précédemment.*

GRAND PROBLÈME. Trouver une **ondelette mère** $\psi(x)$ (continue, différentiable, à support compact ...), pour laquelle **toutes** les fonctions dans (9.1) (pour tout n et pour tout m) forment une **base orthogonale**.

Les diverses réponses à cette question (ondelettes de Meyer, ondelettes de Daubechies ...) nécessitent des moyens lourds d'analyse fonctionnelle (transformée continue de Fourier), seules les ondelettes 'spline' de Lemarié–Battle sont, pour nous, relativement faciles à comprendre.

Comme c'est souvent le cas, il est conseillé de s'attaquer d'abord à un problème plus facile :

Problème plus facile. Trouver une fonction $\varphi(x)$ (qu'on appellera le **père de l'ondelette**) orthogonale à tous ses fonctions translatées

$$\int_{-\infty}^{\infty} \varphi(x-n) \cdot \varphi(x-m)\, dx = 0 \qquad \text{pour tout } n, m \in \mathbb{Z},\ n \neq m. \tag{9.2}$$

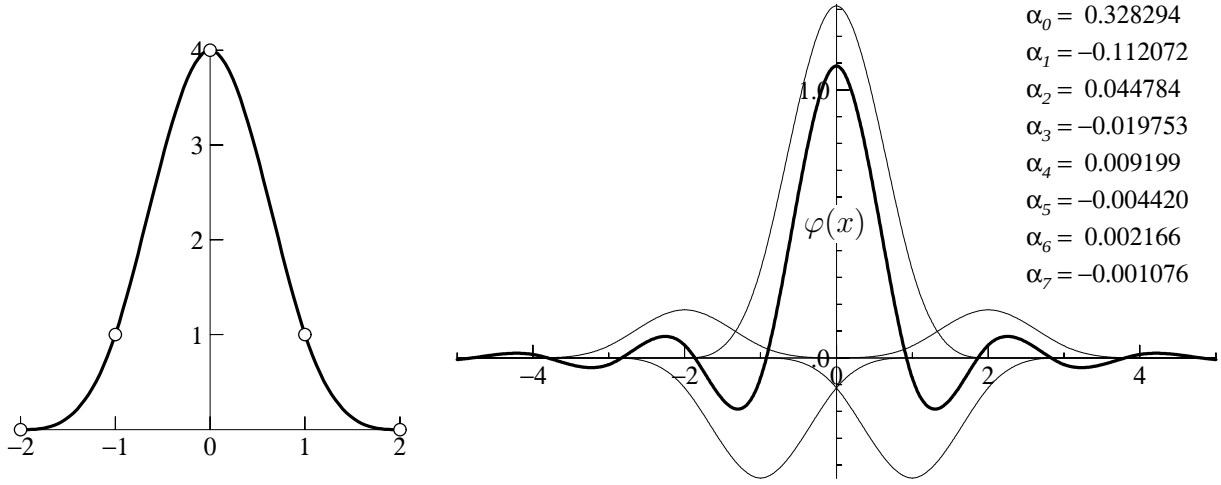


FIG. II.28: B-spline (à gauche), *père* orthogonal de l'ondelette de Battle–Lemarié (à droite).

Calcul du 'père spline' de l'ondelette. Nous partons d'un spline à support compact, le B-spline. Au lieu de la condition (9.2), cette fonction satisfait

$$\int_{-\infty}^{\infty} B(x-n)\cdot B(x-m)\,dx = \begin{cases} d_0 = \frac{604}{35} & \text{si } n=m, \\ d_1 = \frac{1191}{140} & \text{si } |n-m|=1, \\ d_2 = \frac{6}{7} & \text{si } |n-m|=2, \\ d_3 = \frac{1}{140} & \text{si } |n-m|=3, \\ 0 & \text{sinon} \end{cases} \tag{9.3}$$

Nous posons alors

$$\varphi(x) = \alpha_0 B(x) + \alpha_1(B(x-1)+B(x+1)) + \alpha_2(B(x-2)+B(x+2)) + \ldots = \sum_k \alpha_k B(x-k) \tag{9.4}$$

où nous supposons pour l'instant $\alpha_0 = 1$. Les conditions d'orthogonalité (9.2) deviennent alors, grâce à (9.3),

$$d_0 S_\ell + d_1(S_{\ell-1}+S_{\ell+1}) + d_2(S_{\ell-2}+S_{\ell+2}) + d_3(S_{\ell-3}+S_{\ell+3}) = 0 \tag{9.5}$$

où

$$S_\ell = \frac{1}{2}\sum_k \alpha_k \alpha_{k-\ell} = \alpha_\ell + S'_\ell$$

où S'_ℓ est la même somme sans les termes $k=0$ et $k=\ell$. Nous résolvons alors les conditions (9.5) par calcul numérique brutal en posant itérativement

$$\alpha_\ell := -S'_\ell - \frac{d_1}{d_0}(S_{\ell-1}+S_{\ell+1}) - \frac{d_2}{d_0}(S_{\ell-2}+S_{\ell+2}) - \frac{d_3}{d_0}(S_{\ell-3}+S_{\ell+3}) \qquad \ell = 1,2,3,\ldots.$$

Après la convergence de cet algorithme en une trentaine d'itérations, nous normalisons les α pour avoir $\int_{-\infty}^{\infty}(\varphi(x))^2\,dx = 1$ et nous arrivons aux valeurs présentées en figure II.28 à droite et à la fonction φ recherchée.

Analyse multi-échelles.

Cette analyse (Mallat 1987) est la **clef** pour trouver finalement l'ondelette mère. L'idée est de raffiner pas à pas la résolution par un facteur 2 en remplaçant $x \mapsto 2x$. Commençons par le lemme suivant :

Lemme 9.2 *L'espace V^0 engendré par*

$$\dots\ B(x+3),\ B(x+2),\ B(x+1),\ B(x),\ B(x-1),\ B(x-2),\ B(x-3),\ \dots$$

est un sous-espace de V_1 engendré par

$$\dots\ B(2x+3),\ B(2x+2),\ B(2x+1),\ B(2x),\ B(2x-1),\ B(2x-2),\ B(2x-3),\ \dots$$

et nous avons

$$B(x) = \frac{1}{8}B(2x+2) + \frac{1}{2}B(2x+1) + \frac{3}{4}B(2x) + \frac{1}{2}B(2x-1) + \frac{1}{8}B(2x-2) = \sum_{k=-2}^{2} c_k B(2x-k). \tag{9.6}$$

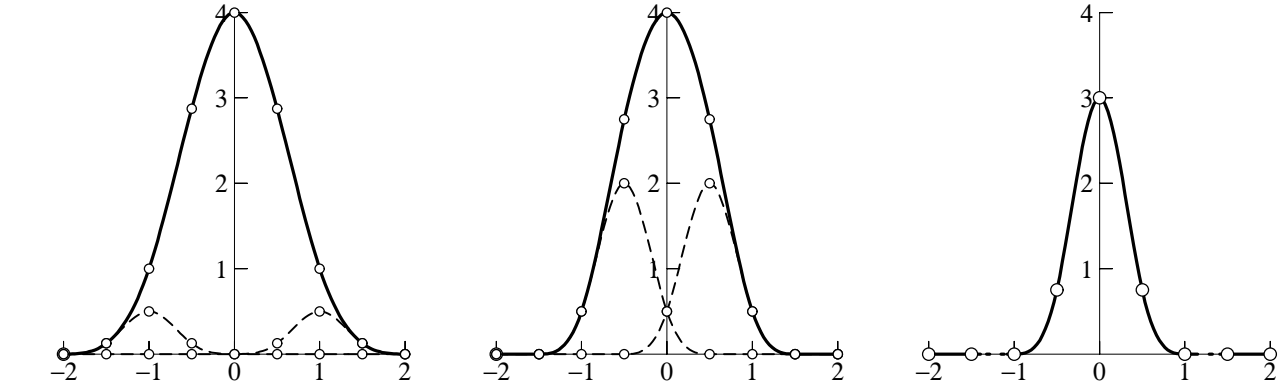


FIG. II.29: Décomposition de $B(x)$ dans la base des $B(2x-n)$.

La *preuve* est indiquée en figure II.29. Vers l'extrémité gauche $B(x) = (x+2)^3$, par contre $B(2x+2) = (2x+2+2)^3 = 8(x+2)^3$. Donc $B(x) - \frac{1}{8}(B(2x+2) + B(2x-2))$ a le support réduit des deux côtés. Puis on réduit une deuxième fois en soustrayant $\frac{1}{2}(B(2x+1) + B(2x-1))$. Il reste un spline en $2x$ dont le support est $[-1,+1]$. Celui doit donc être un multiple de $B(2x)$.

Vers une base "fan" orthogonale de V^1. Les fonctions $\varphi(x-n)$ forment une base orthogonale de V^0. Les fonctions $\varphi(2x-n)$ forment une base orthogonale de V^1, mais elle n'est pas orthogonale à V^0. Essayons de trouver une formule du type

$$\varphi(x) = \sqrt{2} \sum_{n=-\infty}^{\infty} h_n \varphi(2x-n)\ , \tag{9.7}$$

formule qui correspond à (9.6) si on remplace B par φ. Nous développons les deux côtés de cette formule dans la base des $B(2x-m)$:

$$\varphi(x) = \sum_{\ell} \alpha_\ell B(x-\ell) = \sum_{\ell,k} \alpha_\ell c_k B(2x-2\ell-k) = \sum_{m} \Big(\sum_{\ell} \alpha_\ell c_{m-2\ell} \Big) \cdot B(2x-m)$$

pour l'un, et

$$\sqrt{2} \sum_{n} h_n \varphi(2x-n) = \sqrt{2} \sum_{n,\ell} h_n \alpha_\ell B(2x-\ell-n) = \sqrt{2} \sum_{m} \Big(\sum_{\ell} \alpha_\ell h_{m-\ell} \Big) \cdot B(2x-m)$$

pour l'autre. Nous égalisons les deux termes et insérons les c_k de (9.6). Cela nous donne un grand système linéaire pour les h_m que nous écrivons sous la forme

$$\alpha_0 h_m = -\sum_{\ell\neq 0} \alpha_\ell h_{m-\ell} + \frac{1}{\sqrt{2}} \begin{cases} \frac{1}{8}\alpha_{\frac{m}{2}-1} + \frac{3}{4}\alpha_{\frac{m}{2}} + \frac{1}{8}\alpha_{\frac{m}{2}+1} & \text{si } m \text{ est pair} \\ \frac{1}{2}\alpha_{\frac{m-1}{2}} + \frac{1}{2}\alpha_{\frac{m+1}{2}} & \text{si } m \text{ est impair.} \end{cases}$$

Vu que α_0 est le terme dominant, nous résolvons ce système par itérations et obtenons, en une dizaine de cycles, les valeurs

$$\begin{array}{cccc} h_0 = 0.7661300 & h_1 = 0.4339226 & h_2 = -0.0502017 & h_3 = -0.1100370 \\ h_4 = 0.0320809 & h_5 = 0.0420684 & h_6 = -0.0171763 & h_7 = -0.0179823 \\ h_8 = 0.0086853 & h_9 = 0.0082015 & h_{10} = -0.0043538 & h_{11} = -0.0038824 \\ h_{12} = 0.0021867 & h_{13} = 0.0018821 & h_{14} = -0.0011037 & h_{15} = -0.0009272 \\ h_{16} = 0.0005599 & h_{17} = 0.0004621 & h_{18} = -0.0002853 & h_{19} = -0.0002324 \end{array} .$$

Théorème 9.3 *Si l'on pose, avec les valeurs h_n de (9.7),*

$$\psi(x) = \sqrt{2} \sum_{n=-\infty}^{\infty} g_n \varphi(2x-n) \qquad \textit{avec} \qquad g_n = (-1)^n \cdot h_{1-n} \,, \tag{9.8}$$

alors les fonctions

$$\ldots\ \psi(x+3),\ \psi(x+2),\ \psi(x+1),\ \psi(x),\ \psi(x-1),\ \psi(x-2),\ \psi(x-3),\ \ldots$$

sont orthogonales entre elles et orthogonales aux fonctions $\varphi(x-n)$. En d'autres mots, l'espace W^0 engendré pas ces ψ's forme avec V^0 une décomposition orthogonale de V^1 :

$$V^1 = V^0 \oplus W^0 \,.$$

Une illustration est donnée en figure II.30 à gauche : Supposons que seulement h_0 et h_1 soient $\neq 0$. Alors $g_0 = h_1$ et $g_1 = -h_0$ et le vecteur $\psi(x)$ est visiblement orthogonal à $\varphi(x)$.

Le miracle est maintenant qu'on peut continuer en posant à nouveau $x \mapsto 2x$ et on aura successivement des espaces V^m, engendrés par $B(2^m x - n)$, avec

$$V^2 = V^1 \oplus W^1 = V^0 \oplus W^0 \oplus W^1 \,, \qquad V^3 = V^0 \oplus W^0 \oplus W^1 \oplus W^2$$

etcetera. Ainsi *toutes les $\psi(2^m x - n)$ sont orthogonales*. Notre ondelette mère est enfin trouvée et est fièrement présentée dans toute sa beauté en figure II.30 à droite.

Il reste à faire la *preuve* des orthogonalités : on calcule

$$\int_{-\infty}^{\infty} \varphi(x-j)\varphi(x-k)\,dx = 2\sum_{m,n} h_n h_m \int_{-\infty}^{\infty} \varphi(2x-2j-n)\varphi(2x-2k-m)\,dx$$

$$\int_{-\infty}^{\infty} \varphi(x-j)\psi(x-k)\,dx = 2\sum_{m,n} h_n g_m \int_{-\infty}^{\infty} \varphi(2x-2j-n)\varphi(2x-2k-m)\,dx$$

$$\int_{-\infty}^{\infty} \psi(x-j)\psi(x-k)\,dx = 2\sum_{m,n} g_n g_m \int_{-\infty}^{\infty} \varphi(2x-2j-n)\varphi(2x-2k-m)\,dx$$

À cause de l'orthogonalité des $\varphi(2x-n)$, les trois sommes doubles se réduisent à trois sommes simples

$$\sum_n h_n h_{n-2\ell} \qquad \sum_n h_n g_{n-2\ell} \qquad \sum_n g_n g_{n-2\ell} \,.$$

Celle du milieu est zéro par définition des g_n. Car on sait que les $\varphi(x-n)$ sont orthogonaux, la première somme doit être zéro. Donc la troisième somme, qui est la même par définition des g_n, est zéro aussi.

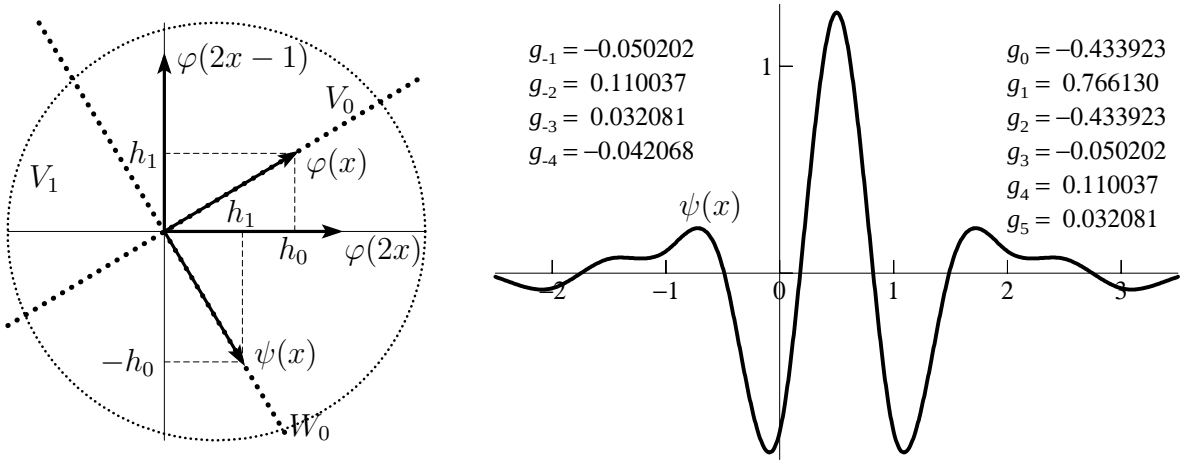


FIG. II.30: Analyse multi-échelles (à gauche) ; Wavelet ‘spline’ de Battle–Lemarié (à droite).