
Cours ENSIMAG 2^e année

— Année 2001/02 —

Modélisation Géométrique

Stefanie Hahmann

Laboratoire LMC-IMAG

e-mail: hahmann@imag.fr

URL <http://www-lmc.imag.fr/lmc-mga/Stefanie.Hahmann>

Contenu

I. Interpolation polynomiale

- 1.1 Problème d'interpolation
- 1.2 Méthode de Lagrange
- 1.3 Interpolation d'Hermite
- 1.4 Méthode de Newton
- 1.5 Méthode des déterminants d'Aitken
- 1.6 Tracé d'une courbe paramétrique polynomiale
- 1.7 Evaluation d'erreur

II. Interpolation Spline

- 2.1 Notations
- 2.2 Base théorique par Π_k^m
- 2.3 Considérons Π_2^3
- 2.4 Représentation des fonctions de S_q
- 2.5 Interpolation
- 2.6 Propriété variationnelle
- 2.7 Algorithme pour splines cubiques naturelles

III. Design polynomial: Courbes de Bézier

- 3.1 Courbe de Bézier
- 3.2 Propriété des polynômes de Bernstein
- 3.3 Algorithme de De Casteljau
- 3.4 Propriétés géométriques des courbes de Bézier
- 3.5 Dérivation d'une courbe de Bézier
- 3.6 Formes polaires

IV. Design Spline: Courbes B-spline

- 4.1 Les fonctions B-spline
- 4.2 Courbes B-spline
- 4.3 Evaluation d'une courbe B-spline: algorithme de De Boor
- 4.4 Propriétés
- 4.5 Insertion d'un nœud

V. Approximation polynomiale

- 5.1 Projection d'un point sur un s.e.v.
- 5.2 Introduction de poids
- 5.3 Introduction de contraintes

VI. Approximation Spline

- 6.1 Approximation et lissage
- 6.2 Algorithmes

Bibliographie

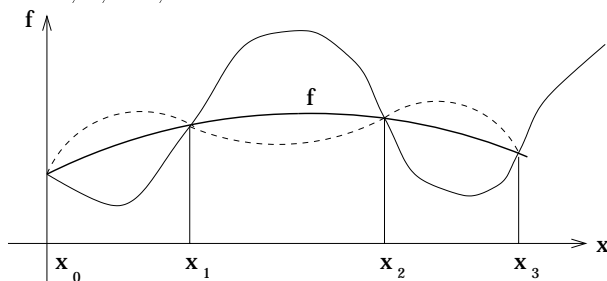
I. Interpolation polynomiale

1.1 PROBLEME D'INTERPOLATION

Soit $f : [a, b] \rightarrow \mathbb{R}$ une fonction définie sur l'intervalle $[a, b]$ à valeurs réelles. Considérons des abscisses distincts x_0, x_1, \dots, x_n de $[a, b]$. Un polynôme P_n de degré $\leq n$ avec des coefficients réels est appelé *polynôme d'interpolation* si

$$P_n(x_i) = f(x_i), \quad i = 0, 1, \dots, n.$$

On posera $f(x_i) = y_i, i = 0, 1, \dots, n$.



Interpolation polynomiale.

Les termes principaux du problème :

- [1.] Existence de solution : Etant donné $n + 1$ valeurs distincts x_i , on cherche un polynôme de degré n admettant ces valeurs,
- [2.] unicité de la solution,
- [3.] méthode de résolution praticable,
- [4.] qualité de la (des) solution(s) : la solution correspond à l'attente de l'utilisateur.

Le polynôme P_n peut s'écrire de la forme suivante :

$$P_n(x) = \sum_{i=0}^n a_i x^i, \quad a_i \in \mathbb{R},$$

où a_0, \dots, a_n sont des nombres réels.

Les coefficients a_0, \dots, a_n vérifient le système de $n + 1$ équations linéaire suivant :

$$\sum_{i=0}^n a_i x_j^i = y_j, \quad 0 \leq j \leq n.$$

Sous forme matricielles, on obtient

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \dots & x_0^n \\ 1 & x_1 & x_1^2 & \dots & x_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^n \end{pmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix}$$

Cette matrice de coefficients est connue comme *matrice de Vandermonde* \mathbf{V} .

$$\det \mathbf{V} = \prod_{i=0}^n \prod_{j=i+1}^n (x_i - x_j).$$

Le déterminant de Vandermonde est non-nul si et seulement si les x_i sont distincts.

Le système d'équations admet une solution unique.

Théorème *Il existe une solution unique pour le problème d'interpolation.*

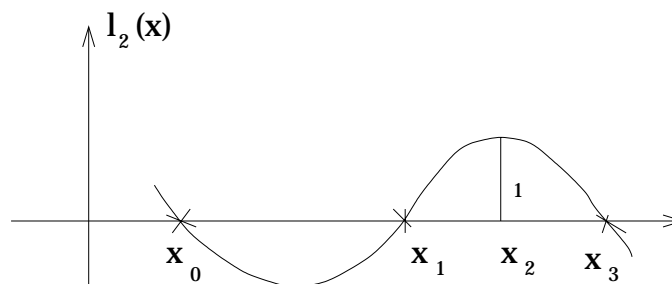
1.2 METHODE DE LAGRANGE (1736–1813)

Lagrange :

Joseph-Louis Comte de Lagrange (1736–1813), mathématicien français, né en Italie. En 1755 : prof. de Géométrie à l'école royale d'Artillerie de Turin, a fondé l'académie des sciences (en Italie), a succédé à Euler comme directeur des mathématiques à l'académie des sciences à Berlin. Revient en France après la mort de Frédéric Le grand, s'est occupé des problèmes iso paramétriques, a fondé le calcul des variations, a contribué au calcul des probabilités, à la théorie des équations et à la théorie des groupes.

La méthode consiste à déterminer les *polynômes de base* $l_j \in P_n$ tels que :

$$l_j(x_i) = \begin{cases} 1 & i = j \\ 0 & \text{sinon} \end{cases}$$



le polynôme l_j , pour $n = 3$ et $j = 2$

Le polynôme d'interpolation s'écrit sous la forme

$$P(x) = \sum_{j=0}^n y_j l_j(x)$$

La valeur de P en x_i est $\sum_{j=0}^n y_j l_j(x_i) = y_i$; $i = 0, \dots, n$. C'est donc le polynôme d'interpolation cherché.

Forme explicite :

$$l_j(x) = \prod_{i=0, i \neq j}^n \frac{x - x_i}{x_j - x_i} = \frac{(x - x_0)(x - x_1) \dots (x - x_{j-1})(x - x_{j+1}) \dots (x - x_n)}{(x_j - x_0)(x_j - x_1) \dots (x_j - x_{j-1})(x_j - x_{j+1}) \dots (x_j - x_n)}$$

Exemple : $n = 2$, $x_0 = 0$, $x_1 = \frac{1}{3}$, $x_2 = 1$;

$$P_n(x) = 3(x - \frac{1}{3})(x - 1)y_0 + (-\frac{9}{2})x(x - 1)y_1 + \frac{3}{2}x(x - \frac{1}{3})y_2.$$

Remarque :

- L'algorithme n'est pas très pratique.
- Il faut complètement recommencer à calculer les $l_j(x)$, si on rajoute des abscisses en plus.

Remarque : base duale

Soit V un espace vectoriel de dimension $n + 1$ et $\phi_0, \phi_1, \dots, \phi_n$ $n + 1$ formes linéaires indépendantes sur V . Alors il existe une unique base $\omega_0, \omega_1, \dots, \omega_n$ de V , dite *base duale* des ϕ_i telle que :

$$\phi_i(\omega_j) = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon.} \end{cases}$$

Avec cette base, on peut explicitement résoudre le problème suivant :

Trouver $v \in V$ tel que $\phi_i(v) = y_i$, $i = 0, \dots, n$. (Les y_i sont des réels donnés).

La solution est

$$v = \sum_{j=0}^n y_j \omega_j.$$

Car $\phi_i(v) = \sum_{j=0}^n y_j \phi_i(\omega_j) = y_i$, $i = 0, \dots, n$.

Dans le cas de l'interpolation polynomiale, on a : $V = \mathcal{P}_n$ et $\phi_i(P) = P(x_i)$, $i = 0, \dots, n$.

Pour $P \in \mathcal{P}_n$. Φ_i sont des formes linéaires sur V , elles sont des applications qui associent à un polynôme P sa valeur en x_i . On trouve alors : $\omega_j = l_j$.

1.3 INTERPOLATION D'HERMITE (1822–1901)

Hermite : Chales Hermite, 1822–1901, mathématicien français, algébriste arithmétique, a développé la théorie des fonctions, notamment les fonctions elliptiques. Avait des difficultés pour passer ces examens (6 ans pour obtenir son premier diplôme)

D'une manière générale, on parle d'interpolation d'Hermite quand l'interpolation ne porte pas seulement sur les valeurs de la fonction mais aussi sur les valeurs des dérivées. On calcule le polynôme $P_3 \in \mathcal{P}_3$ tel que : $P_3(0) = y_0$, $P_3(1) = y_1$, $P_3'(0) = y_0'$, $P_3'(1) = y_1'$. En suivant le principe général de Lagrange (base duale), on a donc les quatre fonctionnelles suivantes, définies sur \mathcal{P}_3 : $\phi_0(P_3) = P_3(0)$, $\phi_1(P_3) = P_3(1)$, $\phi_2(P_3) = P_3'(0)$, $\phi_3(P_3) = P_3'(1)$, et on doit chercher la base de \mathcal{P}_3 duale des ϕ_i .

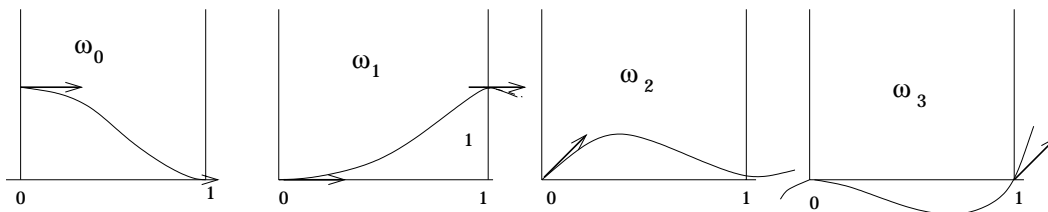
Par exemple $\omega_0 \in \mathcal{P}_3$ doit vérifier :

$$\omega_0(0) = 1, \omega_0(1) = 0, \omega_0'(0) = 0, \omega_0'(1) = 0,$$

alors que $\omega_3 \in \mathcal{P}_3$ vérifiera :

$$\omega_3(0) = 0, \omega_3(1) = 0, \omega_3'(0) = 0, \omega_3'(1) = 1.$$

Graphiquement, les polynômes ω_i ont l'allure suivante



polynômes de base d'Hermite.

Un calcul élémentaire conduit à :

$$\omega_0(x) = (1 - x)^2(1 + 2x)$$

$$\omega_1(x) = x^2(3 - 2x)$$

$$\omega_2(x) = (1 - x)^2x$$

$$\omega_3(x) = x^2(x - 1)$$

La solution du problème est explicitement :

$$P(x) = y_0\omega_0(x) + y_1\omega_1(x) + y_0'\omega_2(x) + y_1'\omega_3(x).$$

Remarque :

Pour tout polynôme P_n de degré n , on a (développement de Taylor) :

$$P_n(x) = P_n(0) + \frac{x}{1!}P_n'(0) + \frac{x^2}{2!}P_n''(0) + \dots + \frac{x^n}{n!}P_n^{(n)}(0).$$

La base canonique peut être considérée comme la base duale de \mathcal{P}_n des fonctionnelles suivantes :

$$\Phi_0(P_n) = P_n(0), \quad \Phi_1(P_n) = \frac{P_n'(0)}{1!}, \quad \Phi_2(P_n) = \frac{P_n''(0)}{2!}, \dots, \quad \Phi_n(P_n) = \frac{P_n^{(n)}(0)}{n!}$$

et ainsi le polynôme $P_n \in \mathcal{P}_n$ vérifiant

$$\Phi_i(P_n) = \frac{P_n^{(i)}(0)}{i!} = a_i, \quad i = 0, \dots, n$$

s'écrit explicitement :

$$P_n(x) = \sum_{i=0}^n a_i x^i.$$

1.4 METHODE DE NEWTON (1643–1727)

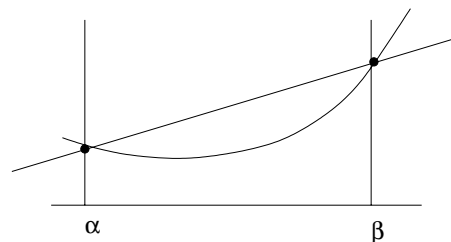
Newton Sir Isaac Newton, 1643–1727, environ un siècle avant Lagrange, physicien, astronome, mathématicien anglais. Est souvent considéré comme le plus influent scientifique de tout le temps ... Calcul différentiel, algèbre, géométrie analytique, loi de la gravité et loi du mouvement des planètes ... Professeur à Combridg, représentant de l'université au parlement et président de la "Royal Society".

La méthode de Newton est basée sur la notion de *différence divisée* d'une variable réelle. On définit la différence divisée d'ordre 1 de f relativement à deux abscisses α et β distincts par :

$$\delta_{\alpha,\beta}^1 f = \frac{f(\beta) - f(\alpha)}{\beta - \alpha}.$$

On voit que $\delta_{\alpha,\beta}^1 f$ représente la pente de la corde correspondant aux deux abscisses α et β . C'est la dérivée première du polynôme d'interpolation de degré 1 de f en α et β .

On voit aussi que si f admet une dérivée première continue, il existe $\xi \in [\alpha, \beta]$ tel que $\delta_{\alpha,\beta}^1 f = f'(\xi)$.



différence divisée d'ordre 1

Si l'on a des abscisses x_i , $i = 0, \dots, n$, toutes distinctes, on pourra donc former les différences divisées d'ordre 1 suivantes :

$$\begin{aligned} \delta_{x_0, x_1}^1 f &= \frac{f(x_1) - f(x_0)}{x_1 - x_0} \\ \delta_{x_{-1}, x_2}^1 f &= \frac{f(x_2) - f(x_1)}{x_2 - x_1} \\ &\vdots \\ \delta_{x_{n-1}, x_n}^1 f &= \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}} \end{aligned}$$

On forme ensuite les différences divisées d'ordre 2 (basées sur trois abscisses) de la façon suivante :

$$\begin{aligned} \delta_{x_0 x_1 x_2}^2 f &= \frac{\delta_{x_1 x_2}^1 f - \delta_{x_0 x_1}^1 f}{x_2 - x_0} \\ \delta_{x_1 x_2 x_3}^2 f &= \frac{\delta_{x_2 x_3}^1 f - \delta_{x_1 x_2}^1 f}{x_3 - x_1} \\ &\vdots \\ \delta_{x_{n-2} x_{n-1} x_n}^2 f &= \frac{\delta_{x_{n-1} x_n}^1 f - \delta_{x_{n-2} x_{n-1}}^1 f}{x_n - x_{n-2}} \end{aligned}$$

Si on remplace les différences divisées d'ordre 1 par leurs expressions, on obtient

$$\begin{aligned} \delta_{x_0 x_1 x_2}^2 f &= \frac{\frac{f(x_2) - f(x_1)}{x_2 - x_1} - \frac{f(x_1) - f(x_0)}{x_1 - x_0}}{x_2 - x_0} \\ &= \frac{f(x_0)}{(x_0 - x_1)(x_0 - x_2)} + \frac{f(x_1)}{(x_1 - x_0)(x_1 - x_2)} + \frac{f(x_2)}{(x_2 - x_0)(x_2 - x_1)}. \end{aligned}$$

Ainsi la différence divisée d'ordre 2 est une combinaison linéaire des valeurs de f aux trois abscisses concernées.

D'une manière générale, on passe des d.d. d'ordre $n - 1$ (basées sur n points) aux d.d. d'ordre n (basées sur $n + 1$ point) par la formule suivante :

Si A représente un ensemble de $n - 1$ abscisses distinctes et si α, β sont deux abscisses distinctes non contenues dans A , alors la d.d. d'ordre n de f basée sur les $n + 1$ abscisses $A \cup \alpha \cup \beta$ s'obtient par la formule :

$$\delta_{A, \alpha, \beta}^n f = \frac{\delta_{A, \alpha}^{n-1} f - \delta_{A, \beta}^{n-1} f}{\alpha - \beta}.$$

Pour calculer les d.d., on adopte la stratégie définie par le tableau triangulaire suivant :

$$\begin{array}{ccccccc}
y_0 = f(x_0) =: \delta_{x_0}^0 f & & & & & & \\
& \searrow & & & & & \\
& & \delta_{x_0 x_1}^1 f & & & & \\
& \nearrow & \searrow & & & & \\
y_1 = f(x_1) =: \delta_{x_1}^0 f & & & \delta_{x_0 x_1 x_2}^2 f & & & \\
& \searrow & \nearrow & \searrow & & & \\
& & \delta_{x_1 x_2}^1 f & & & & \\
& \nearrow & \searrow & \nearrow & & & \\
y_2 = f(x_2) =: \delta_{x_2}^0 f & & & \delta_{x_1 x_2 x_3}^2 f & & & \\
& \searrow & \nearrow & \searrow & \nearrow & & \\
& & \delta_{x_2 x_3}^1 f & & & & \\
& \nearrow & & & & & \\
y_3 = f(x_3) =: \delta_{x_3}^0 f & & & & & & \delta_{x_0 x_1 x_2 x_3}^3 f
\end{array}$$

On montre que la d.d. d'ordre n de f basée sur x_0, \dots, x_n vérifie

$$\begin{aligned}
\delta_{x_0 \dots x_n}^3 f &= \frac{f(x_0)}{(x_0 - x_1)(x_0 - x_2) \dots (x_0 - x_n)} + \frac{f(x_1)}{(x_1 - x_0)(x_1 - x_2) \dots (x_1 - x_n)} + \dots + \\
&\quad \frac{f(x_n)}{(x_n - x_0)(x_n - x_1) \dots (x_n - x_{n-1})} \\
&= \sum_{j=0}^n \frac{f(x_j)}{\prod_{i=0, i \neq j}^n (x_j - x_i)}.
\end{aligned}$$

Cette formule s'appelle "*identité de Newton*". On voit clairement dans cette formule que la d.d. ne dépend pas de l'ordre des abscisses mais seulement de l'ensemble $\{x_0, \dots, x_n\}$. Si f admet une dérivée $n^{\text{ème}}$ continue, alors il existe $\xi \in [a, b]$ (le plus petit intervalle contenant les x_i) tel que

$$\delta_{x_0, \dots, x_n}^n f = \frac{f^n(\xi)}{n!}, \quad \text{si } f \in C^n. \quad (*)$$

Il en résulte de (*) que

$$P \in \mathcal{P}_{n-1} \longrightarrow \delta_{x_0, \dots, x_n}^n P = 0.$$

Exemple : $P(x) = x$

$$\begin{aligned}
\delta_{x_0, x_1, x_2}^2 &= \frac{x_0}{(x_0 - x_1)(x_0 - x_2)} + \frac{x_1}{(x_1 - x_0)(x_1 - x_2)} + \frac{x_2}{(x_2 - x_0)(x_2 - x_1)} \\
&= \frac{x_0(x_1 - x_2) + x_1(x_2 - x_0) + x_2(x_0 - x_1)}{(x_0 - x_1)(x_0 - x_2)(x_1 - x_2)} = 0
\end{aligned}$$

Formule de Newton

(a) à l'ordre 0

Considérons la d.d. d'ordre 1 basée sur x_0 et x :

$$\delta_{x_0 x}^1 f = \frac{f(x) - f(x_0)}{x - x_0}$$

on peut écrire

$$f(x) = f(x_0) + (x - x_0)\delta_{x_0 x}^1 f,$$

dont le premier terme du membre droit est un polynôme de degré 0 et le deuxième terme est le reste. C'est la formule de Newton d'ordre 0.

(b) à l'ordre 1 Considérons la d.d. d'ordre 2 basée sur x_0 , x_1 et x :

$$\delta_{x_0 x_1 x}^2 f = \frac{\delta_{x_0 x}^1 f - \delta_{x_0 x_1}^1 f}{x - x_1}$$

(dans cette formule on a $A = \{x_0\}$).

On a donc :

$$\delta_{x_0 x}^1 f = \delta_{x_0 x_1}^1 f + (x - x_1)\delta_{x_0 x_1 x}^2 f.$$

En remplaçant $\delta_{x_0 x}^1 f$ par son expression ci-dessus dans la formule d'ordre 0, on obtient :

$$f(x) = f(x_0) + (x - x_0)\delta_{x_0 x_1}^1 f + (x - x_0)(x - x_1)\delta_{x_0 x_1 x}^2 f \quad (1)$$

c'est la formule de Newton à l'ordre 1.

Remarques :

- On voit que l'on a exprimé f sous la forme d'une somme d'un polynôme de degré 1 et d'un reste. Dans le cas où f est un polynôme de degré 1, ce reste disparaît : $\delta_{x_0 x_1 x}^2 P_1 = 0$ et l'on obtient :

$$P_1(x) = P_1(x_0) + (x - x_0)\delta_{x_0 x_1}^1 P_1 \quad (2)$$

- Supposons que l'on cherche le polynôme de degré 1 qui interpole f en x_0 et x_1 : $P_1(x_0) = f(x_0)$ et $P_1(x_1) = f(x_1)$.
On aura donc aussi $\delta_{x_0 x_1}^1 P_1 = \delta_{x_0 x_1}^1 f$.
Par conséquent :

$$P_1(x) = f(x_0) + (x - x_0)\delta_{x_0 x_1}^1 f. \quad (3)$$

C'est le polynôme d'interpolation de f d'ordre 1.

- On fera bien la différence entre les formules :

- (1) qui est une identité algébrique pour toute fonction f ,
- (2) qui est une identité pour tout polynôme P_1 de degré 1,
- (3) qui exprime le polynôme d'interpolation de degré 1 de f en x_0 .

- Comparaison avec Lagrange :

Le polynôme P_1 sous forme de Lagrange s'écrit :

$$P_1(x) = \frac{x - x_1}{x_0 - x_1} f(x_0) + \frac{x - x_0}{x_1 - x_0} f(x_1),$$

et sous forme de Newton s'écrit :

$$P_1(x) = f(x_0) + (x - x_0) \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

(c) à l'ordre n

En continuant de poursuivre la construction précédente, on obtient la forme de Newton à l'ordre n :

$$P_n(x) = f(x_0) + (x - x_0) \delta_{x_0 x_1}^1 f + (x - x_0)(x - x_1) \delta_{x_0 x_1 x_2}^2 f + \dots \\ + (x - x_0) \dots (x - x_{n-1}) \delta_{x_0 \dots x_n}^n f + (x - x_0) \dots (x - x_n) \delta_{x_0 \dots x_n x}^{n+1} f.$$

Donc, si f est un polynôme de degré n , on a l'identité :

$$P_n(x) = P_n(x_0) + (x - x_0) \delta_{x_0 x_1}^1 P_n + (x - x_0)(x - x_1) \delta_{x_0 x_1 x_2}^2 P_n + \dots \\ + (x - x_0) \dots (x - x_{n-1}) \delta_{x_0 \dots x_n}^n P_n$$

et ainsi, le polynôme P_n qui interpole f en x_0, x_1, \dots, x_n est donné par :

$$P_n(x) = f(x_0) + (x - x_0) \delta_{x_0 x_1}^1 f + (x - x_0)(x - x_1) \delta_{x_0 x_1 x_2}^2 f + \dots \\ + (x - x_0) \dots (x - x_{n-1}) \delta_{x_0 \dots x_n}^n f$$

Progression des calculs :

- a. **forme permettant la progression du degré** : Il arrive, pour certaines applications, que l'on ne connaisse pas à priori le degré du polynôme. On utilise alors la forme suivante :

$P_0(x) = f(x_0)$	Polynôme de degré 0 interpolant f en x_0
-------------------	--

$P_1(x) = f(x_1) + (x - x_0) \delta_{x_0 x_1}^1 f$	Polynôme de degré 1 interpolant f en x_0, x_1
--	---

⋮

$P_n(x) = P_{n-1}(x) + (x - x_0) \dots (x - x_{n-1}) \delta_{x_0 \dots x_n}^n f$	polynôme de degré n interpolant f en x_0, \dots, x_n
--	--

C'est l'une des avantages de la méthode de Newton de permettre une progression commode du degré. Pour passer du degré $n - 1$ au degré n , il suffit d'ajouter un terme alors que pour la méthode de Lagrange, il faut pratiquement refaire tous les calculs.

b. forme lorsque le degré est fixé à priori :

Lorsque le degré du polynôme est fixé, on peut employer une méthode plus économique pour le calcul du polynôme. Cette méthode est similaire à celle que l'on peut employer pour le polynôme exprimé sous la *forme canonique* (schéma de Horner). Montrons-le pour un polynôme de degré 3 d'abord :

$$\begin{aligned}
 P_3(x) &= a_0 + a_1x + a_2x^2 + a_3x^3 \\
 \text{Horner} \quad &= a_0 + x \left[a_1 + x \left\{ a_2 + x \underbrace{(a_3)}_{\gamma_0} \right\} \right] \\
 &\quad \underbrace{\hspace{10em}}_{\gamma_1} \\
 &\quad \underbrace{\hspace{15em}}_{\gamma_2} \\
 &\quad \underbrace{\hspace{20em}}_{\gamma_3 = P_3(x)}
 \end{aligned}$$

| On pose $\gamma_0 := a_3$, puis :

| $\gamma_1 = x \cdot \gamma_0 + a_2$

| $\gamma_2 = x \cdot \gamma_1 + a_1$

| $\gamma_3 = x \cdot \gamma_2 + a_0$.

| Plus généralement le polynôme $\sum_{i=0}^n a_i x^i$ peut être calculé par

```

|  $\gamma_0$  :=  $a_n$ 
| for  $i := 1$  to  $n$  do |
|  $\gamma_i := x \cdot \gamma_{i-1} + a_{n-i}$ . |

```

Coût : n multiplications + n additions.

On procède de la même façon pour le polynôme exprimé sous la *forme de Newton* :

On posera : $\delta_i := \delta_{x_0, \dots, x_i}^i f$, $i = 0, \dots, n$. Pour $n = 3$, le polynôme P_3 s'écrit :

$$\begin{aligned}
 P_3(x) &= \delta_0 + \delta_1(x - x_0) + \delta_2(x - x_0)(x - x_1) + \delta_3(x - x_0)(x - x_1)(x - x_2) \\
 \text{Horner} \quad &= \delta_0 + (x - x_0) \left[\delta_1 + (x - x_1) \left\{ \delta_2 + (x - x_2) \underbrace{(\delta_3)}_{\gamma_0} \right\} \right] \\
 &\quad \underbrace{\hspace{10em}}_{\gamma_1} \\
 &\quad \underbrace{\hspace{15em}}_{\gamma_2} \\
 &\quad \underbrace{\hspace{20em}}_{\gamma_3 = P_3(x)}
 \end{aligned}$$

On pose $\gamma_3 := \delta_3$, puis $\gamma_1 := (x - x_2) \cdot \gamma_0 + \delta_2$
 $\gamma_2 := (x - x_1) \cdot \gamma_1 + \delta_1$
 $\gamma_1 := (x - x_0) \cdot \gamma_2 + \delta_0$.

Plus généralement, le polynôme $\sum_{i=0}^n \delta_i(x - x_0) \dots (x - x_{i-1})$ s'obtient par

```
|   $\gamma_0$       :=       $\delta_n$       |
| for   $i$     :=  1 to   $n$  do  |
|   $\gamma_i$  :=  $(x - x_{n-i}) \cdot \gamma_{i-1} + \delta_{n-i}$ . |
```

Coût : n multiplications + $2n$ additions.

Remarque :

Considérons les formes linéaires ϕ_i suivantes, définies sur \mathcal{P}_n : $\phi_i(P) = \delta_{x_0 \dots x_i}^i P$, $i = 0, \dots, n$. En appliquant l'identité de Newton, on trouve facilement le polynôme $\omega_j \in \mathcal{P}_n$ tel que :

$$\phi_i(\omega_j) = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon} \end{cases}$$

Il est égal à :

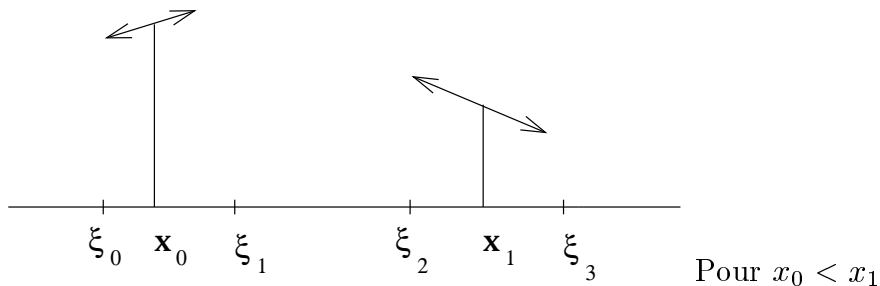
$$\begin{cases} \omega_0(x) = 1 & \text{si } j = 0 \\ \omega_j(x) = (x - x_0) \dots (x - x_{j-1}) & \text{si } j = 1, \dots, n. \end{cases}$$

Les ω_j , $j = 0, \dots, n$ forment une base de \mathcal{P}_n duale des ϕ_i .

Forme de Newton pour l'interpolation d'Hermite :

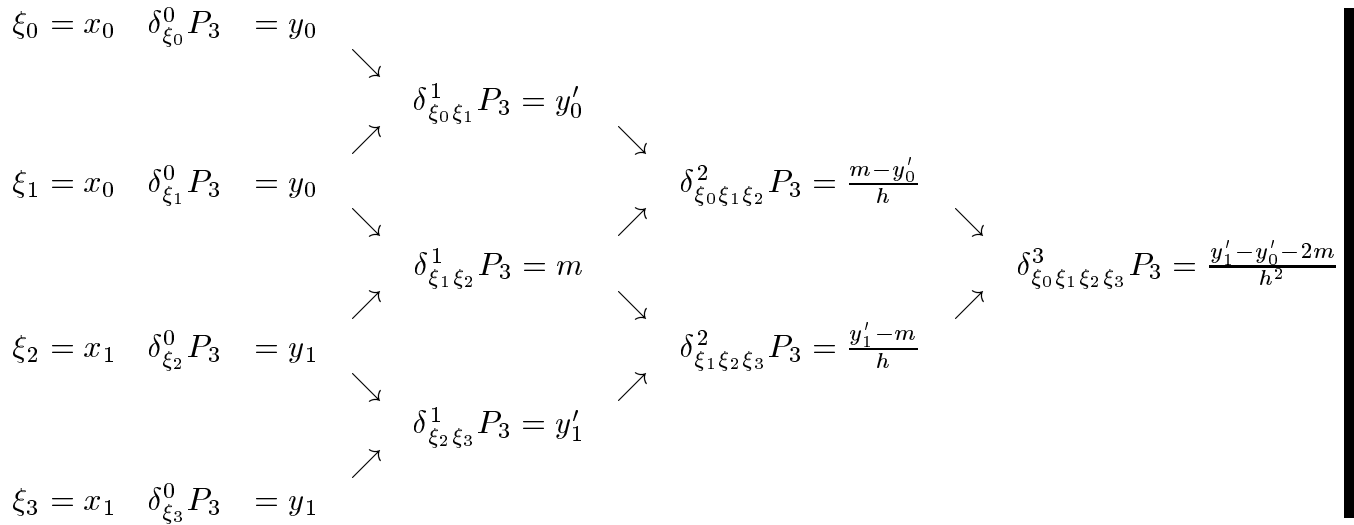
Reprenons le problème de la détermination du polynôme $P_3 \in \mathcal{P}_3$ tel que

$$P_3(x_0) = y_0, P_3(x_1) = y_1 \\ P_3'(x_0) = y_0', P_3'(x_1) = y_1'.$$



Le problème peut être considéré comme le cas limite du problème d'interpolation à 4 points $\xi_0, \xi_1, \xi_2, \xi_3$ dans lequel ξ_0 et ξ_1 tendent vers x_0 et ξ_2, ξ_3 tendent vers x_1 . On sait alors que

$\delta_{\xi_0 \xi_1}^1 f$ tend vers $f'(x_0)$ et $\delta_{\xi_2 \xi_3}^1 f$ tend vers $f'(x_1)$. On considère donc le tableau des d.d. dans lequel on effectue ce passage à la limite (on pose $m = \frac{y_1 - y_0}{x_1 - x_0}$; $h = x_1 - x_0$)



Ainsi, on a la formule de Newton suivante :

$$P_3(x) = \delta_{\xi_0}^0 P_3 + (x - \xi_0) \delta_{\xi_0 \xi_1}^1 P_3 + (x - \xi_0)(x - \xi_1) \delta_{\xi_0 \xi_1 \xi_2}^2 P_3 + (x - \xi_0)(x - \xi_1)(x - \xi_2) \delta_{\xi_0 \xi_1 \xi_2 \xi_3}^3 P_3,$$

qui devient dans le cas présent :

$$P_3(x) = y_0 + (x - x_0)y'_0 + (x - x_0)^2 \frac{m - y'_0}{h} + (x - x_0)^2 (x - x_1) \frac{y'_1 - y'_0 - 2m}{h^2}.$$

Exercice : Déterminer le polynôme de degré 3 qui vérifie ($x_0 < x_1$)

$$P_3(x_0) = y_0 \quad P'_3(x_0) = y'_0 \quad P''_3(x_0) = y''_0 \quad P_3(x_1) = y_1$$

Exercice : Déterminer le polynôme de degré 5 qui vérifie ($x_0 < x_1$)

$$\begin{aligned} P_5(x_0) = y_0 & \quad P'_5(x_0) = y'_0 & \quad P''_5(x_0) = y''_0 \\ P_5(x_1) = y_1 & \quad P'_5(x_1) = y'_1 & \quad P''_5(x_1) = y''_1. \end{aligned}$$

En conclusion, la méthode de Newton est plus économique que la méthode de Lagrange. Elle permet une progression commode en fonction du degré.

On notera cependant qu'elle est particulière aux polynômes alors que la méthode de Lagrange résulte d'un principe plus général (bases duales) qui s'applique à tout sous-espace vectoriel, pourvu que l'on sache expliciter la base duale.

1.5 METHODE DES DETERMINANTS DE AITKEN

Notons $P_{x_0x_1}$ le polynôme de degré 1 tel que :

$$P_{x_0x_1}(x_i) = y_i, \quad i = 0, 1.$$

La formule de Lagrange permet de l'écrire :

$$\begin{aligned} P_{x_0x_1}(x) &= \frac{x - x_0}{x_0 - x_1}y_0 + \frac{x - x_1}{x_1 - x_0}y_1 \\ &= \frac{1}{x_0 - x_1} \det \begin{bmatrix} y_0x - x_0 \\ y_1x - x_1 \end{bmatrix} \end{aligned}$$

ou encore, en posant $P_{x_0}(x) \equiv y_0$, $P_{x_1}(x) \equiv y_1$ (polynômes de degré 0 qui interpolent en x_0 et x_1 respectivement) :

$$P_{x_0x_1}(x) = \frac{1}{x_0 - x_1} \det \begin{bmatrix} P_{x_0}(x)x - x_0 \\ P_{x_1}(x)x - x_1 \end{bmatrix}.$$

On peut, de même, obtenir le polynôme $P_{x_0x_1x_2}$ de degré 2 qui interpole en x_0 , x_1 et x_2 par la formule :

$$P_{x_0x_1x_2}(x) = \frac{x - x_2}{x_1 - x_2}P_{x_0x_1}(x) + \frac{x - x_1}{x_2 - x_1}P_{x_0x_2}(x).$$

En effet, pour $x = x_1$, le deuxième terme disparaît et il reste :

$$P_{x_0x_1x_2}(x_1) = \frac{x_1 - x_2}{x_1 - x_2}P_{x_0x_1}(x_1) = y_1.$$

De même, pour $x = x_2$, le premier terme disparaît et on obtient

$$P_{x_0x_1x_2}(x_2) = \frac{x_2 - x_1}{x_2 - x_1}P_{x_0x_2}(x_2) = y_2.$$

Enfin, pour x_0 , les deux polynômes de degré 1 qui composent la somme sont tous les deux égaux à y_0 et on a

$$P_{x_0x_1x_2}(x_0) = \frac{x_0 - x_2}{x_1 - x_2}y_0 + \frac{x_0 - x_1}{x_2 - x_1}y_0 = y_0.$$

On peut encore écrire $P_{x_0x_1x_2}(x)$ sous la forme

$$P_{x_0x_1x_2}(x) = \frac{1}{x_1 - x_2} \det \begin{bmatrix} P_{x_0x_1}(x)x - x_1 \\ P_{x_0x_2}(x)x - x_2 \end{bmatrix}.$$

Plus généralement, supposons que l'on connaisse $P_{X,a}(x)$, polynôme de degré $n - 1$ qui interpole une fonction f sur $X \cup \{a\}$, où X est un ensemble de $n - 1$ points distincts ($a \notin X$) et que l'on connaisse aussi $P_{X,b}$ (où $b \neq a, b \notin X$), alors on forme le polynôme de degré n qui interpole f sur $X \cup \{a\} \cup \{b\}$ par :

$$P_{X,a,b}(x) = \frac{x-b}{a-b}P_{X,a}(x) + \frac{x-a}{b-a}P_{X,b}(x),$$

que l'on peut encore écrire :

$$P_{X,a,b}(x) = \frac{1}{a-b} \det \begin{bmatrix} P_{X,a}(x)(x-a) \\ P_{X,b}(x)(x-b) \end{bmatrix}.$$

En effet, pour $x = a$, on a $P_{X,a,b}(a) = \frac{a-b}{a-b}P_{X,a}(a) = f(a)$, et de même on trouve $P_{X,a,b}(b) = f(b)$. Par ailleurs, on observe que

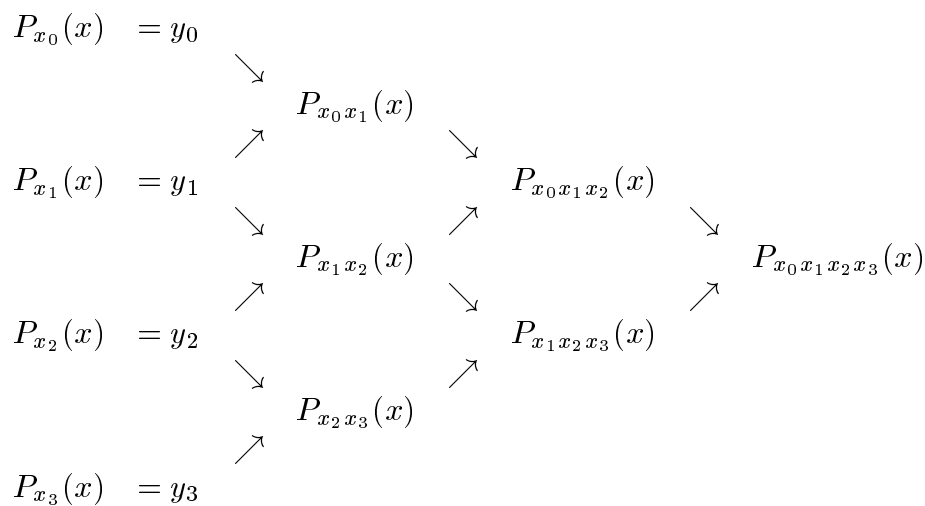
$$\frac{x-b}{a-b} + \frac{x-a}{b-a} \equiv 1$$

($\frac{x-b}{a-b}$ et $\frac{x-a}{b-a}$ sont les coordonnées barycentriques de x par rapport à $[a, b]$).

Donc pour tout $c \in X$, on aura :

$$P_{X,a,b}(c) = \frac{c-b}{a-b}f(c) + \frac{c-a}{b-a}f(c) = f(c).$$

La formule de récurrence précédente permet plusieurs stratégies différentes pour la progression des calculs. Par exemple, on pourra progresser suivant le tableau triangulaire suivant :



1.6 TRACE D'UNE COURBE PARAMETRIQUE POLYNOMIALE

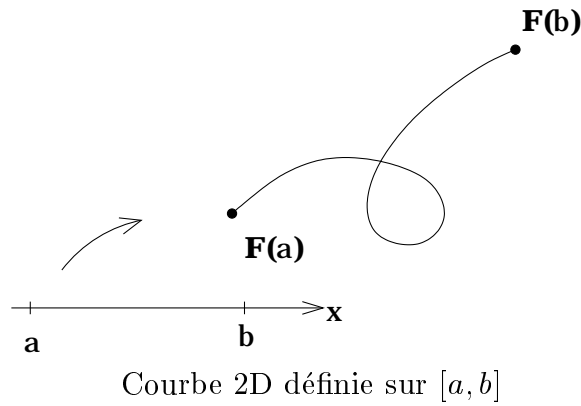
Considérons une courbe polynomiale de \mathbb{R}^2 définie par

$$\mathbf{F}(x) = \begin{pmatrix} p(x) \\ q(x) \end{pmatrix} \text{ pour } x \in [a, b],$$

P, Q sont deux polynômes sur $[a, b]$, tels que

$$\mathbf{F}(x_i) = \mathbf{p}_i, \quad i = 0, \dots, 3.$$

On notera $\mathbf{p}_i^0 = \mathbf{p}_i$.



Pour trouver $\mathbf{F}(x)$, on pourra, en suivant l'algorithme précédent, procéder en 3 étapes.

(1). On forme :

$$P_0^1 = \mathbf{F}_{x_0x_1}(x) = \frac{x - x_1}{x_0 - x_1} \mathbf{p}_0^0 + \frac{x - x_0}{x_1 - x_0} \mathbf{p}_1^0.$$

Comme la somme des coefficients est égale à 1, P_0^1 se trouvera sur la droite engendrée par P_0^0 et P_1^0 et sera à la même position relative que x par rapport à $[x_0, x_1]$.

On forme de même

$$P_1^1 = \mathbf{F}_{x_1x_2}(x) = \frac{x - x_2}{x_1 - x_2} \mathbf{p}_1^0 + \frac{x - x_1}{x_2 - x_1} \mathbf{p}_2^0,$$

$$P_2^1 = \mathbf{F}_{x_2x_3}(x) = \frac{x - x_3}{x_2 - x_1} \mathbf{p}_2^0 + \frac{x - x_2}{x_3 - x_2} \mathbf{p}_3^0.$$

(2). On forme ensuite :

$$\begin{aligned} P_0^2 = \mathbf{F}_{x_0x_1x_2}(x) &= \frac{x - x_2}{x_0 - x_2} \mathbf{F}_{x_0x_1}(x) + \frac{x - x_0}{x_2 - x_0} \mathbf{F}_{x_1x_2}(x) \\ &= \frac{x - x_2}{x_0 - x_2} \mathbf{p}_0^1 + \frac{x - x_0}{x_2 - x_0} \mathbf{p}_1^1, \end{aligned}$$

$$P_1^2 = \mathbf{F}_{x_0x_1x_2}(x) = \frac{x - x_3}{x_1 - x_3} \mathbf{p}_1^1 + \frac{x - x_1}{x_3 - x_1} \mathbf{p}_2^1.$$

Ainsi \mathbf{p}_0^2 se trouve sur la droite engendrée par P_0^1 et P_1^1 et partage le segment $P_0^1P_1^1$ comme x partage $[x_0, x_2]$.

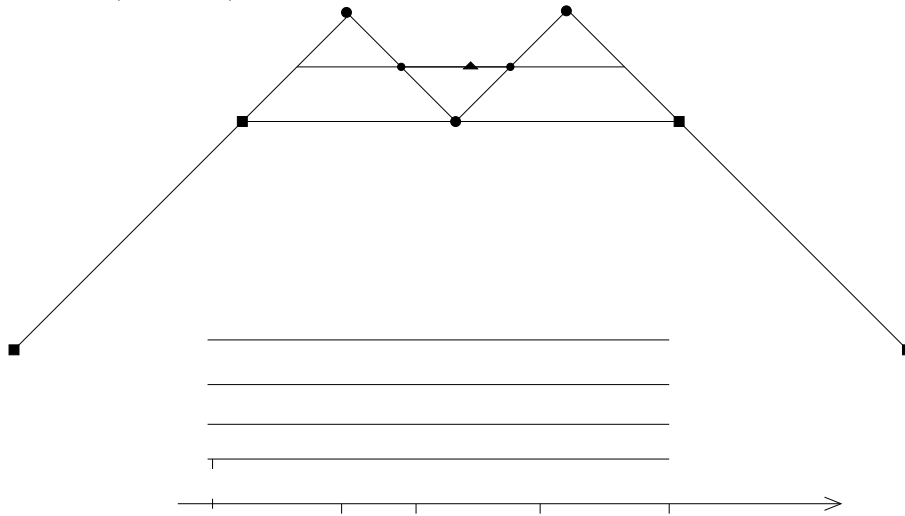
(3). Finalement, on forme

$$\begin{aligned} P_0^3 = \mathbf{F}_{x_0x_1x_2x_3}(x) &= \frac{x - x_3}{x_0 - x_3} \mathbf{F}_{x_0x_1x_2}(x) + \frac{x - x_0}{x_3 - x_0} \mathbf{F}_{x_1x_2x_3}(x) \\ &= \frac{x - x_3}{x_0 - x_3} \mathbf{p}_0^2 + \frac{x - x_0}{x_3 - x_0} \mathbf{p}_1^2. \end{aligned}$$

Ainsi P_0^3 se trouve sur la droite engendrée par \mathbf{p}_0^2 et \mathbf{p}_1^2 et partage $\mathbf{p}_0^2\mathbf{p}_1^2$ comme x partage $[x_0, x_3]$.

Exemple :

$x_0 = 0, x_1 = 1, x_2 = 2, x_3 = 3$ et $x = 1.5$.



\mathbf{p}_1^0 partage $\mathbf{p}_0^0\mathbf{p}_1^0$ comme x partage $[x_0, x_1]$

\mathbf{p}_0^1 partage $\mathbf{p}_0^0\mathbf{p}_1^0$ comme x partage $[x_0, x_1]$.

\mathbf{p}_1^1 partage $\mathbf{p}_1^0\mathbf{p}_2^0$ comme x partage $[x_1, x_2]$.

\mathbf{p}_2^1 partage $\mathbf{p}_2^0\mathbf{p}_3^0$ comme x partage $[x_2, x_3]$.

\mathbf{p}_0^2 partage $\mathbf{p}_0^1\mathbf{p}_1^1$ comme x partage $[x_0, x_2]$.

\mathbf{p}_1^2 partage $\mathbf{p}_1^1\mathbf{p}_2^1$ comme x partage $[x_1, x_3]$.

\mathbf{p}_0^3 partage $\mathbf{p}_0^2\mathbf{p}_1^2$ comme x partage $[x_0, x_3]$.

Représentation paramétrique d'une courbe plane

Les coordonnées cartésiennes x, y d'un point $\mathbf{P}(x, y)$ sont données par des fonctions à valeurs réelles d'un paramètre réel t :

$$x = x(t)$$

$$y = y(t).$$

Ces fonctions sont $C^0[a, b]$ sur l'intervalle du paramètre et décrivent une courbe continue k .

En géométrie différentielle, on exige d'une telle courbe d'être *lisse*, i.e. tangente continue en chaque point de la courbe ($x(t), y(t)$ admettent une dérivée première continue sur $[a, b]$ que ne s'annule jamais simultanément) :

$$\begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \end{bmatrix} \neq \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \forall t \in [a, b]$$

Exercice : $x_0 = 0, x_1 = 1, x_2 = 2, x_3 = 3, x = 1.5$

$\mathbf{F} : \mathbb{R} \Rightarrow \mathbb{R}^2$ à composantes polynomiales de degré 3. On donne quatre points $\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$ de \mathbb{R}^2 .

Si \mathbf{F} vérifie $\mathbf{F}(x_i) = \mathbf{P}_i, i = 0, \dots, 3$.

Montrer que le point $\mathbf{P} = \mathbf{F}(x)$ peut être obtenu simplement par :

$$\mathbf{P} = \mathbf{P}_1 + \mathbf{P}_2 2 + \frac{1}{8} \left[\frac{\mathbf{P}_1 + \mathbf{P}_2}{2} - \frac{\mathbf{P}_0 + \mathbf{P}_3}{2} \right],$$

$$\mathbf{P} = \mathbf{M} + \frac{1}{8}(\mathbf{M} - \mathbf{N})$$

Remarque :

Une méthode classique pour trouver une courbe passant par des points est la suivante :

On a des points $\mathbf{P}_i \in \mathbb{R}^2$. Entre chaque couple de points successifs $\mathbf{P}_i \mathbf{P}_{i+1}$, on place un point \mathbf{Q}_i par la formule

$$\mathbf{Q}_i = \frac{\mathbf{P}_i + \mathbf{P}_{i+1}}{2} + \omega \left[\frac{\mathbf{P}_i + \mathbf{P}_{i+1}}{2} - \frac{\mathbf{P}_{i-1} + \mathbf{P}_{i+2}}{2} \right], \quad i = 1, \dots, n - 2$$

(pour $\omega = \frac{1}{8}$, on retrouve la méthode précédente).

On recommence la construction avec l'ensemble des points $\mathbf{P}_i, \mathbf{Q}_i$ et ainsi de suite jusqu'à ce que les points soient assez rapprochés pour que la courbe polynomiale soit assez lisse.

Le paramètre $\omega > 0$ agit comme un paramètre de tension (courbe plus ou moins proche de la corde).

Pour ω assez grand, on obtient une courbe très cassée, d'allure fractale...

1.7 EVALUATION D'ERREUR

Taylor

Brook Taylor (1685–1731) est 42 ans plus jeune que Newton, mais décède 4 ans plus tard à 46 ans. Analyste, géomètre, peintre, philosophe est néanmoins anglais... Travaux sur le calcul infinitesimal et la perspective. Certains de ces résultats (non publiés) ont été attribués à Johann Bernailli et ont été 60 ans plus tard reconnus à Taylor; par exemple l'importance du développement de Taylor. B. Taylor a fait partie du comité qui a délibéré ce que l'on appelle *formule de Taylor* à l'ordre $k - 1$ avec reste sous forme intégrale.

Une première méthode pour évaluer l'erreur d'interpolation est basée sur la formule de Taylor :

Si f a une dérivée première continue, on a de façon évidente

$$f(x) = f(x_0) + \int_{x_0}^x f'(t) dt.$$

Si f a une dérivée seconde continue, en intégrant par parties, on a

$$\begin{aligned} f(x) &= f(x_0) + [(t-x)f'(t)]_{t=x_0}^{t=x} - \int_{x_0}^x (t-x)f''(t) dt \\ &= f(x_0) + (x-x_0)f'(x_0) + \int_{x_0}^x (x-t)f''(t) dt. \end{aligned}$$

Si f a une dérivée 3^{ème} continue, on peut encore intégrer par parties et on obtient :

$$f(x) = f(x_0) + (x-x_0)f'(x_0) + \frac{(x-x_0)^2}{2!} f''(x_0) + \int_{x_0}^x \frac{(x-t)^2}{2} f'''(t) dt.$$

Plus généralement, si f admet une dérivées k^{e} continue, on obtient la formule de Taylor d'ordre $k - 1$ avec reste sous forme d'intégrale :

$$f(x) = f(x_0) + (x-x_0)f'(x_0) + \dots + \frac{(x-x_0)^{k-1}}{(k-1)!} f^{(k-1)}(x_0) + \int_{x_0}^x \frac{(x-t)^{k-1}}{(k-1)!} f^{(k)}(t) dt.$$

Supposons que l'on soit intéressé aux valeurs de x situées dans un intervalle $[x_0, x_1]$. Si l'on utilise la notion

$$(g)_+ = \begin{cases} g & \text{si } g \geq 0 \\ 0 & \text{sinon} \end{cases} = \max(g, 0)$$

alors pour tout $x \in [x_0, x_1]$, on a :

$$\left| \begin{array}{l} f(x) \\ \hline \end{array} \right| = f(x_0) + \frac{(x-x_0)}{1!} f'(x_0) + \dots + \frac{(x-x_0)^{k-1}}{(k-1)!} f^{(k-1)}(x_0) + \int_{x_0}^{x_1} \frac{(x-t)_+^{k-1}}{(k-1)!} f^{(k)}(t) dt$$

Calcul d'erreur

Prenons l'exemple simple où on remplace la valeur de $f(x)$ sur l'intervalle $[x_0, x_1]$ par la valeur du polyôme $P_1(x)$ de degré 1 qui concide avec f en x_0 et x_1 .

D'après la formule de Lagrange, on a :

$$P_1(x) = \frac{x-x_1}{x_0-x_1} f(x_0) + \frac{x-x_0}{x_1-x_0} f(x_1).$$

Posons :

$$\begin{aligned} e(x) &= P_1(x) - f(x) \\ &= \frac{x-x_1}{x_0-x_1} f(x_0) + \frac{x-x_0}{x_1-x_0} f(x_1) - f(x). \end{aligned}$$

Remplaant $f(x) = \underbrace{P(x)}_{\text{polynome}} + \underbrace{r(x)}_{\text{reste}}$, où, d'après la formule de Taylor :

$$\begin{aligned} P(x) &= f(x_0) + (x-x_0)f'(x_0) \\ r(x) &= \int_{x_0}^{x_1} (x-t)_+ f''(t) dt \end{aligned}$$

On remarque que l'on a : (formule de Lagrange)

$$\frac{x-x_1}{x_0-x_1} P(x_0) + \frac{x-x_0}{x_1-x_0} P(x_1) - P(x), \text{ pour tout } x \text{ et pour } P \text{ avec } f(x_0) = P(x_0), f(x_1) = P(x_1).$$

(C'est en effet l'erreur que l'on commet lorsqu'on interpole P , de degré 1 par un polynôme de degré 1).

On a donc : $(e(x) = \frac{x-x_1}{x_0-x_1} f(x_0) + \frac{x-x_0}{x_1-x_0} f(x_1) - f(x), f = p(x) + r(x))$

$$e(x) = \frac{x-x_1}{x_0-x_1} r(x_0) + \frac{x-x_0}{x_1-x_0} r(x_1) - r(x)$$

$$\text{car } \frac{x-x_1}{x_0-x_1} P(x_0) + \frac{x-x_0}{x_1-x_0} P(x_1) - P(x) = 0.$$

C'est à dire, en développant les calculs (pour $r(x), t = x_0 : (x_0-x_0)_+ = 0 \implies r(x_0) = 0$) :

$$\begin{aligned} e(x) &= 0 + \frac{x-x_0}{x_1-x_0} \int_{x_0}^{x_1} (x_1-t)_+ f''(t) dt - \int_{x_0}^{x_1} (x-t)_+ f''(t) dt \\ &= \int_{x_0}^{x_1} \left[\frac{(x-x_0)(x_1-t)_+}{x_1-x_0} - (x-t)_+ \right] f''(t) dt. \end{aligned}$$

$$\text{Posons } \phi(x, t) = \frac{(x-x_0)(x_1-t)_+}{x_1-x_0} - (x-t)_+.$$

Prenons l'exemple numérique : $x_0 = 0, x_1 = 1, x = \frac{1}{2}$.

Dans ce cas, on a :

$$\phi\left(\frac{1}{2}, t\right) = \frac{1-t}{2} - \left(\frac{1}{2} - t\right)_+, \quad t \in [0, 1].$$

C'est à dire

$$\phi\left(\frac{1}{2}, t\right) = \begin{cases} \frac{t}{2} & \text{si } t \in [0, \frac{1}{2}] \\ \frac{1-t}{2} & \text{si } t \in [\frac{1}{2}, 1]. \end{cases}$$

Comme $\phi(\frac{1}{2}, t)$ est positive pour tout $t \in [0, 1]$ (condition nécessaire pour la formule de la moyenne), en utilisant la formule de la moyenne, on obtient

$$\begin{aligned} e\left(\frac{1}{2}\right) &= \int_0^1 \phi\left(\frac{1}{2}, t\right) f''(t) dt = \left[\int_0^1 \phi\left(\frac{1}{2}, t\right) dt \right] f''(\xi) \\ &= \frac{1}{8} f''(\xi), \quad \text{où } \xi \in [0, 1]. \end{aligned}$$

Ainsi, on a

$$e\left(\frac{1}{2}\right) \leq \frac{1}{8} \max_{t \in [0, 1]} |f''(t)|$$

La même méthode peut être employée dans le cas général :

Supposons que l'on remplace f par un polynôme d'interpolation P_{k-1} de degré $k-1$ basé sur les abscisses x_0, \dots, x_{k-1} de $[a, b]$. Si f admet une dérivée k^{e} continue, l'erreur correspondante en $x \in [a, b]$, $e(x) = P_{k-1}(x) - f(x)$, peut s'écrire :

$$e(x) = \int_a^b \phi(x, t) f^{(k)}(t) dt$$

$$\text{avec } \phi(x, t) = \sum_{i=0}^{k-1} l_i(x) \frac{(x_i - t)_+^{(k-1)}}{(k-1)!} - \frac{(x-t)_+^{(k-1)}}{(k-1)!}$$

où $l_i(x)$ représente le i^{e} me polynôme de la base de Lagrange (de degré $k-1$) correspondant aux abscisses x_0, \dots, x_{k-1} .

Exercice : L'erreur peut être également évaluée en utilisant la formule de Newton.

On a $f(x) = f(x_0) + (x - x_0)\delta_{x_0 x_1}^1 f + (x - x_0)(x - x_1)\delta_{x_0 x_1 x}^2 f$ (formule de Newton à l'ordre 1),

et le polynôme d'interpolation est donné par

$$P_1(x) = f(x_0) + (x - x_0)\delta_{x_0 x_1}^1 f$$

$$(\delta_{x_0 x_1 x}^2 P_1 = 0).$$

On a donc

$$e(x) = P_1(x) - f(x) \\ - \delta_{x_0 x_1 x}^2 f(x - x_0)(x - x_1).$$

Comme on sait que $\delta_{x_0 x_1 x}^2 f = \frac{1}{2!} f''(\xi)$, où $\xi \in [x_0, x_1]$, dépendant de x_0, x_1, x , on a :

$$|e(x)| = \left| \frac{(x - x_0)(x - x_1)}{2!} \right| |f''(\xi)|$$

Exemple numérique : Si l'on prend à nouveau $x_0 = 0, x_1 = 1, x = \frac{1}{2}$, on retrouve le résultat déjà obtenu

$$|e(x)| = \frac{1}{8} |f''(\xi)| \leq \frac{1}{8} \max_{t \in [0,1]} |f''(t)|$$

Plus généralement, pour l'interpolation de f par un polynôme de degré $k - 1$ basé sur x_0, \dots, x_{k-1} , on a :

$$e(x) = P_{k-1}(x) - f(x) = \prod_{i=0}^{k-1} (x - x_i) \frac{f^{(k)}(\xi)}{k!}$$

II. Interpolation Spline

2.1 NOTATIONS

Soit $\Delta = \{t_0, t_1, \dots, t_{n+1}\}$ une subdivision de l'intervalle $[a, b]$:

$$a = t_0 < t_1 < \dots < t_n < t_{n+1} = b.$$

On note Π_k^n l'espace des fonctions qui sont formées de polynômes de degré m sur chaque sous intervalle $[t_i, t_{i+1}]$ et qui appartiennent à $\mathcal{C}^k[a, b]$.

On remarque que si $k = m$, (ou même $k \geq m$) alors on a un seul polynôme de degré m sur $[a, b]$. En revanche, si $k < m$, alors, on peut calculer la dimension de Π_k^m de la manière suivante :

Il y a $n + 1$ intervalles et chaque polynôme dépend de $m + 1$ coefficients, d'où $(n + 1)(m + 1)$ degrés de liberté. Mais il y a n noeuds intérieurs et le raccordement des dérivées impose $k + 1$ conditions à chaque noeud, d'où $n(k + 1)$ conditions.

La dimension de Π_k^m est donc $(n + 1)(m + 1) - n(k + 1)$.

Exemple:

$$\dim(\Pi_{m-1}^m) = n + m + 1$$

$$\dim(\Pi_2^3) = n + 4$$

$$\dim(\Pi_{m-2}^m) = 2n + m + 1$$

2.2 BASE THÉORIQUE POUR Π_k^m

Considérons les primitives de $\mathbb{1}_{[t_i, +\infty[}$. La primitive d'ordre m vaut 0 sur $] - \infty, t_i]$ et $\frac{(t-t_i)_+^m}{m!}$ sur $[t_i, +\infty[$. On les note :

$$\delta_i(t) = \frac{(t-t_i)_+^m}{m!}$$

avec $(t-t_i)_+ = \min(0, (t-t_i))$.

— Dessin —

Un élément de Π_{m-1}^m peut être représenté de la manière suivante :

$$s(t) = P_m(t) + \sum_{i=1}^n d_i \delta_i(t),$$

où P_m est un polynôme de degré m et $d_i = s^{(m)}(t_i^+) - s^{(m)}(t_i^-)$.

La fonction δ_i est en quelque sorte la fonction de base pour une discontinuité de la dérivée $m^{\text{ème}}$ en t_i :

En effet, la dérivée $m^{\text{ème}}$ de s est une fonction en escalier (s est un polynôme de degré m à l'intérieur des $]t_i, t_{i+1}[$), donc une combinaison linéaire de $\mathbb{1}_{[t_i, +\infty[}$. En intégrant m fois, on tombe sur la somme des δ_i , et les constantes d'intégrations successives donnent le polynôme de degré $m-1$.

On voit bien que s dépend de $n+m+1$ coefficients.

De la même façon, un élément de Π_{m-2}^m peut être écrit sous la forme suivante :

$$s(t) = P_m(t) + \sum_{i=1}^n d_i \delta_i(t) + \sum_{i=1}^n \tilde{d}_i \tilde{\delta}_i(t),$$

avec $\tilde{\delta}_i(t) = \frac{(t-t_i)_+^{m-1}}{(m-1)!}$ et $\tilde{d}_i = s^{(m-1)}(t_i^+) - s^{(m-1)}(t_i^-)$. Cette expression dépend de $2n+m+1$ coefficients.

Remarques:

La base précédente est de nature théorique, Elle ne convient pas à un calcul effectif car elle conduit à des erreurs numériques inacceptables.

2.3 CONSIDÉRONS Π_2^3 , dont la dimension est $n + 4$

$$s(t) = P_3(t) + \sum_{i=1}^n d_i \delta_i(t) \quad \text{avec } \delta_i(t) = \frac{(t - t_i)_+^3}{3!}.$$

La dérivée 3^{ème}, $s^{(3)}(t)$ a l'allure suivante :

— Dessin —

Pour $s \in \Pi_2^3$ et $x \in H^2[a, b]$ (l'espace des fonctions telles que x' est absolument continue et x'' est de carré sommable sur $[a, b]$), formons

$$\int_a^b s''(t)x''(t) dt.$$

En faisant une intégration par parties, on obtient :

$$\begin{aligned} \int_a^b s''(t)x''(t) dt &= \sum_{i=0}^n \int_{t_i}^{t_{i+1}} s''(t)x''(t) dt \\ &= \sum_{i=0}^n [s''(t)x'(t)]_{t_i}^{t_{i+1}} - \sum_{i=0}^n \int_{t_i}^{t_{i+1}} s'''(t)x'(t) dt \\ &= \sum_{i=0}^n [s''(t)x'(t)]_{t_i}^{t_{i+1}} - \sum_{i=0}^n [s'''(t)x(t)]_{t_i}^{t_{i+1}}, \end{aligned}$$

car $s^{(4)} = 0$ pour tout $t \in]t_i, t_{i+1}[$. De plus, comme x et x' sont continues en t_i , on peut réarranger la somme :

$$\begin{aligned} \int_a^b s''(t)x''(t) dt &= s''(b)x'(b) - \sum_{i=1}^n \Delta s''(t_i)x'(t_i) - s''(a)x'(a) \\ &\quad - s'''(b)x(b) + \sum_{i=1}^n \Delta s'''(t_i)x(t_i) + s'''(a)x(a). \end{aligned}$$

Si $s \in \Pi_3^2$ (s'' continue en t_i , donc $\Delta s''(t_i) = 0$), on obtient :

$$\begin{aligned} \int_a^b s''(t)x''(t) dt &= \sum_{i=1}^n \Delta s'''(t_i)x(t_i) + s''(b)x'(b) - s''(a)x'(a) \\ &\quad - s'''(b)x(b) + s'''(a)x(a). \end{aligned}$$

Appelons S_2 l'ensemble des éléments de Π_2^3 qui sont polynômes de degré 1 sur $[a, t_1]$ et $[t_n, b]$, i.e. tels que $s''(a) = s'''(a) = 0$ et $s''(b) = s'''(b) = 0$.

On a donc, pour tout $s \in S_2$, et pour tout $x \in H^2[a, b]$:

$$\int_a^b s''(t)x''(t) dt = \sum_{i=1}^n d_i x(t_i) \quad \text{avec} \quad d_i = \Delta s'''(t_i)$$

Définition

Notons S_q l'espace des fonctions $s \in \Pi_{2q-2}^{2q-1}$ qui sont de degré $q-1$ sur les intervalles $[a, t_1]$ et $[t_n, b]$.

Théorème Pour tout $s \in S_q$, et tout $x \in H^q[a, b]$, on a :

$$\int_a^b s^{(q)}(t)x^{(q)}(t) dt = (-1)^q \sum_{i=1}^n d_i x(t_i)$$

où $d_i = \Delta s^{(2q-1)}(t_i)$.

La démonstration est une généralisation de celle donnée ci-dessus.

On remarque que $\dim S_q = n$ (si $n \geq q$)

2.4 REPRÉSENTATION DES FONCTIONS DE S_q ($\subset \Pi_{2q-2}^{2q-1}$) :

On applique le même raisonnement que pour les fonctions de Π_{2q-2}^{2q-1} : La dérivée $2q-1$ ^{ème} est en escalier, et en intégrant, les constantes d'intégrations successives donnent un polynôme qui est forcément de degré $q-1$, puisqu'il l'est sur $[a, t_1]$.

\implies Tout élément de S_q peut être mis sous la forme :

$$s(t) = P_{q-1}(t) + \sum_{i=1}^n d_i \delta_i(t)$$

où $\delta_i(t) = \frac{(t-t_i)_+^{2q-1}}{(2q-1)!}$, $d_i = \Delta s^{(2q-1)}(t_i)$, et P_{q-1} est un polynôme de degré $q-1$.

\Leftarrow Mais inversement, tout élément de cette forme n'appartient pas forcément à S_q : Il faut vérifier que s est un polynôme de degré $q-1$ aussi dans l'intervalle $[t_n, b]$, c'est à dire:

$$S^{(q)}(t) = 0 \text{ pour tout } t \in [t_n, b].$$

C'est à dire

$$\sum_{i=0}^n d_i \frac{(t - t_i)_+^{q-1}}{(q-1)!} = 0$$

qui est équivalent à

$$\sum_{i=0}^n d_i \frac{(t - t_i)^{q-1}}{(q-1)!} = 0.$$

(On enlève le “+”)

Pour exprimer qu'un polynôme de degré $q - 1$ est en fait identiquement nul, il suffit d'exprimer que toutes ses dérivées jusqu'à l'ordre $q - 1$ en un même point sont nulles (cf. développements de Taylor). Cela donne en 0 :

$$\sum_{i=1}^n d_i (t_i)^k = 0 \quad \text{pour } k = 0, \dots, q - 1,$$

d'où le résultat suivant:

Théorème Soit $s \in \Pi_{2q-2}^{2q-1}$

$$s \in S_q \Leftrightarrow s(t) = P_{q-1}(t) + \sum_{i=1}^n d_i \delta_i(t)$$

avec P_{q-1} polynôme de degré $q - 1$ et $\sum_{i=1}^n d_i (t_i)^k = 0$, $k = 0, \dots, q - 1$.

2.5 INTERPOLATION

Théorème Si $n \geq q$, quels que soient les points z_1, \dots, z_n , il existe $\sigma \in S_q$ unique tel que $\sigma(t_i) = z_i$, $i = 1, \dots, n$.

Démonstration :

En prenant la représentation précédente,

$$\sigma(t) = \sum_{i=1}^n d_i \sigma_i(t) + \sum_{i=1}^{q-1} \alpha_i t^i,$$

on doit déterminer les $n + q$ coefficients d_i et α_i . Les conditions

$$\begin{aligned} \sigma(t_i) &= z_i, \quad i = 1, \dots, n \\ \sum_{i=1}^n d_i (t_i)^k &= 0, \quad k = 0, \dots, q - 1 \end{aligned}$$

forment un système linéaire de dimension $n+q$. Ce système a une solution unique si et seulement si le système homogène correspondant (en remplaçant les z_i par zéro) n'admet que la solution nulle.

$$\begin{aligned}\sigma(t_i) &= 0, \quad i = 1, \dots, n \\ \sum_{i=1}^n d_i(t_i)^k &= 0, \quad k = 0, \dots, q-1.\end{aligned}$$

En admettant que les fonctions $\delta_i, i = 1, \dots, n$ et $t^i, i = 0, \dots, q-1$ sont indépendantes, il revient au même de démontrer que

$$\sigma_0 \in S_q, \quad \sigma_0(t_i) = 0, \quad i = 1, \dots, n \quad \Rightarrow \quad \sigma_0 = 0.$$

C'est là qu'intervient le théorème de la partie précédente :

$$\int_a^b \underbrace{\sigma_0^{(q)}(t)}_{\in S_q} \underbrace{\sigma_0^{(q)}(t)}_{\in H^q} dt = (-1)^q \sum_{i=1}^n \Delta \sigma_0^{2q-1}(t_i) \underbrace{\sigma_0(t_i)}_{=0} = 0.$$

Donc, $\sigma_0^{(q)} = 0$. On sait déjà que σ_0 est un polynôme de degré $q-1$. Il a n racines. Comme $n \geq q$, on a $\sigma_0 = 0$. \square

2.6 PROPRIÉTÉ VARIATIONNELLE

Théorème On suppose $n \geq q$. L'élément $\sigma \in S_q$ tel que $\sigma(t_i) = z_i$ pour $i = 1, \dots, n$ vérifie :

$$\int_a^b \left(\sigma^{(q)}(t) \right)^2 dt \leq \int_a^b \left(x^{(q)}(t) \right)^2 dt.$$

pout tout $x \in I_z := \{x \in H^q[a, b] \mid x(t_i) = z_i, \quad i = 1, \dots, n\}$.

Démonstration :

Il suffit d'écrire $x = \sigma + (x - \sigma)$ et de décomposer le carré :

$$\int \left(x^{(q)} \right)^2 = \int \left(\sigma^{(q)} \right)^2 + \underbrace{\int \left(x^{(q)} - \sigma^{(q)} \right)^2}_{\geq 0} + 2 \underbrace{\int \sigma^{(q)} \left(x^{(q)} - \sigma^{(q)} \right)}_{=0}.$$

La dernière intégrale est nulle car

$$\int \underbrace{\sigma^{(q)}}_{\in S_q} \underbrace{\left(x^{(q)} - \sigma^{(q)} \right)}_{\in H^q} = (-1)^q \sum \Delta \sigma^{(2q-1)}(t_i) \underbrace{\left(x(t_i) - \sigma(t_i) \right)}_{=0} = 0.$$

\square

Ainsi, l'unique solution σ du problème

$$\begin{aligned} & \text{Minimiser} && \int_a^b \left(x^{(q)}(t) \right) dt \\ & x \in H^q[a, b] \\ & x(t_i) = z_i, \quad i = 1, \dots, n \end{aligned}$$

est caractérisée par les conditions :

- 1) La restriction de σ à chaque intervalle $[t_i, t_{i+1}]$ est un polynôme de degré $2q - 1$ ($i = 1, \dots, n - 1$).
- 2) Les restrictions $\sigma|_{[a, t_1]}$ et $\sigma|_{[t_n, b]}$ sont de degré $q - 1$.
- 3) $\sigma \in \mathcal{C}^{2q-2}[a, b]$, i.e. $\sigma^{(j)}$ est continue en t_1, \dots, t_n pour $j = 0, \dots, 2q - 2$.
- 4) $\sigma(t_i) = z_i, i = 1, \dots, n$.

Les conditions 1) 2) et 3) caractérisent S_q . La dernière est la condition d'interpolation.

2.7 ALGORITHME: CALCUL DE SPLINES CUBIQUES $S_2 \subset \Pi_2^3$

Prenons d'abord l'exemple de base correspondant à la minimisation de $\int (x'')^2$ avec les conditions $x(t_i) = z_i, i = 1, \dots, n$. On cherche l'élément $\sigma \in S_2$ vérifiant $\sigma(t_i) = z_i, i = 1, \dots, n$.

Notons P_i le polynôme de degré 3 qui coïncide avec σ dans l'intervalle $[t_i, t_{i+1}], i = 0, \dots, n$.

On pourrait chercher pour chaque P_i les coefficients dans la base canonique, mais il est plus économique de procéder autrement : Choisir la base de Hermite de \mathcal{P}_3 :

- $x(x - 1)^2$
- $x^2(x - 1)$
- $2x^3 + 3x^2 + 1$
- $2(1 - x)^3 + 3(1 - x)^2 + 1$.

Ainsi, on obtient l'équation du polynôme directement en fonction des quatre valeurs :

- $P_i(t_i)$
- $P_i(t_{i+1})$
- $P_i'(t_i)$
- $P_i'(t_{i+1})$.

Les valeurs $P_i(t_i) = z_i$ et $P_i(t_{i+1}) = z_{i+1}$ ($i = 1, \dots, n$) sont connues. Il reste donc à déterminer les valeurs $z_i' = P_i'(t_i) = \sigma_i'(t_i)$ pour $i = 1, \dots, n$.

Par construction, on a la même valeur z_i en t_i pour $P_i(t_i)$ et $P_{i-1}(t_i)$, donc, la continuité de σ en t_i est bien assurée. De même, on a la même valeur z_i' en t_i pour $P_i'(t_i)$ et $P_{i-1}'(t_i)$, donc, la continuité de σ' en t_i est bien assurée.

Reste la continuité de σ'' . C'est la condition qui va fournir les équations permettant de calculer les z_i' : il faut assurer

$$P_{i-1}''(t_i) = P_i''(t_i) \quad i = 1, \dots, n \quad (*)$$

La formule de Newton-Hermite donne pour P_i :

$$P_i(t) = z_i + z'_i(t - t_i) + \frac{m_i - z'_i}{h_i}(t - t_i)^2 + \frac{z'_{i+1} - 2m_i + z'_i}{h_i}(t - t_i)^2(t - t_{i+1})$$

avec $h_i = t_{i+1} - t_i$ et $m_i = \frac{z_{i+1} - z_i}{h_i}$.

On a donc, pour $i = 1, \dots, n - 1$:

$$\begin{aligned} P_i''(t_i) &= \frac{2}{h_i}(3m_i - 2z'_i - z'_{i+1}) \\ P_i''(t_{i+1}) &= \frac{2}{h_i}(-3m_i + z'_i + 2z'_{i+1}) \end{aligned}$$

et donc aussi, pour $i = 2, \dots, n$

$$P_{i-1}''(t_i) = \frac{2}{h_{i-1}}(-3m_{i-1} + z'_{i-1} + 2z'_i).$$

La continuité de σ'' en t_i ($i = 2, \dots, n - 1$) sera assurée si l'on a (*)

$$\frac{2}{h_{i-1}}(-3m_{i-1} + z'_{i-1} + 2z'_i) = \frac{2}{h_i}(3m_i - 2z'_i - z'_{i+1}).$$

Ce qui s'écrit encore

$$\frac{1}{h_{i-1}}z'_{i-1} + \left(\frac{2}{h_{i-1}} + \frac{2}{h_i}\right)z'_i + \frac{1}{h_i}z'_{i+1} = \frac{3}{h_{i-1}}m_{i-1} + \frac{3}{h_i}m_i, \quad i = 2, \dots, n - 1 \quad (1)$$

Sachant que σ est de degré $q - 1 = 1$ sur les deux intervalles des bords, on peut écrire les conditions du premier et du dernier point.

En t_1 , on écrit simplement

$$P_1''(t_1) = P_0''(t_1) = 0$$

ce qui donne

$$2z'_1 + z'_2 = 3m_1. \quad (2)$$

De même, en t_n , on a

$$P_n''(t_n) = P_{n-1}''(t_n) = 0$$

d'où

$$3z'_{n-1} + 2z'_n = 3m_{n-1}. \quad (3)$$

On a donc les équations (1)-(3) qui permettent de calculer les z'_i : Il suffit de résoudre le système linéaire.

Dans le cas d'abscisses t_i équidistantes ($h_i = h$), ces équations se simplifient de la façon suivante :

$$\begin{cases} 2z'_1 + z'_2 = \frac{3}{h}(z_2 - z_1) \\ z'_{i-1} + 4z'_i + z_i = \frac{3}{h}(z_{i+1} - z_{i-1}) & i = 2, \dots, n - 1 \\ z'_{n-1} + 2z'_n = \frac{3}{h}(z_n - z_{n-1}) \end{cases}$$

La matrice du système est tridiagonale et symétrique.

III. DESIGN POLYNOMIAL — COURBES DE BEZIER

3.1 COURBES DE BEZIER

Supposons que l'on veuille représenter une courbe sous forme paramétrique par une fonction polynomiale :

$$\mathbf{x} : [a, b] \longrightarrow \mathbb{R}^d.$$

Si \mathbf{x} est exprimée dans la base canonique

$$\mathbf{x}(t) = \sum_{i=0}^n \mathbf{a}_i t^i, \quad \mathbf{a}_i \in \mathcal{E}^d, \quad i = 0, \dots, n,$$

il est très difficile de prévoir l'allure de la courbe $(\mathcal{C}) = \mathbf{x}([a, b])$ en fonction de la position des points $\mathbf{a}_0, \dots, \mathbf{a}_n$.

De même si l'on bouge l'un des points \mathbf{a}_i , il est pratiquement impossible de prévoir le déplacement de la courbe qui va en résulter.

Par contre si on change de base, par exemple si l'on prend la *base de Bernstein* :

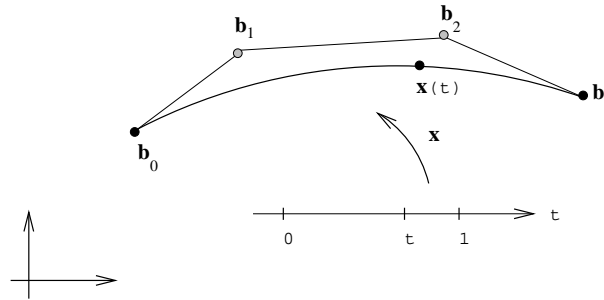
$$B_i^n(t) = \binom{n}{i} (1-t)^{n-i} t^i, \quad \text{où } \binom{n}{i} = \frac{n!}{(n-i)!i!};$$

$$\mathbf{x}(t) = \sum_{i=0}^n \mathbf{b}_i B_i^n(t)$$

s'appelle *courbe de Bézier* et le polygone défini par les points \mathbf{b}_i donne une bonne idée intuitive de la courbe $(\mathcal{C}) = \mathbf{x}([0, 1])$.

Pour un intervalle $[a, b]$ quelconque, il faut simplement faire un changement de variable $u(t) = \frac{t-a}{b-a}$ pour revenir à la définition précédente.

Exemple : $n = 3, d = 2$



Les points $\mathbf{b}_i, i = 0, \dots, n$ s'appellent points de Bézier (ou points de contrôle) et forment le polygone de contrôle.

3.2 PROPRIETES DES POLYNOMES DE BERNSTEIN

Sans restriction, supposons $t \in [0, 1]$.

(1) **Partition de l'unité :** $\sum_{i=0}^n B_i^n(t) \equiv 1, \forall t \in [0, 1]$.

On obtient ce résultat par la formule du binôme :

$$\begin{aligned}
 1 &\equiv [(1-t) + t]^n \equiv (1-t)^n + \binom{n}{1}(1-t)^{n-1}t + \dots + \binom{n}{i}(1-t)^{n-i}t^i + \dots + t^n \\
 &\equiv \sum_{i=0}^n B_i^n(t).
 \end{aligned}$$

(2) **Positivité :** $B_i^n(t) \geq 0, \forall t \in [0, 1], 0 \leq i \leq n$.

(3) **Symétrie :** $B_i^n(t) = B_{n-i}^n(1-t)$

$$\begin{aligned}
 B_i^n(t) &= \binom{n}{i}(1-t)^{n-i}t^i = \binom{n}{n-i}(1 - (1-t))^{n-(n-i)}(1-t)^{n-i} \\
 &= B_{n-i}^n(1-t).
 \end{aligned}$$

(4) **Formule de récurrence :**

$$\begin{aligned}
 &(1-t)B_i^{n-1}(t) \quad i = 0 \\
 B_i^n(t) &= (1-t)B_i^{n-1}(t) + tB_{i-1}^{n-1}(t) \quad i = 1, \dots, n-1 \\
 &tB_{i-1}^{n-1}(t) \quad i = n
 \end{aligned}$$

et $B_0^0(t) \equiv 1$, $B_i^n(t) \equiv 0$ si $i \notin \{0, \dots, n\}$.

On sait que pour $i = 1, \dots, n-1$, on a

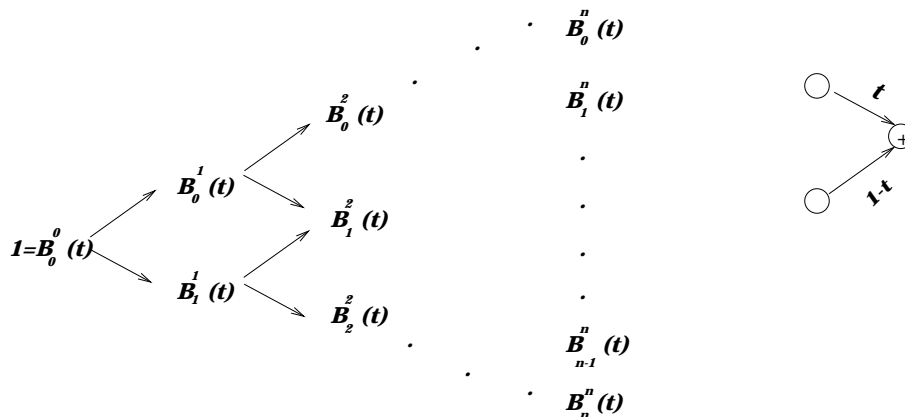
$$\binom{n}{i} = \binom{n-1}{i} + \binom{n-1}{i-1}.$$

C'est une formule classique de combinatoire :

$$\begin{aligned} \binom{n-1}{i} + \binom{n-1}{i-1} &= \frac{(n-1)!}{(n-1-i)!i!} + \frac{(n-1)!}{(n-i)!(i-1)!} \\ &= \frac{(n-1)!(n-i)}{(n-i)!i!} + \frac{(n-1)!i}{(n-i)!i!} \\ &= \frac{(n-1)!(n)}{(n-i)!i!} = \binom{n}{i}. \end{aligned}$$

$$\begin{aligned} B_i^n(t) &= \binom{n}{i} t^i (1-t)^{n-i} \\ &= \binom{n-1}{i} t^i (1-t)^{n-i} + \binom{n-1}{i-1} t^i (1-t)^{n-i} \\ &= (1-t) \left[\binom{n-1}{i} t^i (1-t)^{n-1-i} \right] + t \left[\binom{n-1}{i-1} t^{i-1} (1-t)^{n-i} \right] \\ &= (1-t) B_i^{n-1}(t) + t B_{i-1}^{n-1}(t). \end{aligned}$$

La formule de récurrence permet de calculer les $B_i^n(t)$ avec le *tableau triangulaire*, dit "d'Aitken/Mansfield" suivant :



Les $B_i^n(t)$ peuvent ainsi être construits par interpolation linéaire répétée.

(5) le maximum : $\max_{t \in [0,1]} B_i^n(t) = \frac{n}{i}$.

(6) Valeurs aux extrémités : $B_i^n(0) = \delta_{i,0}$, $B_i^n(1) = \delta_{i,n}$.

(7) **Base polynomiale** : $\{B_i^n, i = 0, \dots, n\}$ constitue une base pour \mathcal{P}_n .

$B_i^n(t)$ est un polynôme de degré n a une racine $i^{\text{ème}}$ en $t = 0$ et une racine $(n - i)^{\text{ème}}$ en $t = 1$, donc

$$\begin{cases} (B_i^n)^{(j)}(0) = 0, \text{ pour } j = 0, \dots, i - 1 \\ (B_i^n)^{(i)} \neq 0. \end{cases}$$

Supposons $f(t) = \sum_{i=0}^n \alpha_i B_i^n(t) = 0$, on a

$$\begin{aligned} f(0) &= \alpha_0 B_0^n(0) + \alpha_1 B_1^n(0) + \dots + \alpha_n B_n^n(0) \\ &= \alpha_0 \underbrace{B_0^n(0)}_{\neq 0} = 0 \implies \alpha_0 = 0 \end{aligned}$$

$$\begin{aligned} f'(0) &= \alpha_1 (B_1^n)'(0) + \dots + \alpha_n (B_n^n)'(0) \\ \alpha_1 (B_1^n)'(0) + 0 + \dots + 0 &= 0 \implies \alpha_1 = 0 \end{aligned}$$

⋮

$$\begin{aligned} f^{(r)}(0) &= \alpha_r (B_r^n)^{(r)}(0) + \dots + \alpha_n (B_n^n)^{(r)}(0) \\ \alpha_r (B_r^n)^{(r)}(0) + 0 + \dots + 0 &= 0 \implies \alpha_r = 0 \end{aligned}$$

(8) **Dérivée** :

$$(B_i^n)'(t) = \begin{cases} -nB_i^{n-1}(t) & i = 0 \\ n(B_{i-1}^{n-1}(t) - B_i^{n-1}(t)) & i = 1, \dots, n - 1 \\ nB_{i-1}^{n-1}(t) & i = n \end{cases}$$

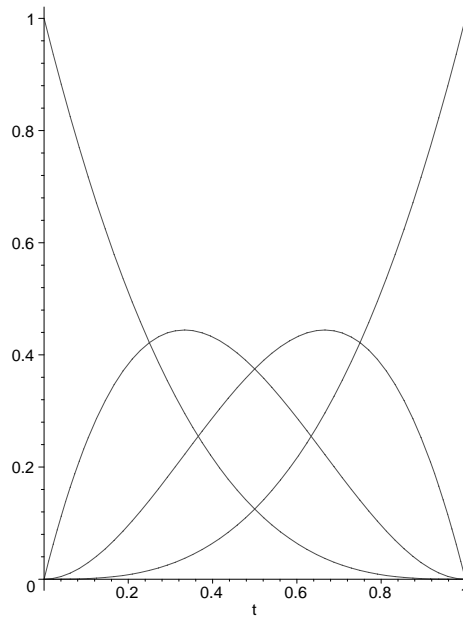
$$(B_i^n)'(t) = i \binom{n}{i} t^{i-1} (1-t)^{n-i} - (n-i) \binom{n}{i} t^i (1-t)^{n-1-i}$$

$$= \frac{i \cdot n!}{i!(n-i)!} t^{i-1} (1-t)^{n-i} - \frac{(n-i) \cdot n!}{i!(n-i)!} t^i (1-t)^{n-1-i}$$

$$= n \frac{(n-1)!}{(i-1)!(n-i)!} t^{i-1} (1-t)^{n-1-(i-1)} - n \frac{(n-1)!}{i!(n-1-i)!} t^i (1-t)^{n-1-i}$$

$$= n(B_{i-1}^{n-1}(t) - B_i^{n-1}(t)).$$

Exemple : $n = 3$



$$\begin{aligned}
 B_0^3 &= (1 - t)^3 \\
 B_1^3 &= 3(1 - t)^2 t \\
 B_2^3 &= 3(1 - t) t^2 \\
 B_3^3 &= t^3
 \end{aligned}$$

3.3 ALGORITHME DE DE CASTELJAU (1959)

Historiquement, c'est avec cet algorithme que les travaux de M. de Casteljaou commençaient en 1959 chez Citron. Ils étaient publiés comme des rapports techniques, tenus très au secret par Citron.

Ces travaux restaient inconnus jusqu'en 1975 quand W. Böhm en a pris connaissance et les a rendu public.

Au début des années 60, Pierre Bézier commençait ses travaux le système UNISURF chez Renault et les publiait, d'où le fait que le nom de Bézier a été donné à ces courbes et surfaces de forme libre.

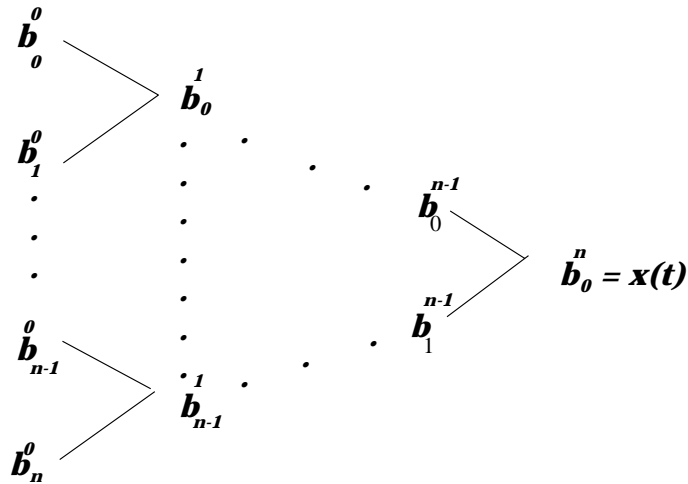
L'algorithme de de Casteljaou sert à évaluer $\mathbf{x}(t)$ en un paramètre t donné avec les points auxiliaires de la manière suivante :

$ \begin{aligned} \mathbf{b}_i^k &= (1 - t)\mathbf{b}_i^{k-1} + t\mathbf{b}_{i+1}^{k-1} \\ \mathbf{b}_i^0 &= \mathbf{b}_i, \quad i = 0, \dots, n \end{aligned} $	$k = 1, \dots, n, \quad i = 0, \dots, n - k.$
---	---

Il est facile à démontrer que $\mathbf{x}(t) = \mathbf{b}_0^n$.

L'algorithme de de Casteljaou est un algorithme triangulaire à n niveau

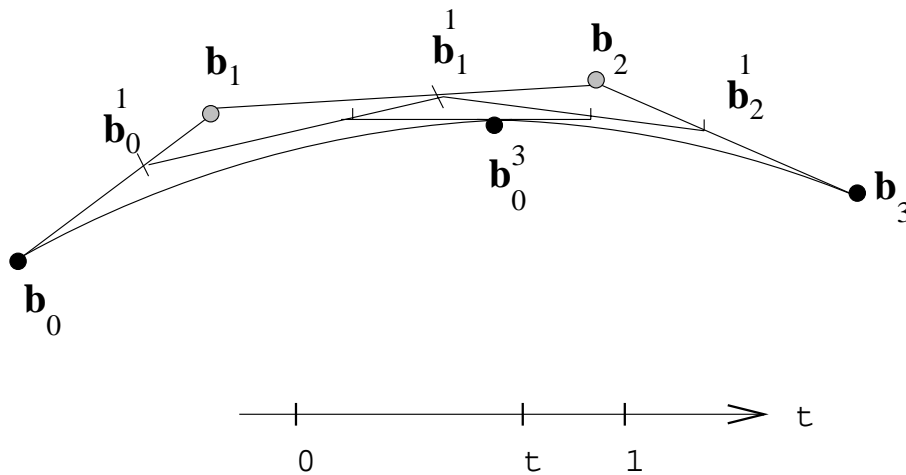
Schéma :



Démonstration :

$$\begin{aligned}
 \mathbf{x}(t) &= \mathbf{b}_0 B_0^n(t) + \mathbf{b}_1 B_1^n(t) + \dots + \mathbf{b}_n B_n^n(t), \quad t \in [0, 1] \\
 &\quad (\text{Aitken/Mansfield ont exprimé } B_i^n \text{ en fonction des } B_j^{n-1}) \\
 &= \mathbf{b}_0(1-t)B_0^{n-1}(t) + \mathbf{b}_1((1-t)B_1^{n-1}(t) + tB_0^{n-1}(t)) + \dots + \mathbf{b}_n tB_{n-1}^{n-1}(t) \\
 &\quad (\text{trier par rapport à } B_i^{n-1}) \\
 &= \underbrace{[(1-t)\mathbf{b}_0 + t\mathbf{b}_1]}_{=: \mathbf{b}_0^1} B_0^{n-1}(t) + \dots + \underbrace{[(1-t)\mathbf{b}_{n-1} + t\mathbf{b}_n]}_{=: \mathbf{b}_{n-1}^1} B_{n-1}^{n-1}(t) \\
 &\quad \vdots \\
 &\quad (\text{répéter ce passage jusqu'à}) \\
 &= \underbrace{[(1-t)\mathbf{b}_0^{n-1} + t\mathbf{b}_1^{n-1}]}_{=: \mathbf{b}_0^n} \underbrace{B_0^0(t)}_{\equiv 1} \\
 &= \mathbf{b}_0^n
 \end{aligned}$$

Graphiquement :



3.4 PROPRIETES GEOMETRIQUES DES COURBES DE BEZIER

(a) Propriété d’enveloppe convexe :

pour $t \in [0, 1]$, la courbe \mathcal{C} se trouve à l’intérieur de l’enveloppe convexe (intersection de tous les ensembles convexes contenant $\{\mathbf{b}_i\}$) des \mathbf{b}_i .

C’est une conclusion directe de la positivité et de la partition de l’unité des polynômes de Bernstein ((1) et (2)).

Chaque point intermédiaire \mathbf{b}_i^r dans l’algorithme de de Casteljau est une combinaison barycentrique convexe des points précédents \mathbf{b}_j^{r-1} . cela implique par exemple que chaque polygone de contrôle plan génère une courbe plane.

L’importance de cette propriété se trouve dans ce qu’on appelle “*test de collision*” :

Si 2 courbes de Bézier représentent le chemin d’un bras de robot, peut-il y avoir une collision ?

Au lieu de calculer les intersections possibles, on fait le test des “bounding boxes” ou des boîtes englobantes “minmax boxes”. (Arêtes parallèles aux axes du système de coordonnées).

C’est un test très rapide s’il y a des milliers d’orbites à tester.

(b) Invariance affine

Les applications affines sont incluses dans tous les systèmes de CAO : les objets doivent être tournés, repositionnés, chargés de taille, etc.

Les courbes de Bézier sont affinesment invariantes, ce qui implique que les deux procédures suivantes mènent au même résultat :

1. calculer d’abord le point de la courbe $\mathbf{x}(\hat{t})$ et puis appliquer la transformation affine,
2. appliquer d’abord la transformation affine aux points de contrôle et puis évaluer $\mathbf{x}(\hat{t})$.

— dessin —

L’invariance affine est une conséquence directe de l’algorithme de de Casteljau qui consiste en la répétition d’interpolations linéaires entre deux points (c.à.d. d’applications affines).

ces opérations elles mêmes sont affinement invariantes, donc leur composition l'est aussi

$$f \text{ affine} \iff f(\alpha \mathbf{a} + (1 - \alpha)\mathbf{b}) = \alpha f(\mathbf{a}) + (1 - \alpha)f(\mathbf{b})$$

Par contre, les courbes de Bézier ne sont pas invariables par projection. Les applications projectives sont par exemple utilisée en infographie pour le rendu réaliste d'objets 3D \implies application perspective.

Une transformation affine est une application T qui applique \mathcal{E}^3 (espace affine euclidien tridimensionnel) sur lui même et qui laisse les combinaisons barycentriques invariantes.

Combinaisons barycentrique (affine)

$$\mathbf{b} = \sum_{i=0}^n \alpha_i \mathbf{b}_i, \quad \mathbf{b}_i \in \mathcal{E}^3, \quad \alpha_0 + \dots + \alpha_n = 1$$

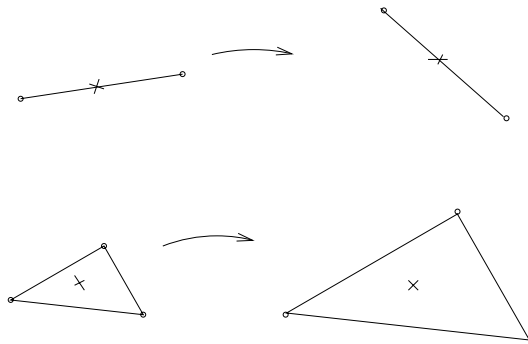
C'est une somme pondérée de points où la somme des poids est égale à 1, c'est l'opération fondamentale pour les points.

Si $\mathbf{x} = \sum_{i=0}^n \alpha_i \mathbf{a}_i$, $\mathbf{x}, \mathbf{b}_i \in \mathcal{E}^3$ et si T est une transformation affine, alors également

$$T\mathbf{x} = \sum_{i=0}^n \alpha_i T\mathbf{a}_i; \quad T\mathbf{x}, T\mathbf{a}_i \in \mathcal{E}^3.$$

Par exemple, au point milieu d'un segment de droite correspondra le milieu de l'image affine de ce segment de ligne droite.

A l'isobarycentre de n points correspond l'isobarycentre des points images.



Soit $\mathbf{x} \in \mathcal{E}^3$, une transformation affine est donnée par

$$T\mathbf{x} = \mathbf{A}\mathbf{x} + \mathbf{V} \quad \begin{array}{l} \text{quad } \mathbf{A} \text{ matrice } 3 \times 3 \\ \text{quad } \mathbf{V} \text{ vecteur de } \mathbb{R}^3. \end{array}$$

Vérification : (T préserve les combinaisons barycentriques ?)

$$\begin{aligned} T\left(\sum \alpha_i \mathbf{a}_i\right) &= \mathbf{A}\left(\sum \alpha_i \mathbf{a}_i\right) + \mathbf{V} \\ &= \sum \alpha_i \mathbf{A} \mathbf{a}_i + \sum \alpha_i \mathbf{V} && (\sum \alpha_i = 1) \\ &= \sum \alpha_i (\mathbf{A} \mathbf{a}_i + \mathbf{V}) \\ &= \sum \alpha_i T \mathbf{a}_i. \end{aligned}$$

Exemples de transformations affines :

- L'identité ($\mathbf{V} = \mathbf{0}$, $\mathbf{A} = \mathbf{I}$).
- La translation ($\mathbf{A} = \mathbf{I}$).
- L'homothétie ($\mathbf{V} = \mathbf{0}$, \mathbf{A} diagonale).
- La rotation ($\mathbf{V} = \mathbf{0}$, ex. rotation autour de l'axes des z) :

$$\mathbf{A} = \begin{pmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

- Le cisaillement

Transformations euclidiennes (\subset) transformations affines

$\mathbf{A}^T \mathbf{A} = \mathbf{I}$ (\mathbf{A} orthogonale).

Elles laissent les longueurs et les mesures d'angles inchangées (ex. rotation, translation).

Les transformation affines peuvent être combinées.

Une transformation compliquée peut être décomposée en une séquence de transformations plus simples.

Théorème : Chaque transformation affine peut être composée de translations, de rotations, de cisaillements et d'homothéties.

$Rang(\mathbf{A}) = 3$: transforme un objet tridimensionnel en un objet tridimensionnel

$Rang(\mathbf{A}) = 2$: T est une projection parallèle sur un plan

$Rang(\mathbf{A}) = 1$: T est une projection parallèle sur une ligne droite.

$T : \mathcal{E}^2 \longrightarrow \mathcal{E}^2$ est uniquement déterminée par un triangle et son image.

\implies 2 triangles quelconques déterminent une transformation du plan dans lui même.

$T : \mathcal{E}^3 \longrightarrow \mathcal{E}^3$ est uniquement déterminée par un tétraèdre et son image.

L'interpolation linéaire : $\mathbf{x} = \mathbf{x}(t) = (1-t)\mathbf{a} + t\mathbf{b}$; $t \in \mathbb{R}$ est affinement invariable (car c'est une combinaison barycentrique).

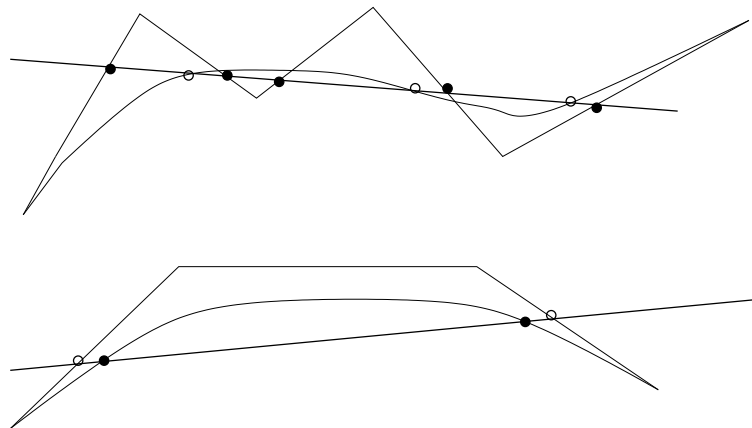
Démonstration : soit $T : \mathcal{E}^3 \rightarrow \mathcal{E}^3$ une transformation affine, alors :

$$\begin{aligned}
 T\mathbf{x} &= (1-t)T\mathbf{a} + tT\mathbf{b} \\
 &= \mathbf{A}((1-t)\mathbf{a} + t\mathbf{b}) + \mathbf{V} \\
 &= (1-t)\mathbf{A}\mathbf{a} + t\mathbf{A}\mathbf{b} + (1-t)\mathbf{V} + t\mathbf{V} \\
 &= (1-t)(\mathbf{A}\mathbf{a} + \mathbf{V}) + t(\mathbf{A}\mathbf{b} + \mathbf{V}) \\
 &= (1-t)T\mathbf{a} + tT\mathbf{b}
 \end{aligned}$$

(c) Variation Diminishing property (Diminution de variation)

Soit \mathcal{C} une courbe de Bézier et \mathcal{P} sont polygone de contrôle. Si l'on dénote $cross\mathcal{C}$ ou $cross\mathcal{P}$ respectivement le nombre d'intersection de \mathcal{C} ou \mathcal{P} avec un plan quelconque, alors

$$cross\mathcal{C} \leq cross\mathcal{P}.$$



aucune droite n'intersecte \mathcal{C} plus souvent que \mathcal{P} .

Cela implique par exemple qu'une polygone convexe génère une courbe convexe.

Attention l'inverse n'est pas vraie.

Les deux propriétés de l'EC et de la VDP ensemble permettent de bien prédire la forme de la courbe.

La propriété suivante améliore encore la prédiction de la forme de la courbe à partir de son polygone de contrôle :

(d) Endpoint Interpolation (Interpolation des extrémités)

$$\mathbf{x}(0) = \mathbf{b}_0$$

$$\mathbf{x}(1) = \mathbf{b}_n$$

la courbe passe par \mathbf{b}_0 et \mathbf{b}_n . C'est une conséquence directe de la propriété (6) des polynômes de Bernstein :

$$B_i^n(0) = \delta_{i,0}$$

$$B_i^n(1) = \delta_{i,n}$$

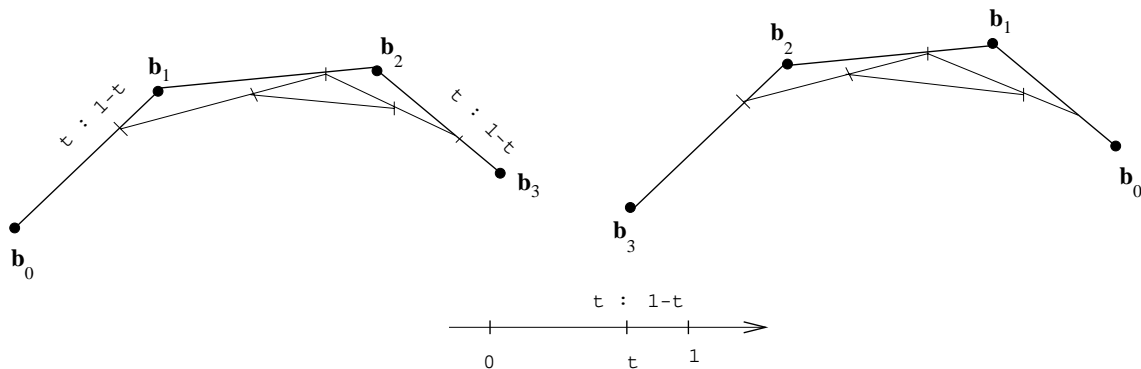
En phase de dessin, c'est une propriété importante de pouvoir contrôler directement les extrémités d'une courbe.

(e) Symétrie :

Les polynômes de Bernstein sont symétrique par rapport à t et $1 - t$, donc

$$\sum_{i=0}^n \mathbf{b}_i B_i^n(t) = \sum_{i=0}^n \mathbf{b}_{n-i} B_i^n(1-t)$$

ce qui change entre les deux courbes c'est leur *sens de parcours*.



(f) Contrôle pseudo-local :

Le polynôme de Bernstein n'a qu'un maximum, atteint en $t = \frac{i}{n}$.

Ceci a des applications en dessin : si nous déplaçant uniquement un des sommets du polygone de contrôle, par exemple \mathbf{b}_i , alors la courbe est principalement affectée par ce changement dans la région autour des points où le paramètre prend la valeur $\frac{i}{n}$.

Ceci rend l'effet du changement raisonnablement prévisible, bien que le changement affecte la courbe *globalement*.

(g) Invariance par transformations affines du paramètre :

Algébriquement $\sum_{i=0}^n \mathbf{b}_i B_i^n(t) = \sum_{i=0}^n \mathbf{b}_i B_i^n\left(\frac{u-a}{b-a}\right)$, $t \in [0, 1]$ et $u \in [a, b]$. (L'algorithme de de Casteljau est "aveugle" vis à vis l'intervalle du paramètre. La transition de l'intervalle $[0, 1]$ à $[a, b]$ est une carte affine).

(h) Dessiner avec les courbes de Bézier :

la courbe 'imite' le parcours du polygone de contrôle.

— dessin —

3.5 DERIVATION D'UNE COURBE DE BEZIER :

Nous avons vu que $\frac{d}{dt} B_i^n(t) = n (B_{i-1}^{n-1}(t) - B_i^{n-1}(t))$, nous pouvons donc déterminer la dérivée d'une courbe de Bézier $\mathbf{x}(t)$:

$$\begin{aligned} \frac{d}{dt} \mathbf{x}(t) &= n \sum_{i=0}^n \mathbf{b}_i (B_{i-1}^{n-1}(t) - B_i^{n-1}(t)) \\ &= n \sum_{i=1}^n \mathbf{b}_i B_{i-1}^{n-1}(t) - n \sum_{i=0}^{n-1} \mathbf{b}_i B_i^{n-1}(t). \end{aligned}$$

Par transformation d'indice dans la première somme, on obtient

$$\frac{d}{dt} \mathbf{x}(t) = n \sum_{i=0}^n (\mathbf{b}_{i+1} - \mathbf{b}_i) B_i^{n-1}(t).$$

On peut simplifier la formule en introduisant l'opérateur de différence avant Δ :

$$\Delta \mathbf{b}_i = \mathbf{b}_{i+1} - \mathbf{b}_i,$$

d'où l'expression suivante pour la dérivée d'une courbe de Bézier :

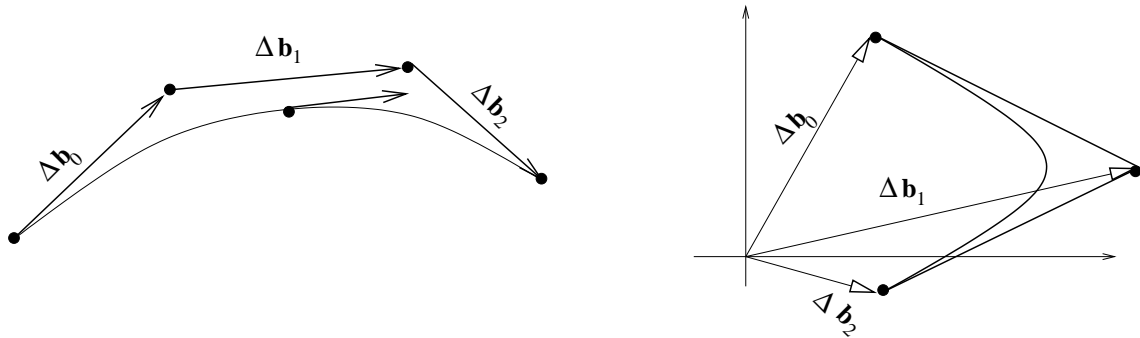
$$\frac{d}{dt} \mathbf{x}(t) = n \sum_{i=0}^{n-1} \Delta \mathbf{b}_i B_i^{n-1}(t), \quad \Delta \mathbf{b}_i \in \mathbb{R}^d.$$

La dérivée d'une courbe de Bézier est donc une autre courbe de Bézier, obtenue par application de l'opérateur différence sur le polygone de contrôle originale.

Cependant, cette courbe de Bézier dérivée n'est plus du tout dans \mathcal{E}^d !

Ces coefficients sont des différences de points, c.à.d. des vecteurs éléments de \mathbb{R}^d .

Afin de visualiser cette courbe dérivée et son polygone dans \mathcal{E}^d , nous pouvons construire un polygone dans \mathcal{E}^d qui consiste des points $\mathbf{a} + \Delta \mathbf{b}_0, \dots, \mathbf{a} + \Delta \mathbf{b}_{n-1}$. Ici \mathbf{a} est arbitraire, par exemple $\mathbf{a} = \mathbf{0}$.



Cette courbe dérivée est appelée une *hodographe*.

Remarque : distinguons entre espace affine \mathcal{E}^d et espace vectoriel \mathbb{R}^d . Les éléments de \mathcal{E}^d sont des points, tandis que les éléments de \mathbb{R}^d sont des vecteurs. La soustraction de deux points $\mathbf{a} - \mathbf{b} = \mathbf{v}$ donne un vecteur. Les points ne peuvent pas être additionnés, par contre les vecteurs le peuvent.

Pour les points les combinaisons barycentriques (=combinaisons affines) (sommés pondérées) existent : $\mathbf{b} = \sum \alpha_i \mathbf{b}_i$, $\mathbf{b}_i \in \mathcal{E}^d$ et $\sum \alpha_i = 1$.

Dérivées d'ordre supérieur

Notons $\Delta^r \mathbf{b}_i = \Delta^{r-1} \mathbf{b}_{i+1} - \Delta^{r-1} \mathbf{b}_i$ l'opérateur de différence avant itéré.

Par exemple

$$\begin{aligned} \Delta^0 \mathbf{b}_i &= \mathbf{b}_i \\ \Delta^1 \mathbf{b}_i &= \mathbf{b}_{i+1} - \mathbf{b}_i \\ \Delta^2 \mathbf{b}_i &= \mathbf{b}_{i+2} - 2\mathbf{b}_{i+1} + \mathbf{b}_i \\ \Delta^3 \mathbf{b}_i &= \mathbf{b}_{i+3} - 3\mathbf{b}_{i+2} + 3\mathbf{b}_{i+1} - \mathbf{b}_i \end{aligned}$$

Les coefficients des expressions à droite sont les coefficients du binôme, formant un triangle à celui de Pascal.

Généralement, on a

$$\Delta^r \mathbf{b}_i = \sum_{j=0}^r \binom{r}{j} (-1)^{r-j} \mathbf{b}_{i+j}.$$

La r -ième dérivée d'une courbe de Bézier :

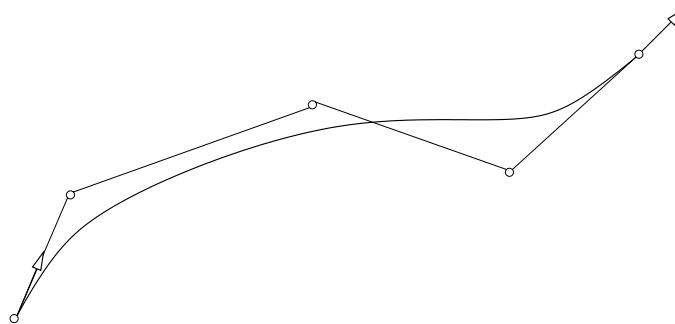
$$\frac{d^r}{dt^r} \mathbf{x}(t) = \frac{n!}{(n-r)!} \sum_{i=0}^{n-r} \Delta^r \mathbf{b}_i B_i^{n-r}(t).$$

Les deux cas particuliers suivants sont importants :

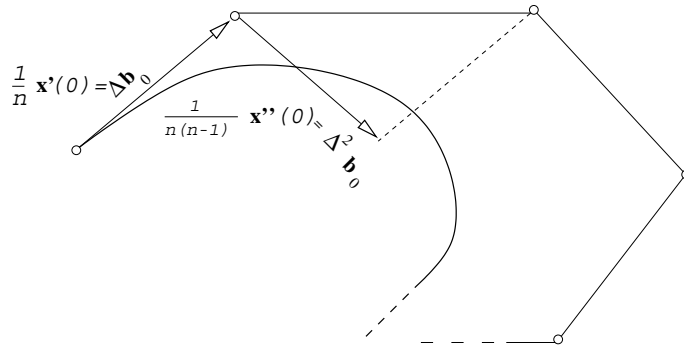
$$\begin{aligned} \frac{d^r}{dt^r} \mathbf{x}(0) &= \frac{n!}{(n-r)!} \Delta^r \mathbf{b}_0, \\ \frac{d^r}{dt^r} \mathbf{x}(1) &= \frac{n!}{(n-r)!} \Delta^r \mathbf{b}_{n-r}. \end{aligned}$$

Théorème de Bézier

La dérivée r -ième d'une courbe de Bézier en une de ses extrémités dépend uniquement des $r + 1$ points de contrôle les plus proche de cette extrémité.



$$\begin{aligned} \mathbf{x}(0) &= \mathbf{b}_0 & \mathbf{x}(1) &= \mathbf{b}_n \\ \mathbf{x}'(0) &= n \Delta \mathbf{b}_0 & \mathbf{x}'(1) &= n \Delta \mathbf{b}_{n-1} \\ \mathbf{x}''(0) &= n(n-1) \Delta^2 \mathbf{b}_0 & \mathbf{x}''(1) &= n(n-1) \Delta^2 \mathbf{b}_{n-2} \end{aligned}$$

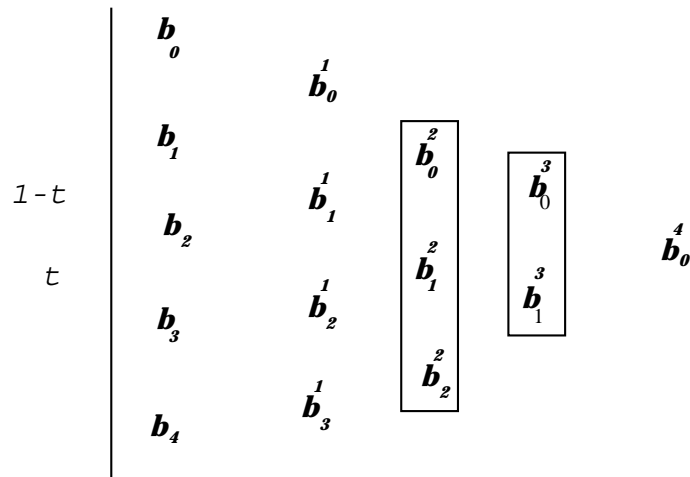


Dérivées et algorithme de de Casteljau

Les dérivées d'une courbe de Bézier peuvent être exprimées en fonction des points intermédiaires générés par l'algorithme de de Casteljau :

$$\frac{d^r}{dt^r} \mathbf{x}(t) = \frac{n!}{(n-r)!} \Delta^r \mathbf{b}_0^{n-r}(t).$$

Exemple : $n = 4$



$$\mathbf{x}'(t) = 4 \cdot \Delta \mathbf{b}_0^3 = 4(\mathbf{b}_1^3 - \mathbf{b}_0^3)$$

$$\mathbf{x}''(t) = 4 \cdot 3 \Delta^2 \mathbf{b}_0^2 = 12(\mathbf{b}_2^2 - 2\mathbf{b}_1^2 + \mathbf{b}_0^2)$$

— dessin —

Ceci est possible, car la sommation et l'opérateur différence commutent :

$$\sum_{j=0}^{n-r} \Delta \mathbf{b}_j = \sum_{j=0}^{n-r} \mathbf{b}_{j+1} - \sum_{j=0}^{n-r} \mathbf{b}_j = \Delta \sum_{j=0}^{n-r} \mathbf{b}_j$$

grâce à ceci, on a

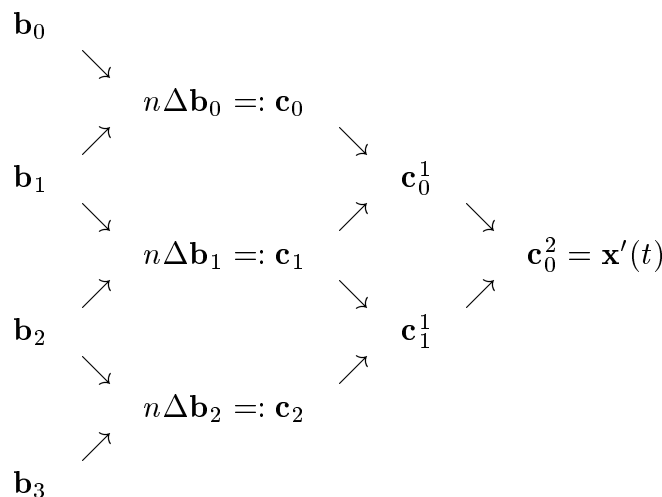
$$\begin{aligned} \frac{d^r}{dt^r} \mathbf{x}(t) &= \frac{n!}{(n-r)!} \sum_{i=0}^{n-r} \Delta^r \mathbf{b}_i B_i^{n-r}(t) \\ &= \frac{n!}{(n-r)!} \Delta^r \sum_{i=0}^{n-r} \mathbf{b}_i B_i^{n-r}(t) \\ &= \frac{n!}{(n-r)!} \Delta^r \mathbf{b}_0^{n-r}. \end{aligned}$$

Ces équations (la première et la dernière) suggèrent deux différentes méthodes de calcul de la r -ième dérivée d'une courbe de Bézier :

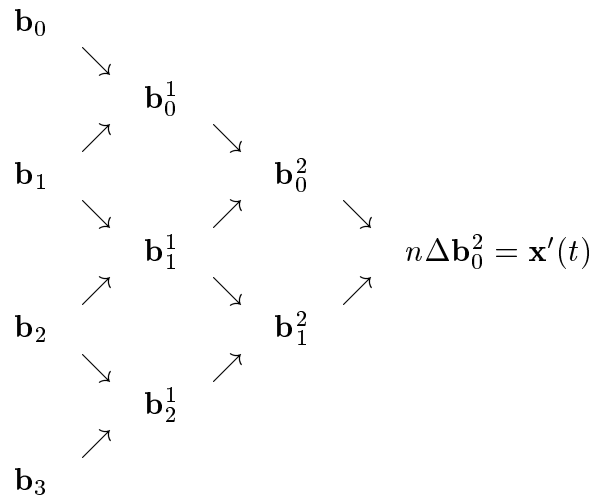
Méthode 1 :

Calculer toutes les différences $r^{\text{ème}}$ vers l'avant des points de contrôle, puis les interpréter comme un nouveau polygone de contrôle de degré $n-r$ et l'évaluer en t .

Exemple : $n = 3$ et $r = 1$



Méthode 2 :



L'algorithme de de Casteljau offre une solution rapide du problème d'obtenir un point sur la courbe et de connaître la dérivée première et/ou seconde en même temps.

Pour obtenir la dérivée r -ième, il faut effectuer r opérations de différence et $(n - r)$ opérations d'évaluation.

3.6 Formes polaires - une autre représentation des courbes de Bézier

Nous représentons ici une autre formalisme en utilisant la notion de "blossom" (*florraison*), ou de forme polaire associée à un polynôme, introduite par Lyle Ramshaw (1987).

1. Formes polaires

Une application $f : \mathbb{R} \rightarrow \mathbb{R}^d$ est dite affine, si

$$\left. \begin{array}{l} \forall a_1, \dots, a_m \in \mathbb{R} \text{ avec } \sum a_i = 1 \\ \forall u_1, \dots, u_m \in \mathbb{R} \end{array} \right\} : f\left(\sum_{i=1}^m a_i u_i\right) = \sum_{i=1}^m a_i f(u_i).$$

Exemple : $m = 2$, $a_1 = \alpha$, $a_2 = 1 - \alpha$, alors pour $t = (1 - \alpha)t_1 + \alpha t_2$:

$$f(t) = (1 - \alpha)f(t_1) + \alpha f(t_2).$$

Une application $f : \mathbb{R}^p \rightarrow \mathbb{R}^d$ est dite *multiaffine* (ou p -affine) si elle est affine séparément par rapport à chaque argument, c.à.d.

$$f(a_1, \dots, a_{j-1}, (1 - \alpha)u_1 + \alpha u_2, a_{j+1}, \dots, a_p) = (1 - \alpha)f(a_1, \dots, a_{j-1}, u_1, a_{j+1}, \dots, a_p) + \alpha f(a_1, \dots, a_{j-1}, u_2, a_{j+1}, \dots, a_p).$$

$f : \mathbb{R}^p \rightarrow \mathbb{R}^d$ est dite symétrique si f ne change pas sa valeur en échangeant deux arguments.

Exemple : $f(u_1, u_2) = f(u_2, u_1)$.

Exemple : $\mathbb{R}^p \rightarrow \mathbb{R} : S_i(u_1, \dots, u_p) = \sum_{1 \leq j_1 < \dots < j_i \leq p} u_{j_1} \dots u_{j_i}, i = 0, \dots, p.$
 (Ce sont les polynômes symétriques élémentaires).

$p = 3$

$$\begin{aligned} S_0(u_1, u_2, u_3) &= 1 \\ S_1(u_1, u_2, u_3) &= u_1 + u_2 + u_3 \\ S_2(u_1, u_2, u_3) &= u_1u_2 + u_1u_3 + u_2u_3 \\ S_3(u_1, u_2, u_3) &= u_1u_2u_3 \end{aligned}$$

chaque terme est linéaire ou constant en u_j donc S_i est affine par rapport à chaque u_j .
 \implies chaque combinaison linéaire $f = \sum_{i=0}^p \mathbf{c}_i S_i$ est une application symétrique, multiaffine.

L'inverse est aussi vraie :

Une application $f : \mathbb{R}^p \rightarrow \mathbb{R}^d$ est multiaffine et symétrique ssi f peut s'écrire comme CL :
 $f = \sum_{i=0}^p \mathbf{c}_i S_i(u_1, \dots, u_p), \quad \mathbf{c}_i \in \mathbb{R}^d$
 des polynômes symétriques élémentaires S_i .

On peut associer aux polynômes une application multiaffine symétrique de la manière suivante :

Théorème général du “Blossoming”

Si $\mathbf{F} \in \mathbb{R}^d : \mathbb{R} \rightarrow \mathbb{R}^d$ est un polynôme de degré p , alors il existe une fonction unique $f : \mathbb{R}^p \rightarrow \mathbb{R}^d$, appelée “blossom de \mathbf{F} ” (ou floraison de \mathbf{F} ou encore “forme polaire de \mathbf{F} ”) associée à \mathbf{F} qui soit :

- multiaffine,
- symétrique,
- et telle que $f(\underbrace{u, u, \dots, u}_p \text{ fois}) = \mathbf{F}(u)$.

Démonstration :

Soit $\mathbf{F}(u) = \sum_{i=0}^p \mathbf{a}_i u^i, \quad \mathbf{a}_i \in \mathbb{R}^d$ Une courbe polynomiale sous la forme canonique.

On observe que $S_i(\underbrace{u, \dots, u}_p \text{ fois}) = \binom{p}{i} u^i$, ce qui implique que $f(u_1, \dots, u_p) = \sum_{i=0}^p \mathbf{a}_i \binom{p}{i}^{-1} S_i(u_1, \dots, u_p)$

est une application multiaffine et symétrique telle que $f(u, \dots, u) = \mathbf{F}(u)$.

f unique : soit $g = \sum_{i=0}^p \mathbf{c}_i S_i$ multiaffine, symétrique et telle que $g(u, \dots, u) = \mathbf{F}(u)$, alors $\sum_{i=0}^p \left(\mathbf{a}_i \binom{p}{i}^{-1} - \mathbf{c}_i \right) u^i$ est le polynôme nul. D'où $\mathbf{a}_i \binom{p}{i}^{-1} = \mathbf{c}_i \implies f \equiv g$.

bullet Pour le degré n :

$$\begin{aligned} f(u_1, \dots, u_n) &= \frac{1}{n!} \sum_{\substack{S \subset \{1, \dots, n\} \\ S \neq \emptyset}} (-1)^{n-|S|} |S|^n \mathbf{F} \left(\frac{\sum_{i \in S} u_i}{|S|} \right) \\ &= \frac{1}{n!} \sum_{k=1}^n (-1)^{n-k} k^n \sum_{\substack{S \subset \{1, \dots, n\} \\ |S|=k}} \mathbf{F} \left(\frac{\sum_{i \in S} u_i}{k} \right) \end{aligned}$$

($|S|$ cardinal de S).

Par exemple, pour $n = 3$, on trouve

$$\begin{aligned} f(u_1, u_2, u_3) &= \frac{1}{3!} \left[(-1)^{3-1} 1^3 [\mathbf{F}(u_1) + \mathbf{F}(u_2) + \mathbf{F}(u_3)] + (-1)^{3-2} 2^3 [\mathbf{F}(\frac{u_1+u_2}{2}) + \right. \\ &\quad \left. + \mathbf{F}(\frac{u_1+u_3}{2}) + \mathbf{F}(\frac{u_2+u_3}{2})] + (-1)^{3-3} 3^3 [\mathbf{F}(\frac{u_1+u_2+u_3}{3})] \right]. \end{aligned}$$

Application aux courbes de Bézier

Une courbe de Bézier est une courbe polynomiale de degré n qui s'écrit dans la base de Bernstein

$$\mathbf{F}(t) = \sum_{i=0}^n \mathbf{b}_i B_i^n(t), \quad t \in [0, 1]$$

avec $B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}$, $t = \frac{b-u}{b-a}$ paramètre local.

Il existe une forme polaire de \mathbf{F} . Les points de Bézier sont obtenus comme suivant :

Les points de Bézier $\mathbf{b}_0, \dots, \mathbf{b}_n$ d'une courbe de Bézier $\mathbf{F} : \mathbb{R} \rightarrow \mathbb{R}^d$ par rapport à l'intervalle $[0, 1]$ se calculent avec

$$\mathbf{b}_i = f(\underbrace{0, \dots, 0}_{n-i}, \underbrace{1, \dots, 1}_i), \quad i = 0, \dots, n.$$

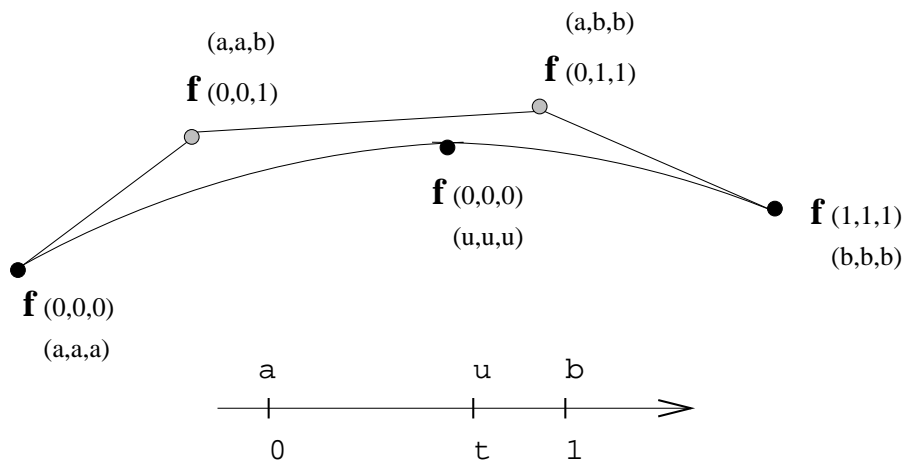
Démonstration :

Soit $t \in [0, 1] \implies t$ peut être représenté comme une combinaison affine de 0 et 1; $t = (1 - t) \cdot 0 + t \cdot 1$

f n -affine \implies

$$\begin{aligned} f(\underbrace{t, \dots, t}_n) &= (1 - t)f(\underbrace{0, t, \dots, t}_{n-1}) + tf(\underbrace{1, t, \dots, t}_{n-1}) \\ &= \sum_{i=0}^n \binom{n}{i} (1 - t)^{n-i} t^i f(\underbrace{0, \dots, 0}_{n-i}, \underbrace{1, \dots, 1}_i) \\ &= \sum_{i=0}^n \mathbf{b}_i B_i^n(t) \end{aligned}$$

Exemple $n = 3$



Algorithme de Decasteljau

On prendra la notation commode a^i pour désigner $\underbrace{a, \dots, a}_{i \text{ fois}}$, c.à.d. l'argument répété i -fois.

On posera pour $r = 0, \dots, n$: $\mathbf{b}_i^r = f(0^{n-i-r}, 1^i, t^r)$, $i = 0, \dots, n - r$.

Ainsi, on a l'algorithme de De Casteljau

$$\mathbf{b}_i^0 = \mathbf{b}_i, \quad \mathbf{b}_i^r = (1 - t)\mathbf{b}_i^{r-1} + t\mathbf{b}_{i+1}^{r-1}, \quad r = 0, \dots, n, \text{quad } i = 0, \dots, n - r,$$

et $\mathbf{b}_0^n = f(t, \dots, t) = \mathbf{F}(t)$

Partant de $\mathbf{b}_i^r = f(0^{n-i-r}, 1^i, t^r)$, on obtient la relation

$$f(0^{n-i-r}, 1^i, t^{r-1}, t) = (1 - t)f(0^{n-i-(r-1)}, 1^i, t^{r-1}) + tf(0^{n-i-r}, 1^i, t^{r-1}, 1)$$

(car $t = (1 - t)0 + t \cdot 1$).

C.à.d.

$$f(0^{n-i-r}, 1^i, t^r) = (1-t)f(0^{n-i-(r-1)}, 1^i, t^{r-1}) + tf(0^{n-i-r}, 1^{i+1}, t^{r-1})$$

ce qui correspond à $\mathbf{b}_i^r = (1-t)\mathbf{b}_i^{r-1} + t\mathbf{b}_{i+1}^{r-1}$

Base duale

L'application qui à un polynôme F de degré n fait correspondre sa floraison f est un *isomorphisme*.

Considérons les $n+1$ formes linéaires

$$\phi_i : F \in \mathcal{P}_n \longrightarrow f(0^{n-i}1^i) \in \mathbb{R}, \quad i = 0, \dots, n,$$

et l'application

$$\Phi : F \in \mathcal{P}_n \longrightarrow [\phi_0(F), \dots, \phi_n(F)] \in \mathbb{R}^{n+1}.$$

Il résulte directement de l'algorithme de De Casteljau qu'un polynôme F de degré n est parfaitement déterminé à partir de $\Phi(F)$; ($\Phi(F)$ sont les points de contrôle qui déterminent F), en d'autres termes l'application Φ est injective (donc bijective, car Φ linéaire).

Ainsi les $n+1$ formes linéaires ϕ_i sont linéairement indépendantes.

Il existe donc une base de \mathcal{P}_n duale des ϕ_i , on la notera B_j^n , $j = 0, \dots, n$.

B_j^n est le polynôme de degré n tel que

$$\phi_i(B_j^n) = \delta_{ij}, \quad \text{i.e. } \phi_i(B_j^n) = f(0^{n-i}1^i) = b_j(0^{n-i}1^i) = \delta_{ij}.$$

C.à.d., si $f = b_j$ désigne la floraison associée à B_j^n , telle que $f = b_j(0^{n-i}1^i) = \delta_{ij}$, c'est le polynôme que l'on obtient par l'algorithme de De Casteljau si l'on rend tous les b_j égaux à 0, sauf $b_i = 1$.

F polynôme de degré n , alors $f : \mathbb{R}^n \longrightarrow \mathbb{R}$

$$f(u, u, \dots, u) = F(u)$$

Φ est un isomorphisme et peut s'écrire :

$$\Phi(F) = \begin{pmatrix} \phi_0(F) \\ \vdots \\ \phi_i(F) \\ \vdots \\ \phi_n(F) \end{pmatrix} = \begin{pmatrix} 0^n 1^0 \\ \vdots \\ 0^{n-i} 1^i \\ \vdots \\ 0^0 1^n \end{pmatrix}$$

$$\phi_j(B_i^n) = \delta_{ij} = f_j(0^{n-i}1^i).$$

Calcul des dérivées

Théorème :

Si F est un polynôme de degré n et f sa floraison, alors

$$F'(t) = n[f(t, \dots, t, u+1) - f(t, \dots, t, u)], \forall u \in \mathbb{R}$$

$$= n \left[\frac{f(t, \dots, t, u+h) - f(t, \dots, t, u)}{h} \right], \forall u \in \mathbb{R}, \forall h \neq 0.$$

Démonstration :

Il suffit d'observer que l'on a :

$$F(t) = f(u_1(t), u_2(t), \dots, u_n(t)), \quad \text{si } u_j(t) \equiv t,$$

et d'appliquer la formule de dérivation d'une fonction composée :

$$F'(t) = \sum_{j=1}^n \frac{\partial f}{\partial u_j}(t, \dots, t) \cdot \frac{\partial u_j}{\partial t}(t),$$

on a bien sur $\frac{\partial u_j}{\partial t}(t) = 1$. Comme la fonction f est affine par rapport à chaque variable, alors

$$\begin{aligned} \frac{\partial f}{\partial u_1}(t, \dots, t) &= \frac{f(u+h, t, \dots, t) - f(u, t, \dots, t)}{h} \\ \frac{\partial f}{\partial u_2}(t, \dots, t) &= \frac{f(t, u+h, t, \dots, t) - f(t, u, t, \dots, t)}{h} \\ &= \frac{f(u+h, t, \dots, t) - f(u, t, \dots, t)}{h} = \frac{\partial f}{\partial u_1}(t, \dots, t) \end{aligned}$$

Ainsi, en utilisant la symétrie, toutes ces dérivées partielles $\frac{\partial f}{\partial u_j}(t, \dots, t)$ sont égales à $\frac{f(t, \dots, t, u+h) - f(t, \dots, t, u)}{h}$.

□

Floraison associée à F'

Si F est un polynôme de degré n , F' est un polynôme de degré $n-1$. Il lui correspond donc une floraison que l'on notera $f^{[1]}$ fonction de $n-1$ variables.

Il est facile à voir que l'on a

$$\begin{aligned}
f^{[1]} &= n[f(u_1, \dots, u_{n-1}, u+1) - f(u_1, \dots, u_{n-1}, u)], \forall u \in \mathbb{R} \\
&= n \left[\frac{f(u_1, \dots, u_{n-1}, u+h) - f(u_1, \dots, u_{n-1}, u)}{h} \right], \forall u \in \mathbb{R}, \forall h \neq 0.
\end{aligned}$$

En itérant, on peut trouver des expressions pour les dérivées d'ordre supérieur; par exemple pour F'' :

$$F''(t) = (n-1) \left[\frac{f^{[1]}(t^{n-2}, v+k) - f^{[1]}(t^{n-2}, v)}{k} \right], \quad \forall v, \quad \forall h \neq 0.$$

En reportant l'expression de $f^{[1]}$, on obtient alors :

$$F''(t) = \frac{n(n-1)}{h \cdot k} [f(t^{n-2}, v+k, u+h) - f(t^{n-2}, v+k, u) - f(t^{n-2}, v, u+h) + f(t^{n-2}, v, u)],$$

$\forall u, v, \forall h \neq 0, k \neq 0.$

On peut bien se donner à u, v, h, k des valeurs particulières, par exemple : $u = v = t$ et $k = h$:

$$F''(t) = \frac{n(n-1)}{h \cdot k} [f(t^{n-2}, t+h, t+h) - 2f(t^{n-1}, t+h) + f(t^n)]$$

ou encore, pour $h = 1$:

$$F''(t) = n(n-1) [f(t^{n-2}, (t+1)^2) - 2f(t^{n-1}, t+1) + f(t^n)]$$

Si les $\mathbf{b}_i, i = 0, \dots, n$ sont les points de Bézier de F par rapport à $[0, 1]$, on a :

$$F''(t) = n(n-1)(\mathbf{b}_2 - 2\mathbf{b}_1 + \mathbf{b}_0).$$

Mais comme on a aussi (pour $h = -1$) :

$$F''(t) = n(n-1) [f(t^{n-2}, (t-1)^2) - 2f(t^{n-1}, t-1) + f(t^n)]$$

on obtient

$$F''(1) = n(n-1)(\mathbf{b}_{n-2} - 2\mathbf{b}_{n-1} + \mathbf{b}_n).$$

La formule générale pour la dérivée $k^{\text{ième}}$ est la suivante :

$$F^{(k)}(t) = \frac{n(n-1)\dots(n-k+1)}{h^k} \sum_{j=0}^k \binom{k}{j} (-1)^{k-j} f(t^{n-j}(t+h)^j).$$

Par exemple un polynôme de degré 3 avec points de Bézier relatifs à $[0, 1]$ $\mathbf{b}_0, \dots, \mathbf{b}_3$, on obtient ($h = 1$) :

$$F'''(0) = 6(\mathbf{b}_3 - 2\mathbf{b}_1 + \mathbf{b}_0).$$

IV. Design Spline

4.1 LES FONCTIONS B-SPLINES

Nous allons dans ce chapitre généraliser la notion de fonctions B-splines définies sur une suite de noeuds qui permet des noeuds multiples. Cela va permettre d'obtenir des fonctions polynomiales de degré k par morceaux qui peuvent se raccorder aux noeuds avec un ordre de continuité différent d'un noeud à l'autre.

On se donne une subdivision $\Gamma = (\tau_1, \dots, \tau_{l-1})$ de l'intervalle $[a, b] \in \mathbb{R}$ t.q.:

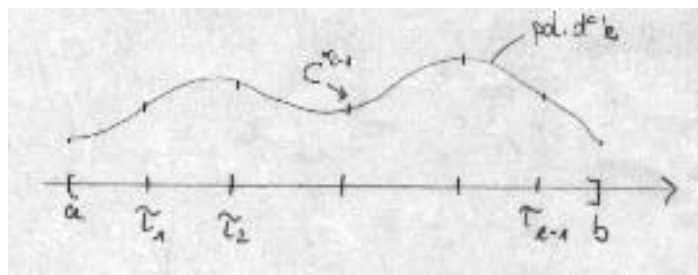
$$a < \tau_1 < \dots < \tau_{l-1} < b$$

et on posera $a = \tau_0, b = \tau_l$.

On se donne également $l - 1$ nombres entiers r_i vérifiant $0 \leq r_i \leq k$.

Notation $\Pi_{\tau,r}^k$ désignera l'espace vectoriel des fonctions sur $[a, b]$, polynomiales de degré k par morceaux avec raccord de classe C^{r_i-1} en τ_i ($1 \leq i \leq l - 1$).

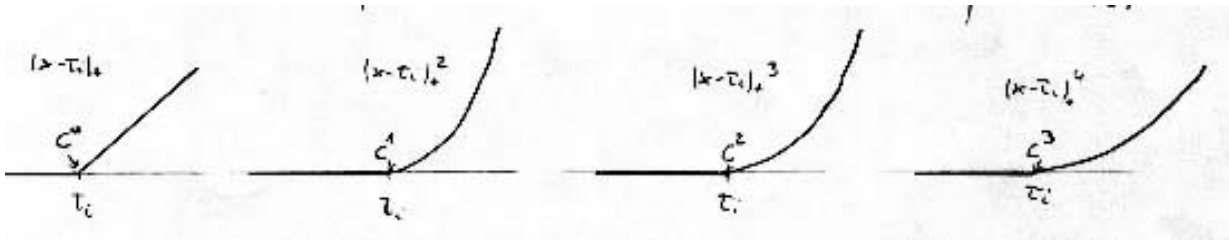
La dimension de $\Pi_{\tau,r}^k$ est égale à $\dim \Pi_{\tau,r}^k = (k + 1)l - \sum_{i=1}^{l-1} r_i$



On sait que les fonctions $(x - \tau_i)_+^j$, $i = 0, \dots, l - 1$, $j = r_i, \dots, k$ restreintes à $[a, b]$ forment une base de $\Pi_{\tau, r}^k$.

Explication:

$(x - \tau_i)_+$ est de classe C^0 . Par récurrence sur j , $(x - \tau_i)_+^j$ est de classe C^{j-1} en τ_i . Donc $(x - \tau_i)_+^j \in \Pi_{\tau, r}^k$ pour $r_i \leq j \leq k$. Comme il y a $\sum_{i=1}^l (k - r_i + 1) = (k + 1)l - \sum r_i = \dim \Pi_{\tau, r}^k$ fonctions de cette sorte, il suffit de vérifier que les $(x - \tau_i)_+^j$ sont linéairement indépendants.



On considère maintenant une suite de points réels

$$T = (t_0, \dots, t_m)$$

appelés **noeuds**, tels que $t_i \leq t_{i+1}$.

S'il y a p t_i égaux à τ , on dit que τ est un **noeud de multiplicité** p .

D'autre part, on pose:

$$\omega_{ij}(x) = \begin{cases} \frac{x-t_i}{t_{i+j}-t_i} & \text{si } t_i < t_{i+j} \\ 0 & \text{sinon} \end{cases} \quad \text{pour } 1 \leq j \leq m+1-i.$$

Définition Les fonctions définies par récurrence de la manière suivante pour une suite de noeuds T donnée,

$$N_i^0(x) = \begin{cases} 1 & \text{si } x \in [t_i, t_{i+1}[\\ 0 & \text{sinon} \end{cases}$$

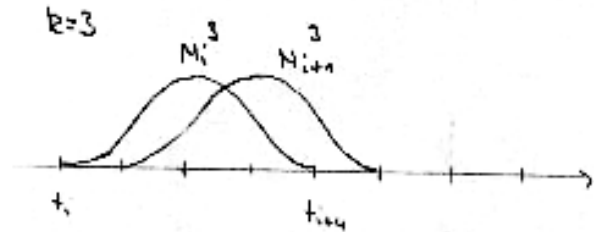
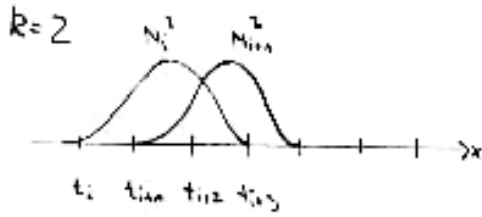
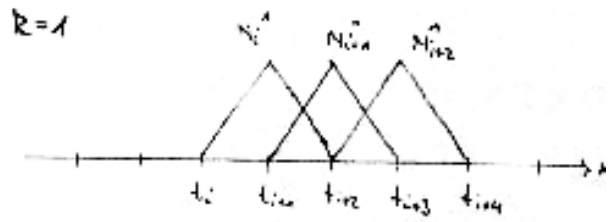
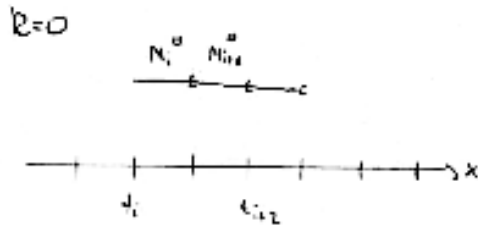
$$N_i^k(x) = \omega_{ik}(x)N_i^{k-1}(x) + (1 - \omega_{i+1,k}(x))N_{i+1}^{k-1}(x)$$

s'appellent **fonctions B-splines** de degré k . On les note $N_{i,t}^k(x)$ ou simplement $N_i^k(x)$.

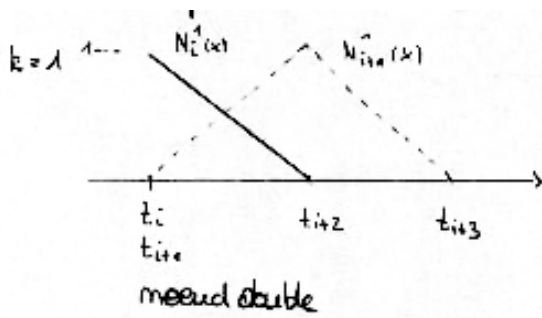
Remarque:

Si, pour un indice i , $t_i = t_{i+k+1}$ (i.e. t_i est un noeud de multiplicité $\geq k+2$), on a $N_i^k \equiv 0$.

Exemple:
Noeuds simples

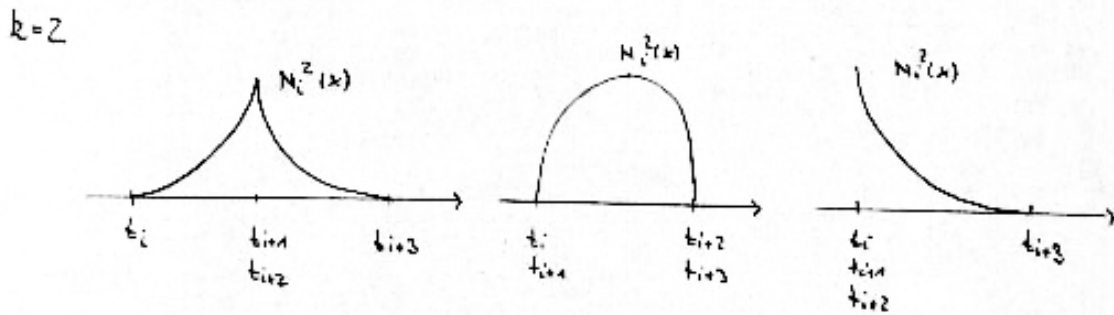


Noeuds multiples

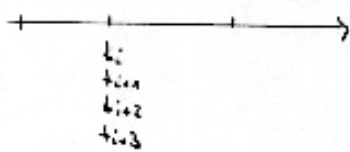


$$N_i^1(x) = \frac{x-t_i}{t_{i+1}-t_i} N_i^0(x) + \frac{t_{i+2}-x}{t_{i+2}-t_{i+1}} N_{i+1}^0(x)$$

$= 0$ d'après la définition de w_{i+1}



$N_i^2(x) \equiv 0$ noeud de multiplicité $\geq k+2$



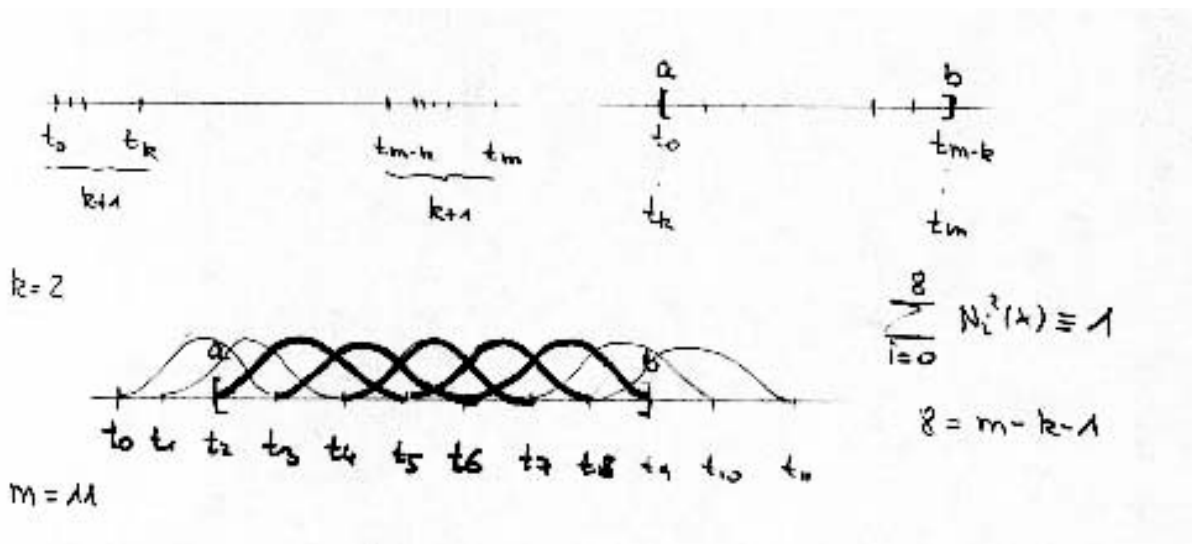
$$\left. \begin{array}{l} w_{i+2}(x) = 0 \quad t_i = t_{i+2} \\ w_{i+1,2}(x) = 0 \quad t_{i+1} = t_{i+3} \end{array} \right\} \Rightarrow N_i^2(x) \equiv 0$$

Proposition:

- a) $N_i^k(x)$ est polynomiale de degré k par morceaux
- b) $N_i^k(x) = 0$ pour $x \notin [t_i, t_{i+k+1}[$
- c) $N_i^k(x) > 0$ pour $x \in]t_i, t_{i+k+1}[$
 $N_i^k(t_i) = 0$ sauf si $t_i = \dots = t_{i+k} < t_{i+k+1}$ (noeud d'ordre $k+1$), auquel cas $N_i^k(t_i) = 1$.
- d) soit $[a, b]$ un intervalle tel que $t_k \leq a, t_{m-k} \geq b$, alors $\sum_{i=0}^{m-k-1} N_{i,t}^k(x) = 1 \quad \forall x \in [a, b]$.
- e) soit $x \in]t_i, t_{i+k+1}[$, alors $N(x) = 1$ ssi $t_{i+1} = \dots = t_{i+k} = x$.
- f) $N_i^k(x)$ est continue à droite, pour tout $x \in \mathbb{R}$.

Remarques:

- b) montre que chaque $N_i^k(x)$ a un petit support. On appelle $[t_i, t_{i+k+1}[$ le **support local** de $N_i^k(x)$.
- d) montre que les B-Splines forment une partition de l'unité.



Proposition Pour tout $k \geq 0$ et tout $x \in \mathbb{R}$, N_i^k est dérivable à droite et l'on a

$$\frac{d}{dx} N_i^k(x) = k \left[\frac{N_i^{k-1}(x)}{t_{i+k} - t_i} - \frac{N_{i+1}^{k-1}(x)}{t_{i+k-1} - t_{i+1}} \right]$$

avec la convention que l'on remplace par 0 une expression dont le dénominateur est nul.

Démonstration:

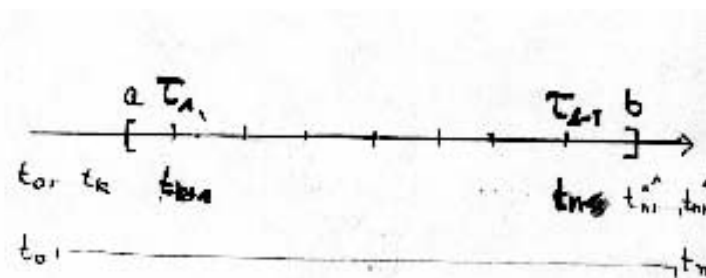
— exercice —

On peut maintenant définir la suite de noeuds t_i de façon à ce que les fonctions B-Splines que l'on en déduit forment une base de $\Pi_{\tau,r}^k$.

Posons $n + 1 = l(k + 1) - \sum_{i=1}^{l-1} r_i$, $n + 1$ est la dimension de l'espace $\Pi_{\tau,r}^k$. Considérons une suite de points t_i , $0 \leq i \leq m = n + k + 1$ vérifiant les conditions suivantes:

- 1) $t_i \leq t_{i+1}$
- 2) $t_j \leq a$ ($0 \leq j \leq k$)
- 3) $t_j \geq b$ ($n + 1 \leq j \leq n + k + 1$)
- 4) $(k + 1 - r_i =: p_i)$ t_j sont égaux à τ_i , pour tout i ($1 \leq i \leq l - 1$).

De tels t_i sont appelés noeuds, et ils sont déterminés de manière unique par les suites τ et r , sauf (t_0, \dots, t_k) et $(t_{n+1}, \dots, t_{n+k+1})$ seulement assujettis aux conditions 2) et 3).

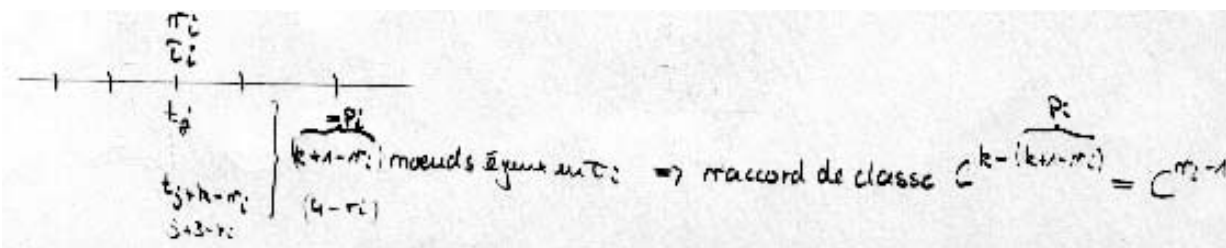
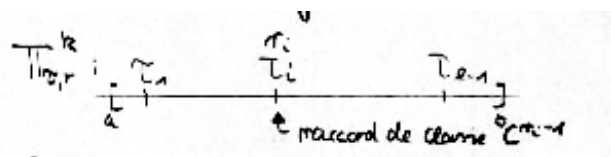


Définition

Soit t_i ($0 \leq i \leq m$) une suite de points de \mathbb{R} telle que $t_i \leq t_{i+1}$, k est un entier positif ou nul, $[a, b]$ un intervalle tel que $t_k \leq a$, $t_{m-k} \geq b$.

On note Π_t^k l'espace vectoriel des fonctions polynomiales par morceaux sur $[a, b]$ de degré $\leq k$, avec raccords de classe C^{k-p_j} en t_j , si t_j est un noeud de multiplicité p_j .

Par convention, un raccord de classe C^{k-p_j} avec $k - p_j < 0$ n'impose aucune condition en t_j .



Avec la correspondance entre les (τ_i, r_i) et les t_i définie ci-dessus, on a donc:

$$\Pi_{\tau,r}^k = \Pi_t^k.$$

Si tous les noeuds sont de multiplicité $\leq k + 1$, on pose $(n + 1) = m - k$ et l'on a:

$$\dim \Pi_t^k = (n + 1)$$

Théorème *Supposons que tous les noeuds soient de multiplicité $\leq k + 1$, alors les $N_{i,t}^k$, ($i = 0, \dots, n$) forment une base de Π_t^k .*

Démonstration:

— voir J.J. Risler [??] —

Remarques:

- La suite des τ_i désigne les emplacements des t_i qui sont différents.
- Un noeud t_j de multiplicité p_j implique que les éléments de l'espace Π_t^k ont un raccord de classe C^{k-p_j} en t_j .

Résumé:

Ce qui est important pour la suite, c'est

k le degré de la courbe spline $\in \Pi_t^k$,

$m + 1$ le nombre de noeuds t_i ,

$T = (t_0, \dots, t_m)$ la suite des noeuds,

ainsi que la dimension $(n + 1) = m - k$ de Π_t^k est fixé et on peut ainsi définir un **fonction spline**

$$S(x) = \sum_{i=0}^n a_i N_i^k(x)$$

comme élément de Π_t^k .

4.2 COURBES B-SPLINE

Les courbes splines paramétriques définies dans la base des fonctions B-spline sont déterminées par une suite de noeuds T ainsi que par des points de contrôle.

Définition

Soit $k, n \in \mathbf{N}$, $n \geq k$, $T = (t_0, \dots, t_{n+k+1})$ avec $t_i \leq t_{i+1}$, $N_{i,t}^k$ la fonction B-spline de degré k , on appelle

$$S(x) = \sum_{i=0}^n d_i N_i^k(x), \quad x \in [t_k, t_{n+1}[$$

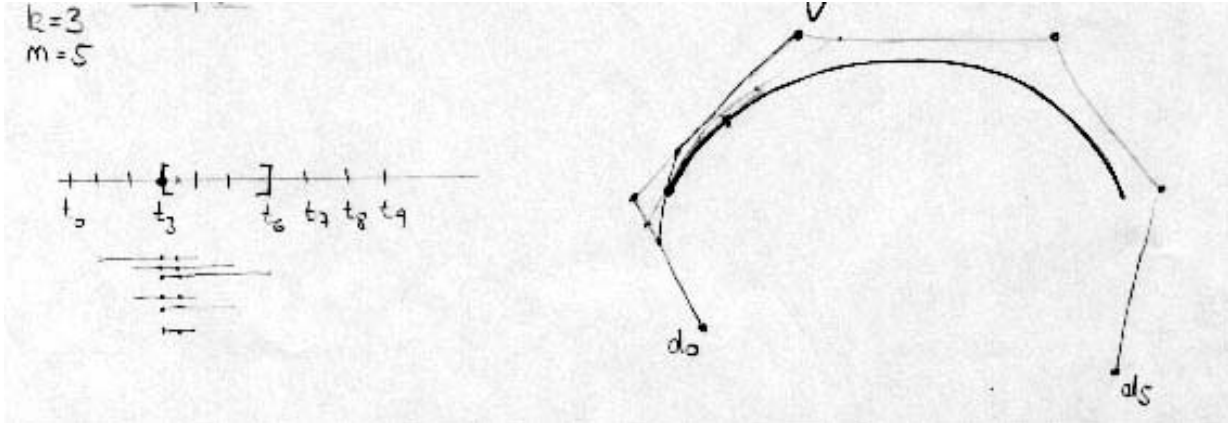
une **courbe B-spline**. Les coefficients $d_i \in \mathbb{R}^d$ sont appelés **points de de Boor** ou **points de contrôle**, et forment le polygone de contrôle.

Notons les relations suivantes:

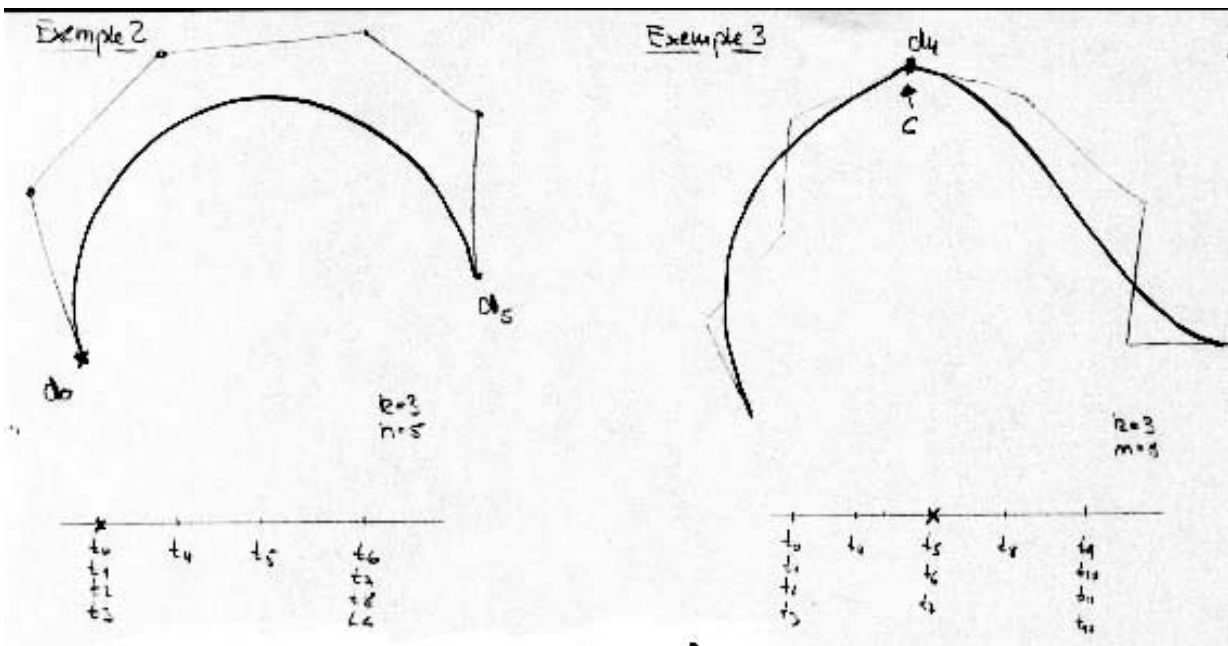
$$m = n + k + 1$$

$$n = m - k - 1.$$

Exemple 1:



Exemple 2 et 3:



4.3 EVALUATION D'UNE COURBE B-SPLINE

La formule de récurrence des fonctions B-spline permet de déduire directement la proposition suivante:

soit $x \geq t_k$, $S(x) = \sum_{i=0}^k d_i N_i^k(x)$, on a:

$$S(x) = \sum d_i^0 N_i^k(x) = \sum d_i^1 N_i^{k-1}(x) = \dots = \sum d_i^k N_i^0(x) \quad (*_{Boor})$$

avec

$$\begin{aligned} \mathbf{d}_i^0 &= \mathbf{d}_i \text{ et} \\ \mathbf{d}_i^{r+1} &= \omega_{i,k-r}(x)\mathbf{d}_i^r + (1 - \omega_{i,k-r}(x))\mathbf{d}_{i-1}^r. \end{aligned}$$

Exemple: Passage de \mathbf{d}_i^0 à \mathbf{d}_i^1 :

$$\begin{aligned} \sum \mathbf{d}_i^0 N_i^k &= \sum \mathbf{d}_i (\omega_{ik} N_i^{k-1} + (1 - \omega_{i+1,k}) N_{i+1}^{k-1}) \\ &= \sum \mathbf{d}_i \omega_{ik} N_i^{k-1} + \sum \mathbf{d}_i (1 - \omega_{i+1,k}) N_{i+1}^{k-1} \\ &= \sum \mathbf{d}_i \omega_{ik} N_i^{k-1} + \sum \mathbf{d}_{i-1} (1 - \omega_{ik}) N_i^{k-1} \\ &= \sum (\omega_{ik} \mathbf{d}_i + (1 - \omega_{ik}) \mathbf{d}_{i-1}) N_i^{k-1} \\ &= \sum \mathbf{d}_i^1 N_i^{k-1}. \end{aligned}$$

Si l'on veut évaluer $S(x)$ par $x \in [t_j, t_{j+1}[$, on a donc $S(x) = \mathbf{d}_j^k(x)$ puisque

$$N_j^0(x) = \begin{cases} 1 & \text{si } t \in [t_j, t_{j+1}[\\ 0 & \text{sinon.} \end{cases}$$

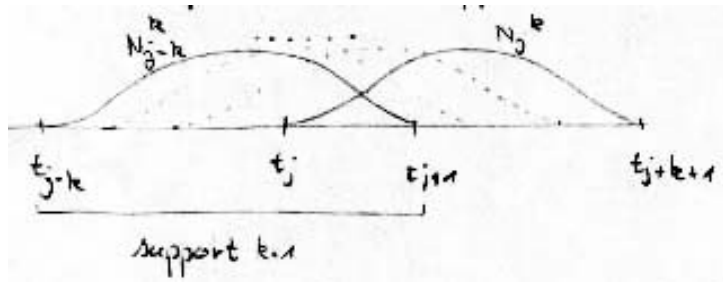
Quels indices aux sommes dans $(*_{Boor})$?

Pour calculer $\mathbf{d}_j^k(x)$, il suffit d'évaluer \mathbf{d}_i^r seulement pour $j - k + r \leq i \leq j$, les autres $N_i^{k-r}(x)$ étant nuls puisque $x \in [t_j, t_{j+1}[$.

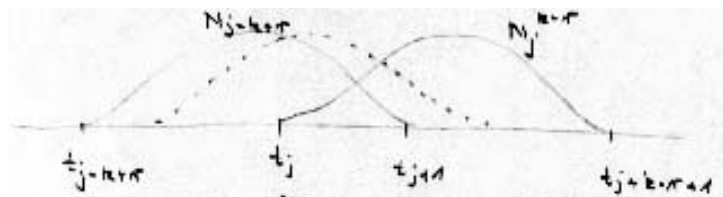
On a ainsi

$$S(x) = \sum_{i=j-k}^j \mathbf{d}_i^0(x) N_i^k(x) = \sum_{i=j-k+1}^j \mathbf{d}_i^1(x) N_i^k(x) - 1(x) = \dots = \mathbf{d}_j^k(x), \quad j = k, \dots, n.$$

* Pour $j - k \leq i \leq j$, le support des N_i^k intersecte l'intervalle $[t_j, t_{j+1}[$.



* Pour $j - k + r \leq i \leq j$, le support des N_i^{k-r} intersecte $[t_j, t_{j+1}[$.



$$S(x) = \sum_{i=j-k}^j \mathbf{d}_i^0 N_i^k(x) = \dots = \sum_{i=j-k+r}^j \mathbf{d}_i^r N_i^{k-r}(x) = \dots = \mathbf{d}_j^k$$

pour $x \in [t_j, t_{j+1}[$ et $j = k, \dots, n$.

Nous avons bien dit que $S(x)$ est défini sur $[t_k, t_{n+1}[$. Pour $x = t_{n+1}$ nous ne pourrions pas appliquer l'algorithme de Boor, car $x \in [t_{n+1}, t_{n+2}[\Rightarrow j = n+1 \Rightarrow \{\mathbf{d}_{n+1-k}, \dots, \mathbf{d}_{n+1}\}$ fait intervenir le point de contrôle \mathbf{d}_{n+1} non existant.

Algorithme d'évaluation de de Boor

$x \in [t_j, t_{j+1}[$ j étant l'indice maximal en cas de noeuds multiples $t_{j-P_j} = \dots = t_j$ et si $x = t_j$.

$$\mathbf{d}_i^0 := \mathbf{d}_i, \quad (j-k \leq i \leq j)$$

Pour $r = 0, \dots, k-1$

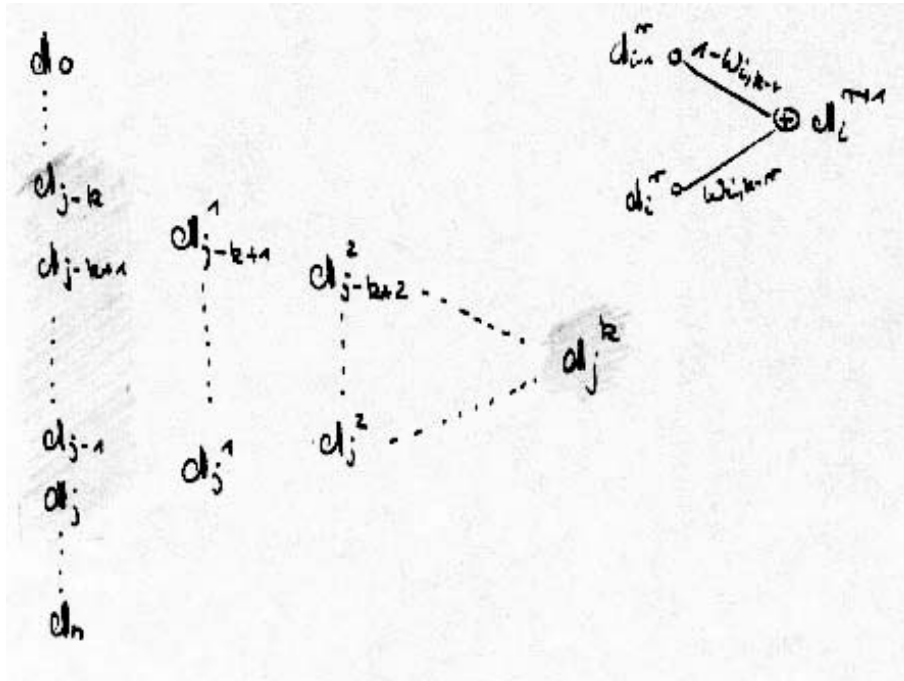
$$i = j-k+r, \dots, j$$

$$\begin{aligned} \mathbf{d}_i^{r+1}(x) &= \omega_{i,k-r}(x) \mathbf{d}_i^r + (1 - \omega_{i,k-r}(x)) \mathbf{d}_{i-1}^r \\ &= \frac{x - t_i}{t_{i+k-r} - t_i} \mathbf{d}_i^r + \frac{t_{i+k-r} - x}{t_{i+k-r} - t_i} \mathbf{d}_{i-1}^r \end{aligned}$$

$$\mathbf{d}_i^k(x) = S(x).$$

Cet algorithme se présente sous forme triangulaire et porte le nom "algorithme de de Boor".

Soit $x \in [t_j, t_{j+1}[$, alors $S(x) = \mathbf{d}_j^k(x)$, ($j \in \{k, \dots, n\}$).



Remarques:

- Quand $\omega_{i,j} = \frac{0}{0}$ arrive, on prend par convention $\omega_{i,j} = 0$
- cet algorithme nécessite le calcul de $(k(k+1))/2$ combinaisons convexes.
- il est numériquement stable.
- le calcul des $\omega_{i,k-r}(x)$ est souvent très simple en pratique (par exemple si les noeuds t_i sont des entiers).
- il est très bien adapté au tracé graphique des courbes splines.

Contrôle locale sur la courbe:

- * support d'une fonction de base $N_j^k(x) : [t_j, t_{j+k+1}[$
- * influence du point de contrôle $\mathbf{d}_j : [t_j, t_{j+k+1}[$
- * influence du paramètre $x \in [t_j, t_{j+1}[: \mathbf{d}_{j-k}, \dots, \mathbf{d}_j$

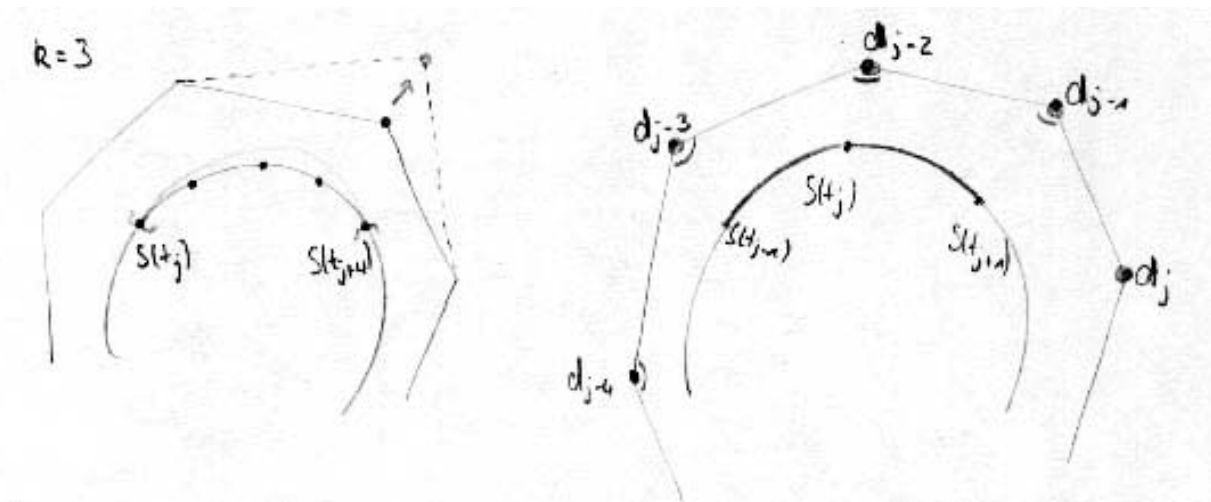
4.4 PROPRIÉTÉS

(i) Contrôle local

- Le support local des fonctions de base N_j^k implique que chaque point de contrôle n'a qu'une influence locale sur la courbe. Si le degré est k , alors \mathbf{d}_j n'intervient que dans le calcul de $k+1$ segments de courbe $S(t)$ pour $t \in [t_j, t_{j+k+1}[$.
- Le point $S(x_0)$ de la courbe ne dépend que de $k+1$ points de contrôle, puisque pour $t_j \leq x_0 < t_{j+1}$

$$S(x_0) = \sum_{i=0}^n \mathbf{d}_i N_i^k(x_0) = \sum_{i=j-k}^j \mathbf{d}_i N_i^k(x_0).$$

Exemple:



(ii) Diminution de la variation (des fonctions B-spline)

Définition Soit $\mathbf{a} = (a_0, \dots, a_n)$ une suite de nombres réell. On appelle variation de \mathbf{a} , noté $V(\mathbf{a})$, le nombre de changements de signe de la suite \mathbf{a} (i.e. le nombre d'indices i tels que $a_i \cdot a_{i+r} < 0$, avec $a_i = \dots = a_{i+r-1} = 0$).

Exemple: $\mathbf{a} = (-1, 0, 0, 2, 0, 4, 0, 0, -5, 6) : V(\mathbf{a}) = 3$.

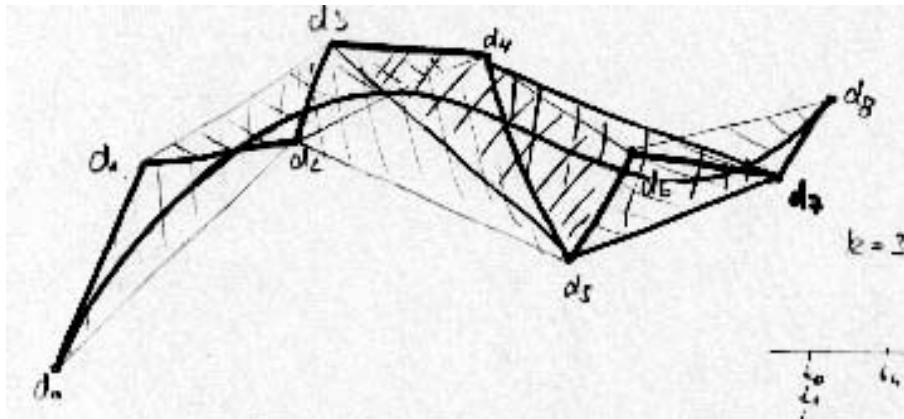
Définition Pour une fonction $f : [a, b] \rightarrow \mathbb{R}$, on note $V(f)$ l avariation de f , ou nombre de changements de signe de f , le nombre $\text{Sup}V(f(x_1), \dots, f(x_r))$ pour toutes les suites de points $x = (x_1, \dots, x_r)$ de $[a, b]$ ($x_1 < \dots < x_r$).

Proposition Soit $s(x) = \sum_{i=0}^n d_i N_i^k(x)$ une fonction B-spline (définie sur $[a, b]$ t.q. $t_k \leq a$ et $t_{n+1} \geq b$), alors $V(s) \leq V(\mathbf{d})$.

Démonstration: cf. J.-j. Risler

(iii) Enveloppe convexe

Chaque point de la courbe est dans l'enveloppe convexe de $k + 1$ points de contrôle: $s(\tau)$ avec $\tau \in [t_j, t_{j+1}[$ est dans l'EC de $\{\mathbf{d}_{j-k}, \dots, \mathbf{d}_j\}$.



Démonstration: résulte du fait que $\sum_{i=0}^n N_i^k(x) = 1$ et du fait que si $t \in [t_j, t_{j+1}[$, on a $N_i^k(x) = 1$ pour $i \leq j - k - 1$ et $i \geq j + 1$.

— Dessin —

(iv) Nœuds multiples au bord

En design paramétrique on préfère travailler avec des nœuds de multiplicité $k + 1$ aux deux extrémités de l'intervalle, i.e. $t + 0 = \dots = t_k$ et $t_{n+1} = \dots = t_{n+k+1}$.

La courbe passe ainsi par \mathbf{d}_0 et \mathbf{d}_n et est tangente au premier et dernier segment du polygone de contrôle. Cela permet un meilleur contrôle de la courbe.

Dans cette configuration de nœuds, on met par convention $\mathbf{s}(t_{n+1}) = \mathbf{d}_n$ (la courbe n'étant pas définie pour $x = t_{n+1}$).

(v) Nœuds intérieurs simples

Dans le cas où les nœuds t_i ($i = k + 1, \dots, n$) sont simples, la courbe $\mathbf{s}(x)$ est continue de classe C^{k-1} , et elle est formée de $n - k + 1$ arcs paramétrés polynomiaux de degré $\leq k$.

(vi) Invariance affine

$\mathbf{s}(x)$ est invariante sous transformation affine. Voir chapitre sur les courbes de Bézier pour plus de détails.

4.4 INSERTION D'UN NŒUD

Soit $\mathbf{s}(x) = \sum_{i=0}^n \mathbf{d}_i N_{i,t}^k(x)$ une courbe B-spline de degré k basée sur une suite de nœuds $T = (t_0, \dots, t_{n+k+1})$.

Rajoutons un nœud \hat{t} tel que $\hat{t} \in [t_k, t_{n+1}[$ (\hat{t} peut être confondu avec un des t_i).

A la nouvelle suite de nœuds, noté \hat{T} , ainsi construite, correspond une famille de B-splines \hat{N}_i^k ($= N_{i,\hat{t}}^k$), $0 \leq i \leq n + 1$ et l'on a $\Pi_{\hat{t}}^k \subset \Pi_t^k$.

Théorème On a

$$\sum_{i=0}^n \mathbf{d}_i N_{i,t}^k(x) = \sum_{i=0}^{n+1} \hat{\mathbf{d}}_i N_{i,\hat{t}}^k(x)$$

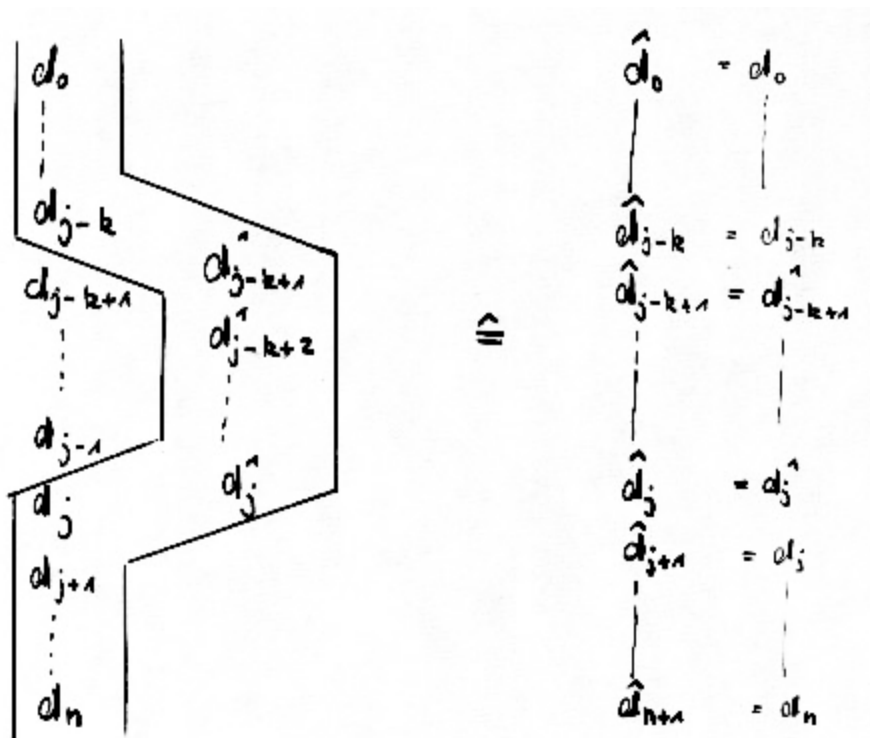
avec

$$\hat{\mathbf{d}}_l = \begin{cases} \mathbf{d}_l & l = 0, \dots, j - k \\ \omega_{l,k}(\hat{t})\mathbf{d}_l + (1 - \omega_{l,k}(\hat{t}))\mathbf{d}_{l-1} & l = j - k + 1, \dots, j \\ \mathbf{d}_{l-1} & l = j + 1, \dots, n + 1 \end{cases}$$

quand $\hat{t} \in [t_j, t_{j+1}[$.

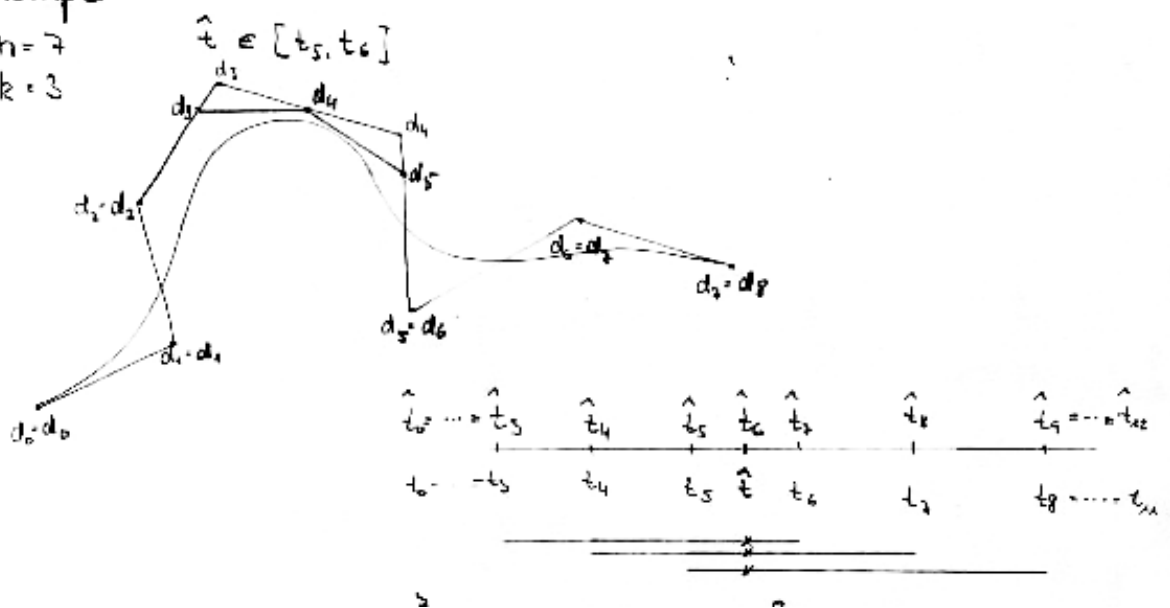
Démonstration: par récurrence.

Cet algorithme correspond à la première étape de l'algorithme de de Boor (évaluation) pour $x = \hat{t} \in [t_j, t_{j+1}[$:



Exemple

$n=7$
 $k=3$



Les deux courbes de l'exemple $s(x) = \sum_{i=0}^7 d_i N_{i,t}^3(x)$ et $\hat{s}(x) = \sum_{i=0}^8 \hat{d}_i N_{i,\hat{t}}^3(x)$ sont identiques !

Conclusion:

L'insertion de nœuds permet d'introduire plus de degrés de liberté pour la courbe. Chaque nœud inséré représente un point de contrôle supplémentaire. L'insertion de nœuds offre donc plus de flexibilité pour le design avec des courbes B-spline.

V. Approximation au sens des moindres carrés

5.1 PROJECTION D'UN POINT SUR UN S.E.V.

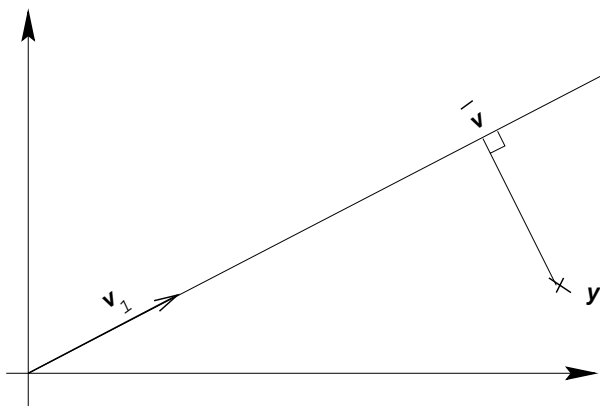
Etudier le problème de la *projection d'un point sur un sous-espace vectoriel* (ou sur une variété affine) de l'espace euclidien \mathbb{R}^N .

Application : Problème de l'approximation au sens des moindres carrés d'une fonction (ou de données) par un polynôme (ou par un élément d'un sous-espace des fonctions de dimension finie).

Un exemple introductif : (très élémentaire)

On note $\mathbf{V} = \{\mathbf{v} \in \mathbb{R}^2 \mid \mathbf{v} = \lambda_1 \mathbf{v}_1, \lambda_1 \in \mathbb{R}\} (\subset \mathbb{R}^2)$ pour le sous-espace de \mathbb{R}^2 formé des vecteurs proportionnels à un vecteur $\mathbf{v}_1 = [v_1^1, v_1^2] \neq \mathbf{0}$ donné.

Etant donné $\mathbf{y} \in \mathbb{R}^2$ ($\mathbf{y} \notin \mathbf{V}$), on cherche le vecteur de \mathbf{V} qui se trouve à une courte distance de \mathbf{y} , au sens de la norme euclidienne.



Si l'on pose $\|\mathbf{x}\| = \sqrt{(x^1)^2 + (x^2)^2}$, pour $\mathbf{x} = [x^1, x^2] \in \mathbb{R}\}$,
on cherche donc $\bar{\mathbf{v}} = \bar{\lambda}_1 \mathbf{v}_1$ tel que

$$\|\bar{\mathbf{v}} - \mathbf{y}\| = \min_{\mathbf{v} \in \mathbf{V}} \|\mathbf{v} - \mathbf{y}\|.$$

Si l'on pose $\bar{\mathbf{v}} = \bar{\lambda}_1 \mathbf{v}_1$, cela revient à chercher $\bar{\lambda}_1$ tel que

$$(\bar{\lambda}_1 v_1^1 - y^1)^2 + (\bar{\lambda}_1 v_1^2 - y^2)^2 = \min_{\lambda_1} (\lambda_1 v_1^1 - y^1)^2 + (\lambda_1 v_1^2 - y^2)^2.$$

L'élément $\bar{\mathbf{v}} \in \mathbf{V}$ solution de ce problème est caractérisé par le fait que le vecteur erreur $\bar{\mathbf{v}} - \mathbf{y}$ est orthogonal à \mathbf{V} , c.a.d. à \mathbf{v}_1 :

$$\langle \bar{\mathbf{v}} - \mathbf{y}, \mathbf{v}_1 \rangle = 0,$$

où $\langle \mathbf{x}_1, \mathbf{x}_2 \rangle = x_1^1 x_2^1 + x_1^2 x_2^2$, d'où en remplaçant $\bar{\mathbf{v}}$ par $\bar{\lambda}_1 \mathbf{v}_1$:

$$\bar{\lambda}_1 \langle \mathbf{v}_1, \mathbf{v}_1 \rangle = \langle \mathbf{y}, \mathbf{v}_1 \rangle,$$

ce qui donne

$$\bar{\lambda}_1 = \frac{\langle \mathbf{y}, \mathbf{v}_1 \rangle}{\langle \mathbf{v}_1, \mathbf{v}_1 \rangle} = \frac{y^1 v_1^1 + y^2 v_1^2}{(v_1^1)^2 + (v_1^2)^2}.$$

Généralement, soit $\mathbf{E} = \mathbb{R}^N$ (N entier donné), et on considère le sous-espace vectoriel

$$\mathbf{V} = \left\{ \mathbf{v} \in \mathbb{R}^N \mid \mathbf{v} = \sum_{j=1}^n \lambda_j \mathbf{v}_j \right\},$$

où $\mathbf{v}_j = [v_j^1, \dots, v_j^N] \in \mathbb{R}^N$.

Etant donné $\mathbf{y} = [y^1, \dots, y^N] \in \mathbb{R}^N$, on se propose de minimiser $\|\mathbf{v} - \mathbf{y}\|$,
pour $\mathbf{v} \in \mathbf{V}$,

$$\text{où } \|\mathbf{x}\| = \sqrt{\sum_{i=1}^N (x^i)^2},$$

C.a.d. de trouver l'élément $\bar{\mathbf{v}} \in \mathbf{V}$ tel que

$$\|\bar{\mathbf{v}} - \mathbf{y}\| = \min_{\mathbf{v} \in \mathbf{V}} \|\mathbf{v} - \mathbf{y}\|.$$

Explicitement :

$$\sum_{i=1}^N \left(\sum_{j=1}^n \bar{\lambda}_j v_j^i - y^i \right)^2 = \min_{\lambda_1, \dots, \lambda_n} \sum_{i=1}^N \left(\sum_{j=1}^n \lambda_j v_j^i - y^i \right)^2$$

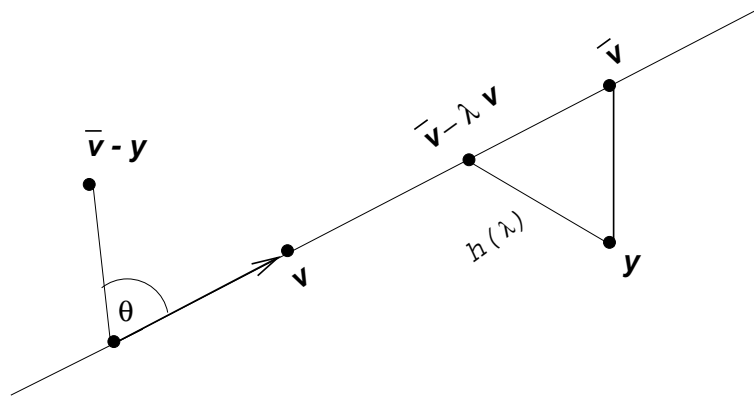
Théorème de caractérisation :

Une condition nécessaire et suffisante pour que $\bar{\mathbf{v}} \in \mathbf{V}$ soit solution et que l'on ait : $\langle \bar{\mathbf{v}} - \mathbf{y}, \mathbf{v} \rangle = 0$, pour tout $\mathbf{v} \in \mathbf{V}$.

Démonstration

(a) Condition nécessaire:

Soit $\bar{\mathbf{v}} \in \mathbf{V}$ solution du problème. Supposons que l'on ait $\langle \bar{\mathbf{v}} - \mathbf{y}, \mathbf{v} \rangle > 0$ pour un élément $\mathbf{v} \in \mathbf{V}$ et montrons que cela conduit à une contradiction :



L'angle θ entre $\bar{\mathbf{v}} - \mathbf{y}$ et \mathbf{v} est $< \frac{\pi}{2}$.

Considérons l'élément $\bar{\mathbf{v}} - \lambda \mathbf{v}$ ($\lambda > 0$) et posons

$$\begin{aligned} H(\lambda) &= \|\bar{\mathbf{v}} - \lambda \mathbf{v} - \mathbf{y}\|^2 \\ &= \|\bar{\mathbf{v}} - \mathbf{y}\|^2 + \lambda^2 \|\mathbf{v}\|^2 - 2\lambda \langle \bar{\mathbf{v}} - \mathbf{y}, \mathbf{v} \rangle. \end{aligned}$$

$$H(0) = \|\bar{\mathbf{v}} - \mathbf{y}\|^2 > 0,$$

$$H'(0) = -2 \langle \bar{\mathbf{v}} - \mathbf{y}, \mathbf{v} \rangle < 0.$$

Il existe donc $\lambda > 0$ tel que $H(\lambda) < H(0)$. Mais ceci est en contradiction avec $\bar{\mathbf{v}}$ est une solution du problème.

(b) Condition suffisante:

Soit $\bar{\mathbf{v}} \in \mathbf{V}$ tel que $\langle \bar{\mathbf{v}} - \mathbf{y}, \mathbf{v} \rangle = 0$ pour tout $\mathbf{v} \in \mathbf{V}$. On a

$$\begin{aligned} \|\bar{\mathbf{v}} - \mathbf{y}\|^2 &= \|(\mathbf{v} - \bar{\mathbf{v}}) + (\bar{\mathbf{v}} - \mathbf{y})\|^2 \\ &= \|\mathbf{v} - \bar{\mathbf{v}}\|^2 + \|\bar{\mathbf{v}} - \mathbf{y}\|^2 + 2 \langle \mathbf{v} - \bar{\mathbf{v}}, \bar{\mathbf{v}} - \mathbf{y} \rangle \end{aligned}$$

mais $\langle \mathbf{v} - \bar{\mathbf{v}}, \bar{\mathbf{v}} - \mathbf{y} \rangle = \langle \bar{\mathbf{v}} - \mathbf{y}, \mathbf{v} - \bar{\mathbf{v}} \rangle = 0$ (car $\mathbf{v} - \bar{\mathbf{v}} \in \mathbf{V}$). D'où

$$\|\bar{\mathbf{v}} - \mathbf{y}\|^2 = \|\mathbf{v} - \bar{\mathbf{v}}\|^2 + \|\bar{\mathbf{v}} - \mathbf{y}\|^2,$$

on a donc bien $\|\bar{\mathbf{v}} - \mathbf{y}\|^2 \geq \|\bar{\mathbf{v}} - \mathbf{y}\|^2$, pour tout $\mathbf{v} \in \mathbf{V}$. □

Théorème d'unicité :

La solution $\bar{\mathbf{v}}$ du problème est unique

Démonstration :

Supposons que l'on ait 2 solutions $\bar{\mathbf{v}}$ et $\bar{\bar{\mathbf{v}}}$. D'après le théorème de caractérisation, on aurait donc :

$$\langle \bar{\mathbf{v}} - \mathbf{y}, \mathbf{v} \rangle = 0 \quad \forall \mathbf{v} \in \mathbf{V}$$

$$\langle \bar{\bar{\mathbf{v}}} - \mathbf{y}, \mathbf{v} \rangle = 0 \quad \forall \mathbf{v} \in \mathbf{V}.$$

Soit en soustrayant, pour $\mathbf{v} = \bar{\mathbf{v}} - \bar{\bar{\mathbf{v}}}$:

$$\langle \bar{\mathbf{v}} - \bar{\bar{\mathbf{v}}}, \mathbf{v} \rangle = \langle \bar{\mathbf{v}} - \bar{\bar{\mathbf{v}}}, \bar{\mathbf{v}} - \bar{\bar{\mathbf{v}}} \rangle = 0$$

donc $\|\bar{\mathbf{v}} - \bar{\bar{\mathbf{v}}}\|^2 = 0$.

Détermination de la solution

La solution $\bar{\mathbf{v}} = \sum_{i=1}^n \bar{\lambda}_i \mathbf{v}_i$ est obtenue en résolvant un système linéaire :

En reportant cette expression de $\bar{\mathbf{v}}$ dans le théorème de caractérisation, on obtient

$$\left\langle \sum_{i=1}^n \bar{\lambda}_i \mathbf{v}_i - \mathbf{y}, \mathbf{v} \right\rangle = 0 \quad \forall \mathbf{v} \in \mathbf{V},$$

et en particulier pour $\mathbf{v} = \mathbf{v}_j$ ($j \in \{1, \dots, n\}$), on obtient

$$\left\langle \sum_{i=1}^n \bar{\lambda}_i \mathbf{v}_i - \mathbf{y}, \mathbf{v}_j \right\rangle = 0, \quad j = 1, \dots, n.$$

d'où le système linéaire

$$\sum_{i=1}^n \bar{\lambda}_i \langle \mathbf{v}_i, \mathbf{v}_j \rangle = \langle \mathbf{y}, \mathbf{v}_j \rangle, \quad j = 1, \dots, n. \quad (S)$$

Le système (S) admet une solution unique si et seulement si le déterminant de la matrice $(\langle \mathbf{v}_i, \mathbf{v}_j \rangle)$ est différent de 0, c.à.d. si et seulement si les éléments $\mathbf{v}_1, \dots, \mathbf{v}_n$ sont *linéairement indépendants*.

Remarque : D'après le théorème précédent, la solution $\bar{\mathbf{v}}$ est toujours unique, mais si les $\mathbf{v}_1, \dots, \mathbf{v}_n$ ne sont pas linéairement indépendants, les $\bar{\lambda}_i$ tels que $\bar{\mathbf{v}} = \sum_{i=1}^n \bar{\lambda}_i \mathbf{v}_i$ ne sont pas uniques.

Système des équations normales

Le système (S) appelé *système des équations normales*, peut être obtenu d'une façon différente :

Posons

$$\begin{aligned} \Phi(\lambda) = \Phi(\lambda_1, \dots, \lambda_n) &= \sum_{k=1}^N \left(\sum_{i=1}^n \lambda_i v_i^k - y^k \right)^2 \\ &= \left\| \sum_{i=1}^n \lambda_i \mathbf{v}_i - \mathbf{y} \right\|^2. \end{aligned}$$

On veut minimiser la fonction Φ par rapport à $\lambda = (\lambda_1, \dots, \lambda_n)$. Cette fonction étant *convexe* et *dérivable*, elle sera minimum en $\bar{\lambda} = (\bar{\lambda}_1, \dots, \bar{\lambda}_n)$ si et seulement si

$$\frac{\partial \Phi}{\partial \lambda_j}(\bar{\lambda}) = 0, \quad j = 1, \dots, n.$$

Or on a

$$\frac{\partial \Phi}{\partial \lambda_j}(\bar{\lambda}) = 2 \sum_{k=1}^N \left(\sum_{i=1}^n \bar{\lambda}_i v_i^k - y^k \right) v_j^k, \quad j = 1, \dots, n,$$

on aura alors les conditions

$$\sum_{k=1}^N \left(\sum_{i=1}^n \bar{\lambda}_i v_i^k v_j^k \right) = \sum_{k=1}^N y^k v_j^k, \quad j = 1, \dots, n,$$

Soit encore

$$\sum_{i=1}^n \bar{\lambda}_i \left(\sum_{k=1}^N v_i^k v_j^k \right) = \sum_{k=1}^N y^k v_j^k, \quad j = 1, \dots, n,$$

qui est bien identique au système (S).

Application : Approximation d'une fonction (ou de données) au sens au sens des moindres carrés

Soit une fonction $f : [a, b] \rightarrow \mathbb{R}$ connue en t_k , soit de façon exacte $y^k = f(t_k)$, soit de façon approchée $y^k = f(t_k) + \epsilon_k$, $k = 1, \dots, N$.

On veut représenter f par une fonction de la forme

$$g = \sum_{i=1}^n \lambda_i g_i$$

où les g_i sont des fonctions données ($g_i : [a, b] \rightarrow \mathbb{R}$).

Exemple : $g_i(t) = t^{i-1}$, $i = 1, \dots, n$.

La fonction g est alors un polynôme de degré $n - 1$.

On se propose de minimiser la somme des carrés des erreurs en t_k :

$$\sum_{k=1}^N \left(\sum_{i=1}^n \lambda_i g_i(t_k) - y^k \right)^2.$$

Si l'on pose $v_i^k = g_i(t_k)$, on est ramené au problème déjà traité consistant à minimiser par rapport à $\lambda = (\lambda_1, \dots, \lambda_n)$ la quantité :

$$\sum_{k=1}^N \left(\sum_{i=1}^n \lambda_i v_i^k - y^k \right)^2 = \left\| \sum_{i=1}^n \lambda_i \mathbf{v}_i - \mathbf{y} \right\|^2.$$

Si les \mathbf{v}_i sont indépendants, la solution unique $\bar{\lambda}_1, \dots, \bar{\lambda}_n$ est obtenue en résolvant le système linéaire :

$$\sum_{i=1}^n \mathcal{M}_{ij} \bar{\lambda}_i = m_j, \text{quad } j = 1, \dots, n,$$

avec $\mathcal{M}_{ij} = \sum_{k=1}^N g_i(t_k) g_j(t_k) = \langle \mathbf{g}_i, \mathbf{g}_j \rangle$ ($\mathbf{g}_i = [g_i(t_1), \dots, g_i(t_N)]^T$),
et $m_j = \sum_{k=1}^N y^k g_j(t_k) = \langle \mathbf{y}, \mathbf{g}_j \rangle$.

Ecriture matricielle : Posons

$$\mathbf{A} = \begin{bmatrix} v_1^1 & \dots & v_n^1 \\ v_1^2 & \dots & v_n^2 \\ \vdots & & \vdots \\ v_1^N & \dots & v_n^N \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^N \end{bmatrix}, \quad \lambda = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_N \end{bmatrix}.$$

Le vecteur solution $\bar{\lambda}$ est alors obtenu en résolvant :

$$(\mathbf{A}^T \mathbf{A}) \bar{\lambda} = \mathbf{A}^T \mathbf{y} \tag{S'}$$

5.2 INTRODUCTION DE POIDS

Reprenons le problème de l'approximation de f par une fonction de la forme $\sum \lambda_i g_i$, mais supposons que l'on veuille minimiser

$$\sum_{k=1}^N \omega_k \left(\sum_{i=1}^n \lambda_i g_i(t_k) - y^k \right)^2,$$

où les ω_k sont des *poids positifs* qui permettent d'attribuer plus ou moins une importance à l'erreur commise en t_k .

Avec les mêmes notations, cela revient à minimiser

$$\sum_{k=1}^N \omega_k \left(\sum_{i=1}^n \lambda_i v_i^k - y^k \right)^2.$$

En introduisant un nouveau produit scalaire

$$\langle \mathbf{x}, \mathbf{y}_\Omega \rangle = \sum_{k=1}^N \omega_k x^k y^k$$

et la norme associée $\|\mathbf{y}\|_\Omega = \sqrt{\langle \mathbf{y}, \mathbf{y}_\Omega \rangle} = \sqrt{\sum_{k=1}^N \omega_k (y^k)^2}$, cela revient à minimiser

$$\left\| \sum_{i=1}^n \lambda_i \mathbf{v}_i - \mathbf{y} \right\|_\Omega = \|\mathbf{v} - \mathbf{y}\|_\Omega,$$

et la solution $\bar{\mathbf{v}} = \sum_{i=1}^n \bar{\lambda}_i \mathbf{v}_i$ est caractérisée par les mêmes conditions d'orthogonalité mais écrite avec le nouveau produit scalaire :

$$\langle \bar{\mathbf{v}} - \mathbf{y}, \mathbf{v} \rangle_\Omega = 0, \text{ pour tout } \mathbf{v} \in \mathbf{V},$$

c.à.d. explicitement :

$$\sum_{i=1}^n \bar{\lambda}_i \left(\sum_{k=1}^N \omega_k v_i^k v_j^k \right) = \sum_{k=1}^N \omega_k y^k v_j^k, \quad j = 1, \dots, n.$$

Ce système linéaire s'écrit matriciellement :

$$(\mathbf{A}^T \Omega \mathbf{A}) \bar{\boldsymbol{\lambda}} = \mathbf{A}^T \Omega \mathbf{y}$$

où Ω est une matrice $N \times N$ diagonale, d'éléments diagonaux ω_k .

Choix des poids Si les valeurs y^k peuvent être considérées comme des variables aléatoires d'écart type σ_k , il est conseillé de choisir $\omega_k = \frac{1}{\sigma_k^2}$.

Il arrive aussi que l'on introduise des poids du fait que certaines données sont de nature différente : par exemple les données relatives à la dérivée de la fonction. C'est ce qui apparaîtra dans l'exemple suivant :

Exemple 1 :

Supposons que l'on connaisse $f(0) = 1$, $f(\frac{1}{2}) = \frac{1}{2}$, $f(1) = \frac{1}{4}$ ainsi que $f'(0) = -2$.

On veut représenter $f(t)$ par $g(t) = \frac{\lambda_1}{1+t} + \lambda_2$ en choisissant λ_1, λ_2 qui minimisent :

$$\sum_{k=1}^3 (g(t_k) - f(t_k))^2 + \omega (g'(t_1) - f'(t_1))^2$$

($t_1 = 0, t_2 = \frac{1}{2}, t_3 = 1$).

Posons :

$$\mathbf{v}_1 = \begin{bmatrix} \frac{1}{1+t_1} \\ \frac{1}{1+t_2} \\ \frac{1}{1+t_3} \\ \frac{1}{(1+t_1)^2} \end{bmatrix} = \begin{bmatrix} 1 \\ \frac{2}{3} \\ \frac{1}{2} \\ -1 \end{bmatrix}, \quad \mathbf{v}_2 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{\Omega} = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & \omega \end{bmatrix}.$$

Le système s'écrit $(\mathbf{A}^T \mathbf{\Omega} \mathbf{A}) \bar{\lambda} = \mathbf{A}^T \mathbf{\Omega} \mathbf{y}$, i.e.

$$\underbrace{\begin{bmatrix} 1 & \frac{2}{3} & \frac{1}{2} & -1 \\ 1 & 1 & 1 & 0 \end{bmatrix}}_{\mathbf{A}^T} \underbrace{\begin{bmatrix} 1 & 1 \\ \frac{2}{3} & 1 \\ \frac{1}{2} & 1 \\ -\omega & 0 \end{bmatrix}}_{\mathbf{\Omega} \mathbf{A}} \underbrace{\begin{bmatrix} \bar{\lambda}_1 \\ \bar{\lambda}_2 \end{bmatrix}}_{\bar{\lambda}} = \underbrace{\begin{bmatrix} 1 & \frac{2}{3} & \frac{1}{2} & -1 \\ 1 & 1 & 1 & 0 \end{bmatrix}}_{\mathbf{A}^T} \underbrace{\begin{bmatrix} 1 \\ \frac{2}{2} \\ \frac{1}{4} \\ -2\omega \end{bmatrix}}_{\mathbf{\Omega} \mathbf{y}},$$

donc

$$\begin{bmatrix} \frac{61}{36} + \omega & \frac{13}{6} \\ \frac{13}{6} & 3 \end{bmatrix} \begin{bmatrix} \bar{\lambda}_1 \\ \bar{\lambda}_2 \end{bmatrix} = \begin{bmatrix} \frac{35}{24} + 2\omega \\ \frac{7}{4} \end{bmatrix}$$

Exemple 2 :

Etant donné un polynôme de degré 1 : $p(x) = ax + b$, on considère la quantité

$$\sum_{i=1}^3 (p(x_i) - y_i)^2 + \omega \left(\int_0^1 p(x) dx - m \right)^2,$$

où $x_1 = 0, x_2 = 0.5, x_3 = 1; y_1, y_2, y_3, m$ donnés, $\omega > 0$ donné.

Déterminer \bar{a} et \bar{b} qui minimisent cette quantité.

5.3 INTRODUCTION DE CONTRAINTES

On considère l'ensemble W suivante :

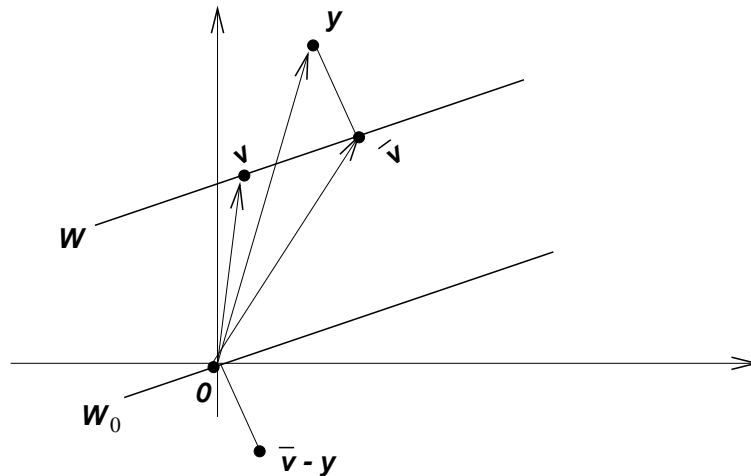
$$W = \left\{ \mathbf{v} \in \mathbf{V} \mid \mathbf{v} = \sum_{j=1}^n \lambda_j \mathbf{v}_j \text{ avec } \sum_{j=1}^n \alpha_{h_j} \lambda_j = 0, h = 1, \dots, m \right\}$$

W est un espace affine de \mathbb{R}^N qui est parallèle au sous-espace vectoriel W_0 suivant :

$$W_0 = \left\{ \mathbf{v} \in \mathbf{V} \mid \mathbf{v} = \sum_{i=1}^n \lambda_i \mathbf{v}_i \text{ avec } \sum_{j=1}^n \alpha_{h_j} \lambda_j = 0, h = 1, \dots, m \right\}$$

Un espace affine est plus restrictif qu'un espace vectoriel.

Etant donné $\mathbf{y} \in \mathbb{R}^N$, on cherche la meilleure approximation de \mathbf{y} dans W (au sens de la norme euclidienne), c.à.d. la projection de \mathbf{y} sur W .



un élément $\bar{\mathbf{v}} \in W$ est solution si et seulement si $\bar{\mathbf{v}} - \mathbf{y}$ est orthogonal à W_0 .

Pourquoi W est affine ?

soit $\mathbf{v} = \sum r_i \mathbf{v}_i \in W$ et $\mathbf{v}' = \sum r'_i \mathbf{v}_i \in W$, alors $\alpha \mathbf{v} + \beta \mathbf{v}' = \sum (\alpha r_i + \beta r'_i) \mathbf{v}_i$ et

$$\begin{aligned} \sum \alpha_{h_j} [\alpha r_j + \beta r'_j] &= \alpha \sum \alpha_{h_j} r_j + \beta \sum \alpha_{h_j} r'_j \\ &= \alpha \beta_h + \beta \beta_h \\ &= \beta_h \text{ seulement si } \alpha + \beta = 1 \\ &\implies W \text{ est espace affine.} \end{aligned}$$

(α, β ne sont pas n'importe quelle combinaison linéaire).

Application : $f : \mathbb{R} \rightarrow \mathbb{R}^d$ affine si

$$\left. \begin{array}{l} \forall a_1, \dots, a_m \in \mathbb{R} \text{ avec } \sum a_i = 1 \\ \forall t_1, \dots, t_m \in \mathbb{R} \end{array} \right\} f\left(\sum_{i=1}^m a_i t_i\right) = \sum_{i=1}^m a_i f(t_i).$$

Exemple : $m = 2$, $a_1 = \alpha$, $a_2 = 1 - \alpha$, alors pour $t = (1 - \alpha)t_1 + \alpha t_2$:

$$f(t) = (1 - \alpha)f(t_1) + \alpha f(t_2).$$

Problème : on ne connaît pas facilement une base de W_0 !

Méthode 1 (on connaît une base pour W_0) :

Si l'on peut trouver une base w_1, \dots, w_{n-m} de W_0 (on suppose ici que les conditions sont indépendante et que donc W_0 est de dimension $n - m$), alors on écrit :

$$\langle \bar{\mathbf{v}} - \mathbf{y}, \mathbf{w}_k \rangle = 0, \quad k = 1, \dots, n - m.$$

Ainsi un élément $\bar{\mathbf{v}} = \sum_{i=1}^n \bar{\lambda}_i \mathbf{v}_i$ est solution si et seulement si

$\begin{aligned} \sum_{j=1}^n \bar{\lambda}_j \langle \mathbf{v}_j, \mathbf{w}_k \rangle &= \langle \mathbf{y}, \mathbf{w}_k \rangle, \quad k = 1, \dots, n - m \\ \sum_{j=1}^n \alpha_{h_j} \bar{\lambda}_j &= \beta_h, \quad h = 1, \dots, m \end{aligned}$

Méthode 2 : base de W_0 inconnue

Utiliser les multiplicateurs de Lagrange [Strang90]

<p>Le Lagrangien :</p> $\mathcal{L}(\lambda, \mu) = \frac{\infty}{\epsilon} \oplus(\lambda) + \sum_{\langle = \infty}^{\hat{\Phi}} \mu_{\langle} \left[\sum_{ = \infty}^{\setminus} \alpha_{\langle } \lambda_{ } - \beta_{\langle} \right]$
<p>où</p> $\Phi(\lambda) = \left\ \sum_{i=1}^n \lambda_i \mathbf{v}_i - \mathbf{y} \right\ ^2.$

On exprime que $(\bar{\lambda}, \bar{\mu})$ est solution si et seulement si $(\bar{\lambda}, \bar{\mu})$ est un point de selle du lagrangien, c.à.d. si et seulement si

$$\begin{cases} \frac{\partial \mathcal{L}}{\partial \lambda_k}(\bar{\lambda}, \bar{\mu}) = 0 & k = 1, \dots, n \\ \frac{\partial \mathcal{L}}{\partial \mu_h}(\bar{\lambda}, \bar{\mu}) = 0 & h = 1, \dots, m \end{cases}$$

En revenant à la définition de γ , cela donne

$$\gamma_j = \left\langle \sum_{k=1}^n \bar{\lambda}_k \mathbf{v}_k - \mathbf{y}, \mathbf{v}_j \right\rangle = \sum_{h=1}^m \mu_h \alpha_{hj}, \quad j = 1, \dots, n,$$

Cette condition est identique à la première partie de (S), à la différence que $\mu_h = -\bar{\mu}_h$ (aucune importance car c'est un vecteur auxiliaire).

VI. Approximation spline

6.1 Approximation et lissage

Supposons maintenant que les données z_i soient entachées d'erreurs.

Elles peuvent être de la forme $z_i = f(t_i) + \epsilon_i$, où f est une fonction théorique sous-jacente (inconnue) que l'on cherche à retrouver et les ϵ_i sont les erreurs.

Celles-ci peuvent en général être assimilées à des réalisations d'une variable aléatoire dont les caractéristiques (fonction de densité, moyenne, écart type, ...) sont connues ou non.

2 exemples :

- variable aléatoire gaussienne de moyenne 0 et d'écart type e ,
- variable aléatoire de densité uniforme entre $-\epsilon_i$ et $+\epsilon_i$.

Pour retrouver f , on introduit :

- un critère de lissage,
- un critère d'approximation des données.

Le *critère de lissage* le plus utilisé est :

$$\mathcal{L}(x) = \int_a^b \left(x^{(q)}(t) \right)^2 dt$$

Comme critère d'approximation des données, on prend souvent :

$$\mathcal{A}(x) = \sum_{i=1}^n \omega_i (x(t_i) - z_i)^2$$

où $\omega_i = \frac{1}{\sigma_i^2}$, σ_i étant l'écart type de la donnée z_i .

Si toutes les données sont approximativement de même précision, on pourra prendre

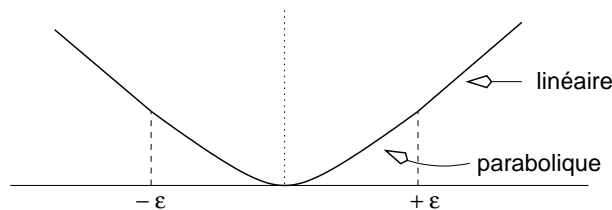
$$\mathcal{A}(x) = \sum_{i=1}^n (x(t_i) - z_i)^2$$

Il arrive néanmoins dans des cas particuliers que l'on prenne :

$$\mathcal{A}(x) = \sum_{i=1}^n \phi_i(x(t_i) - z_i)$$

où $\phi_i : \mathbb{R} \rightarrow \mathbb{R}^+$ est une fonction convexe, dérivable et non nécessairement la fonction $z \rightarrow \omega_i z^2$ comme plus haut.

Par exemple la fonction suivante :



On cherche alors \bar{x} qui minimise

$$\mathcal{L}(x) + \rho \mathcal{A}(x), \quad \text{pour } x \in X \text{ (à préciser).}$$

où $\rho > 0$ est un coefficient positif qui régit l'importance relative que l'on accorde caractère lisse de la solution et à l'approximation des données.

Selon l'origine du problème, il arrive que l'on considère plutôt (problème des moindres carrés régularisé) :

$$\mathcal{A}(x) + z \mathcal{L}(x)$$

ce qui est évidemment équivalent pour $z = \frac{1}{\rho}$.

Considérons **l'exemple type** suivant :

$$\text{minimiser } \left[\int_a^b (x''(t))^2 dt + \rho \sum_{i=1}^n (x(t_i) - z_i)^2 \right] \text{ pour } x \in H^2[a, b],$$

où $a = t_0 < t_1 < \dots < t_{n+1} = b$.

Posons encore $I_y = \{x \in H^2[a, b] / x(t_i) = y_i, i = 1, \dots, n\}$.

Il est clair que l'on a :

$$H^2[a, b] = \bigcup_{y \in \mathbb{R}^n} I_y.$$

On peut donc décomposer la minimisation en deux étapes :

$$\min_{y \in \mathbb{R}^n} \min_{x \in I_y} \left[\int_a^b (x''(t))^2 dt + \rho \sum_{i=1}^n (y_i - z_i)^2 \right]$$

qui s'écrit aussi

$$\min_{y \in \mathbb{R}^n} \left[\left(\min_{x \in I_y} \left[\int_a^b (x''(t))^2 dt \right] \right) + \rho \sum_{i=1}^n (y_i - z_i)^2 \right]$$

Posons $\Phi(y) = \min_{x \in I_y} \int_a^b [x''(t)]^2 dt$

Propriétés de la fonction Φ :

On a évidemment $\Phi(y) \geq 0, \forall y \in \mathbb{R}^n$.

Pour $n \geq 2$, on sait que la solution du problème

$$\text{minimiser } \int [x''(t)]^2 dt \text{ pour } x \in H^2[a, b], x(t_i) = y_i, i = 1, \dots, n$$

est la fonction unique $\sigma \in S_2$ telle que $\sigma(t_i) = y_i, i = 1, \dots, n$.

On a alors $\Phi(y) = \int_a^b [\sigma''(t)]^2 dt$.

Pour mieux voir la dépendance de σ de y , introduisons la base suivante de S_2 :

$$\sigma_j \in S_2, \quad \sigma_j(t_i) = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon} \end{cases}$$

Par linéarité, l'élément $\sigma \in S_2$ vérifiant $\sigma(t_i) = y_i, i = 1, \dots, n$ s'écrit

$$\sigma = \sum_{i=1}^n y_i \sigma_i$$

[NB : $\sigma_1, \dots, \sigma_n$ constitue la de S_2 duale des fonctionnelles $\phi_i : s \in S_2 \rightarrow s(t_i), i = 1, \dots, n$].

On a donc aussi

$$\sigma'' = \sum_{i=1}^n y_i \sigma_i''$$

et par conséquent

$$\Phi(y) = \int_a^b \left[\sum_{i=1}^n y_i \sigma_i''(t) \right]^2 dt.$$

En développant, on obtient

$$\begin{aligned} \Phi(y) &= \int_a^b \left[\sum_{i=1}^n \sum_{j=1}^n y_i y_j \sigma_i''(t) \sigma_j''(t) \right] dt \\ &= \sum \sum \left[\int_a^b \sigma_i''(t) \sigma_j''(t) dt \right] \cdot y_i y_j. \end{aligned}$$

On posant $\omega_{ij} = \int_a^b \sigma_i''(t) \sigma_j''(t) dt$ et $\Omega = (\omega_{ij})$ une matrice $n \times n$, on a donc

$$\Phi(y) = \sum \sum \omega_{ij} y_i y_j = \langle y, \Omega y \rangle.$$

La fonction Φ est donc quadratique est positive, donc convexe.

Calculons ω_{ij} : en considérant σ_j comme un élément de S_2 et σ_i comme un élément de $H^2[a, b]$, on obtient

$$\omega_{ij} = \int_a^b \sigma_i''(t) \sigma_j''(t) dt = \sum_{k=1}^n \Delta \sigma_j'''(t_k) \underbrace{\sigma_i(t_k)}_{\delta_{ik}} = \Delta \sigma_j'''(t_i).$$

Mais en échangeant les rôles de $\underbrace{\sigma_j}_{H^2}$ et $\underbrace{\sigma_i}_{S_2}$, on a $\omega_{ij} = \omega_{ji}$ et

$$\omega_{ij} = \Delta \sigma_j'''(t_i).$$

Comme la matrice Ω est symétrique, la dérivée de Φ en y est donnée par

$$g = \Phi'(y) = 2\Omega y \text{ (gradient de } \Phi).$$

$$\begin{aligned} \Phi : \mathbb{R}^n \longrightarrow \mathbb{R} : \quad \Phi(y) &= \Phi(y_1, \dots, y_n) = \sum_{i=1}^n \sum_{j=1}^n \omega_{ij} y_i y_j = \langle y, \Omega y \rangle \\ &= \dots + \omega_{ij} y_i y_j + \dots + \omega_{ji} y_j y_i + \dots \end{aligned}$$

$$\Phi'(y) = \begin{bmatrix} \frac{\partial \Phi}{\partial y_1} \\ \vdots \\ \frac{\partial \Phi}{\partial y_n} \end{bmatrix} = \begin{bmatrix} \sum_{k=1}^n 2\omega_{1k} y_k \\ \vdots \\ \sum_{k=1}^n 2\omega_{nk} y_k \end{bmatrix} = 2\Omega y.$$

$$\Phi' : \begin{pmatrix} \mathbb{R}^n \longrightarrow L(\mathbb{R}^n, \mathbb{R}) \\ y \longrightarrow 2\Omega y. \end{pmatrix}$$

On remarque que $g_i = 2 \sum_{j=1}^n \omega_{ij} y_j = 2 \sum_{j=1}^n \Delta \sigma_j'''(t_i) y_j = 2 \Delta \sigma'''(t_i)$ (car $\sigma(t) = \sum_{i=1}^n \sigma_i(t) y_i$).

Ainsi, le gradient de la fonction Φ en y est égal (à un coefficient près) au vecteur des discontinuités des dérivées 3^e de $\sigma = \sum_{i=1}^n y_i \sigma_i$ en t :

$$\begin{aligned} 2 \sum \Delta \sigma_j'''(t_i) y_j &= 2 \left(\sum \sigma_j'''(t_i^+) y_j - \sum \sigma_j'''(t_i^-) y_j \right) \\ &= 2 \left(\sum \lim_{\substack{h \rightarrow 0 \\ h > 0}} \sigma_j'''(t_i + h) y_j - \sum \lim_{\substack{h \rightarrow 0 \\ h > 0}} \sigma_j'''(t_i - h) y_j \right) \\ &= 2 \left(\lim_{\substack{h \rightarrow 0 \\ h > 0}} \sum \sigma_j'''(t_i + h) y_j - \lim_{\substack{h \rightarrow 0 \\ h > 0}} \sum \sigma_j'''(t_i - h) y_j \right) \\ &= 2 \left(\lim_{\substack{h \rightarrow 0 \\ h > 0}} \sigma'''(t_i + h) - \lim_{\substack{h \rightarrow 0 \\ h > 0}} \sigma'''(t_i - h) \right) \\ &= 2 \Delta \sigma'''(t_i). \end{aligned}$$

Il reste à étudier la minimisation de

$$\Phi(y) + \rho \|y - z\|^2, \quad \text{pour } y \in \mathbb{R}^n$$

où $\|y - z\|^2 = \sum_{i=1}^n (y_i - z_i)^2$.

Il s'agit d'une fonction convexe quadratique. Elle est minimum en \bar{y} ssi ses dérivées partielles en ce points sont nulles, i.e. :

$$g_i + 2\rho(\bar{y} - z_i) = 0, \quad i = 1, \dots, n \quad ((g_i) = \Phi'(y)).$$

Soit

$$(\Omega \bar{y})_i + \rho(\bar{y}_i - z_i) = 0,$$

donc

$$\Omega \bar{y} + \rho(\bar{y} - z) = 0. \quad (*)$$

Ainsi on a :

$$\bar{y} = \left(I + \frac{1}{\rho}\Omega\right)^{-1}z.$$

On voit que le vecteur \bar{y} dépend linéairement du vecteur z .

La solution σ est alors la spline d'interpolation correspondante à \bar{y} , c.à.d. telle que

$$\sigma \in S_2 \text{ et } \sigma(t_i) = \bar{y}_i, \quad i = 1, \dots, n.$$

Le système (*) peut être exprimé directement en terme de σ : (car $(\Omega \bar{y})_i = \Delta \sigma'''(t_i)$)

$$\Delta \sigma'''(t_i) + \rho(\sigma(t_i) - z_i) = 0, \quad i = 1, \dots, n.$$

En conclusion, la **solution σ du problème** est caractérisée par

- 1) la restriction de σ à $[t_i, t_{i+1}]$ est un polynôme de degré 3, pour $i = 1, \dots, n - 1$,
- 2) les restrictions de σ $[a, t_1]$ et $[t_n, b]$ sont des polynômes de degré 1,
- 3) $\sigma \in C^2[a, b]$,
- 4) $\Delta \sigma'''(t_i) + \rho(\sigma(t_i) - z_i) = 0, \quad i = 1, \dots, n.$

De manière générale, on peut étudier le problème de minimisation suivant :

$$(\mathcal{P}_L) \quad \Phi(y) + \rho\Psi(y)$$

$$\text{où } \Phi(y) = \min_{\substack{x \in X \\ Ax=y}} |x|^2, \quad \Psi(y) = \sum_{i=1}^n \phi_i(y_i - z_i),$$

et $|\cdot|$ est une semi-norme sur X associée à un produit scalaire (\cdot, \cdot) , A est une application linéaire continue et surjective de X dans \mathbb{R}^n , $\phi_i : \mathbb{R} \rightarrow \mathbb{R}$ convexe et dérivable.

On met ensuite en évidence un sous-espace S "orthogonal" pour (\cdot, \cdot) à

$$I_0 = \ker(A) = \{x / Ax = 0\},$$

ainsi qu'une application linéaire $\Lambda : S \rightarrow \mathbb{R}^n$ telle que

$$(s, x) = \langle \Lambda(s), \Lambda(x) \rangle, \quad \forall s \in S, \forall x \in X.$$

On montre que pour $y = (y_1, \dots, y_n)$ quelconque, il existe un seul élément $\sigma \in S$ vérifiant $\Lambda(\sigma) = y$.

Cet élément dépend linéairement de y (on peut l'écrire $\sigma = \sum_{i=1}^n y_i \sigma_i$).

On notera $\sigma = \mathcal{S}(y)$, où $\mathcal{S} : \mathbb{R}^n \rightarrow S$ l'opérateur spline d'interpolation.

On sait que $\sigma = \mathcal{S}(y)$ est l'unique solution du problème de la minimisation de $|x|^2$ avec les conditions $x(t_i) = y_i, i = 1, \dots, n$.

On a donc

$$\Phi(y) = (\sigma, \sigma) = \langle \Lambda(\sigma), \Lambda(\sigma) \rangle = \langle \Lambda \mathcal{S}(y), y \rangle .$$

Posons $\Omega = \Lambda \circ \mathcal{S} (\mathbb{R}^n \rightarrow \mathbb{R}^n)$.

Ainsi $\Phi'(y) = 2\Omega y = 2\Lambda (\mathcal{S}(y)) = 2\Lambda(\sigma)$, si $\sigma = \mathcal{S}(y)$.

La solution de (\mathcal{P}_L) est caractérisée par

$$\Phi'(y) + \rho \Psi'(y) = 0,$$

donc

$$2\Omega y + \rho \Psi'(y) = 0.$$

Ainsi la solution du problème :

$$\text{minimiser } (|x|^2 + \rho \Psi(Ax))$$

sera caractérisée par les conditions

$\begin{aligned} \sigma &\in S \\ 2\Lambda(\sigma) + \rho \Psi'(A\sigma) &= 0 \end{aligned}$
--

6.2 ALGORITHMES

On considère ici l'exemple de base

$$\min_{x \in H^2[a,b]} \left[\int_a^b (x''(t))^2 dt + \rho \sum_{i=1}^n (x(t_i) - z_i)^2 \right].$$

On veut calculer la fonction $\sigma \in S_2$ vérifiant

$$\Delta \sigma'''(t_i) + \rho(\sigma(t_i) - z_i) = 0, \quad i = 1, \dots, n.$$

Pour simplifier, on suppose que $t_{i+1} - t_i = h, i = 1, \dots, n - 1$.

Dans chaque intervalle $[t_i, t_{i+1}], i = 1, \dots, n - 1$, la restriction de σ est un polynôme P_i de degré déterminé par $y_i, y_{i+1}, y'_i := P'_i(t_i), y'_{i+1} := P'_i(t_{i+1})$.

L'objectif est de calculer :

$$y_i \text{ et } y'_i \text{ pour } i = 1, \dots, n.$$

NB : La connaissance de y_1 et y'_1 permet de calculer $P_0(t)$ qui est un polynôme de degré 1 (de même pour y_n et y'_n pour $P_{n-1}(t)$).

Par le choix même des inconnues, le raccordement C^1 est assuré en t_i . On a :

$$\begin{aligned} P_{i-1}(t_i) &= P_i(t_i) \quad (= y_i), \quad i = 1, \dots, n-1 \\ P'_{i-1}(t_i) &= P'_i(t_i) \quad (= y'_i), \quad i = 1, \dots, n-1 \end{aligned}$$

On va former $2n$ équation pour le calcul des y_i et y'_i , $i = 1, \dots, n$, en exprimant :

1) les raccordement de la dérivée seconde

$$P''_{i-1}(t_i) = P''_i(t_i), \quad i = 1, \dots, n-1$$

2) l'équation de lissage :

$$P_i'''(t_i) - P_{i-1}'''(t_i) = -\rho(y_i - z_i), \quad i = 1, \dots, n-1.$$

L'équation de raccordement de la dérivée secondes a déjà été écrite pour l'interpolation (voir sect. 4.14) :

$$y'_{i-1} + 4y'_i + y'_{i+1} - \frac{3}{h}(y_{i+1} - y_{i-1}) = 0, \quad i = 2, \dots, n-1$$

avec les deux conditions particulières en t_1 et t_n s'exprimant par

$$P_1'''(t_1) = 0 \text{ et } P_{n-1}'''(t_n) = 0$$

pouvant s'écrire :

$$\begin{aligned} 2y'_1 + y'_2 - \frac{3}{h}(y_2 - y_1) &= 0 \\ y'_{n-1} + 2y'_n - \frac{3}{h}(y_n - y_{n-1}) &= 0 \end{aligned}$$

Pour écrire l'équation de lissage, rappelons que (sect. 4.13)

$$\begin{aligned} P_i''(t_i) &= \frac{2}{h}(3m_i - 2y'_i - y'_{i+1}) \\ P_i''(t_{i+1}) &= \frac{2}{h}(-3m_i + 2y'_i + 2y'_{i+1}) \end{aligned} \quad m_i = \frac{y_{i+1} - y_i}{h}$$

(Il faut noter que $P_i \in \mathcal{P}_3 \text{i} \implies P_i'' \in \mathcal{P}_1 \implies P_i'''$ est constant).

Donc, si $\nu_i = P_i'''(t)$ pour $t \in]t_i, t_{i+1}[$ on a

$$\nu_i = \frac{P_i''(t_{i+1}) - P_i''(t_i)}{h} = \frac{6}{h^2}(y'_i + y'_{i+1} - 2m_i)$$

L'équation de lissage s'écrit donc

$$\frac{6}{h^2} \left(y'_{i+1} - y'_{i-1} - \frac{2}{h}(y_{i+1} - 2y_i + y_{i-1}) \right) + \rho(y_i - z_i) = 0, \quad i = 2, \dots, n-1.$$

En t_1 , l'équation de lissage prend une forme particulière du fait que $\Delta\sigma'''(t_1) = P_1'''(t_1) - \underbrace{0}_{P_0(t_1)=0} = \frac{6}{h^2}(y'_1 - (y'_2 - 2m_1))$, on obtient donc :

$$\frac{6}{h^2} \left(y'_1 + y'_2 - \frac{2}{h}(y_2 - y_1) \right) + \rho(y_1 - z_1) = 0.$$

Et de même en t_n , $\Delta\sigma'''(t_n) = 0 - P_{n-1}'''(t_n)$

$$\frac{6}{h^2} \left(y'_{n-1} + y'_n - \frac{2}{h}(y_n - y_{n-1}) \right) + \rho(y_n - z_n) = 0.$$

On peut écrire toutes ces équations sous la forme d'un système tridiagonale par blocs 2×2 .

Posons $\delta_i := hy'_i$ et $\omega := \rho \frac{h^3}{2}$, les équations peuvent donc s'écrire :

$$y'_{i-1} + 4y'_i + y'_{i+1} - \frac{3}{h}(y_{i+1} - y_{i-1}) = 0,$$

les deux membres multipliés par h donne

$$hy'_{i-1} + 4hy'_i + hy'_{i+1} - 3y_{i+1} + 3y_{i-1} = 0$$

d'où

$$\mathbf{3y}_{i-1} + \delta_{i-1} + \mathbf{0} \cdot \mathbf{y}_i + \mathbf{4\delta}_i - \mathbf{3y}_{i+1} + \delta_{i+1} = \mathbf{0} \quad \mathbf{i} = \mathbf{2} \dots, \mathbf{n} - \mathbf{1}.$$

Les équations

$$\frac{6}{h^2} \left(y'_{i+1} - y'_{i-1} - \frac{2}{h}(y_{i+1} - 2y_i + y_{i-1}) \right) + \rho(y_i - z_i) = 0$$

s'écrivent (en multipliant par $\frac{h^2}{2}$) :

$$3h(y'_{i+1} - y'_{i-1}) - 6y_{i+1} + 12y_i - 6y_{i-1} + \frac{\rho h^3}{2}(y_i - z_i) = 0$$

d'où

$$-\mathbf{6y}_{i-1} - \mathbf{3\delta}_{i-1} + (\mathbf{12} + \omega)\mathbf{y}_i + \mathbf{0} \cdot \delta_i - \mathbf{6y}_{i+1} + \mathbf{3\delta}_{i+1} = \omega\mathbf{z}_i, \quad \mathbf{i} = \mathbf{2} \dots, \mathbf{n} - \mathbf{1}.$$

Soit matriciellement : ($i = 2 \dots, n - 1$)

$$\begin{array}{ccccccc} \begin{bmatrix} -6 & -3 \\ 3 & 1 \end{bmatrix} & \begin{bmatrix} y_{i-1} \\ \delta_{i-1} \end{bmatrix} & + & \begin{bmatrix} 12 + \omega & 0 \\ 0 & 4 \end{bmatrix} & \begin{bmatrix} y_i \\ \delta_i \end{bmatrix} & + & \begin{bmatrix} -6 & 3 \\ -3 & 1 \end{bmatrix} & \begin{bmatrix} y_{i+1} \\ \delta_{i+1} \end{bmatrix} & = & \begin{bmatrix} \omega z_i \\ 0 \end{bmatrix} \\ \hline A & \Lambda_{i-1} & + & B & \Lambda_i & + & A^T & \Lambda_{i+1} & = & D_i \end{array}$$

Les équations en t_1 peuvent s'écrire

$$\begin{aligned} (6 + \omega)y_1 + 3\delta_1 - 6y_2 + 3\delta_2 &= \omega z_1 \\ -3y_1 - 2\delta_1 + 3y_2 - \delta_2 &= 0. \end{aligned}$$

Soit matriciellement

$$\begin{array}{ccccccc} \begin{bmatrix} 6 + \omega & 3 \\ 3 & 2 \end{bmatrix} & \begin{bmatrix} y_1 \\ \delta_1 \end{bmatrix} & + & \begin{bmatrix} -6 & 3 \\ -3 & 1 \end{bmatrix} & \begin{bmatrix} y_2 \\ \delta_2 \end{bmatrix} & = & \begin{bmatrix} \omega z_1 \\ 0 \end{bmatrix} \\ \hline B' & \Lambda_1 & + & C & \Lambda_2 & = & D_1 \end{array}$$

et en t_n :

$$\begin{array}{ccccccc} \begin{bmatrix} -6 & -3 \\ 3 & 1 \end{bmatrix} & \begin{bmatrix} y_{n-1} \\ \delta_{n-1} \end{bmatrix} & + & \begin{bmatrix} 6 + \omega & -3 \\ -3 & 2 \end{bmatrix} & \begin{bmatrix} y_n \\ \delta_n \end{bmatrix} & = & \begin{bmatrix} \omega z_n \\ 0 \end{bmatrix} \\ \hline A & \Lambda_{n-1} & + & B'' & \Lambda_n & = & D_n \end{array}$$

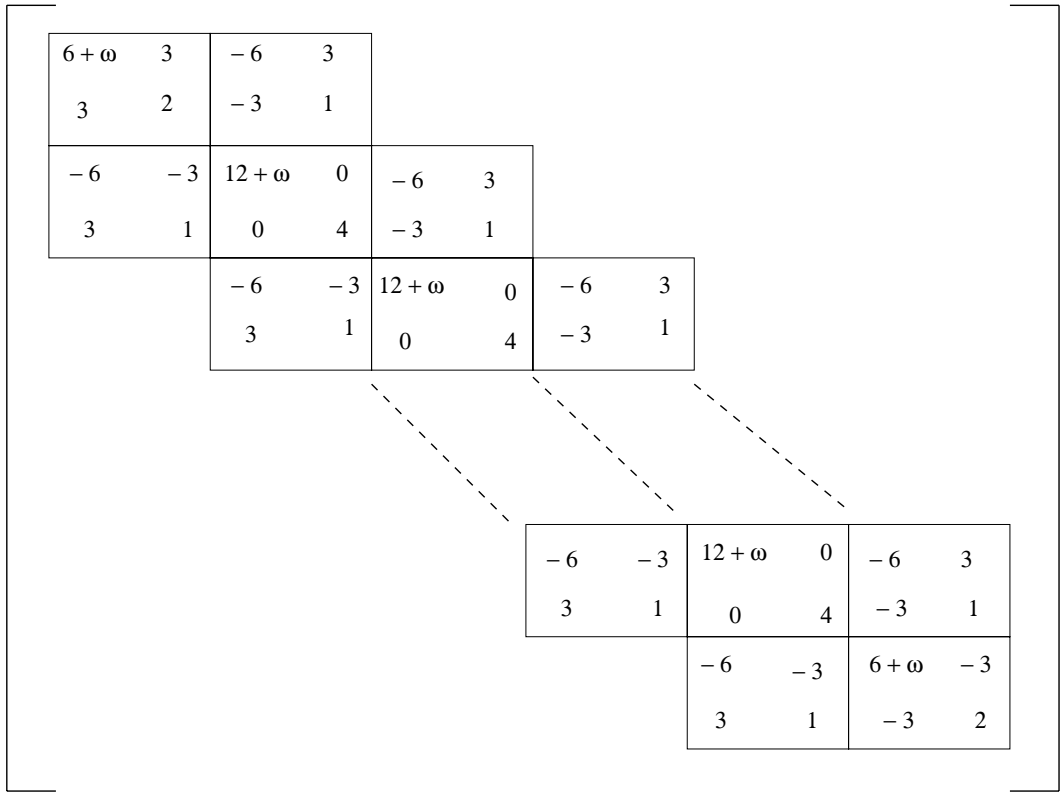
d'où

$$\begin{bmatrix} B' & C & & & & & \\ A & B & C & & & & \\ & A & B & C & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & A & B & C & \\ & & & & A & B'' & \end{bmatrix} \begin{bmatrix} \Lambda_1 \\ \Lambda_2 \\ \Lambda_3 \\ \vdots \\ \Lambda_{n-1} \\ \Lambda_n \end{bmatrix} = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ \vdots \\ D_{n-1} \\ D_n \end{bmatrix}.$$

Mais $C = A^T$, B, B', B'' sont des matrices symétriques, donc la matrice du système est elle aussi symétrique et tridiagonale :

$$\begin{bmatrix} B' & A^T & & & & & \\ A & B & A^T & & & & \\ & A & B & A^T & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & A & B & A^T & \\ & & & & A & B'' & \end{bmatrix} \begin{bmatrix} \Lambda_1 \\ \Lambda_2 \\ \Lambda_3 \\ \vdots \\ \Lambda_{n-1} \\ \Lambda_n \end{bmatrix} = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ \vdots \\ D_{n-1} \\ D_n \end{bmatrix}.$$

On peut écrire la matrice de ce système sous la forme :



Bibliographie

G. FARIN, Curves and Surfaces for CAGD, a practical guide, Academic Press, 6th edition 2001.

G. FARIN, Courbes et Surfaces pour la CGAO, Masson, 1992.

G. FARIN, NURB Curves and Surfaces, AK Peters 1994.

G. FARIN, D. Hansford, The essentials of CAGD, AK Peters 2001.

J. HOSCHEK, D. LASSER, Fundamentals of Computer Aided Geometric Design, AK Peters 1993.

J.J. RISLER, Méthodes mathématiques pour la CAO, Masson, 1991

J.C. LEON,

G. STRANG, Introduction to applied mathematics, Wellesley-Cambridge Press, 1990.