

Programmation Numérique

Polytech Lille — IS 3 — Cours 2

17 octobre 2023

Interpolation polynomiale

Thm Étant donnés $n + 1$ points

$$\begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, \begin{pmatrix} x_1 \\ y_1 \end{pmatrix}, \dots, \begin{pmatrix} x_n \\ y_n \end{pmatrix},$$

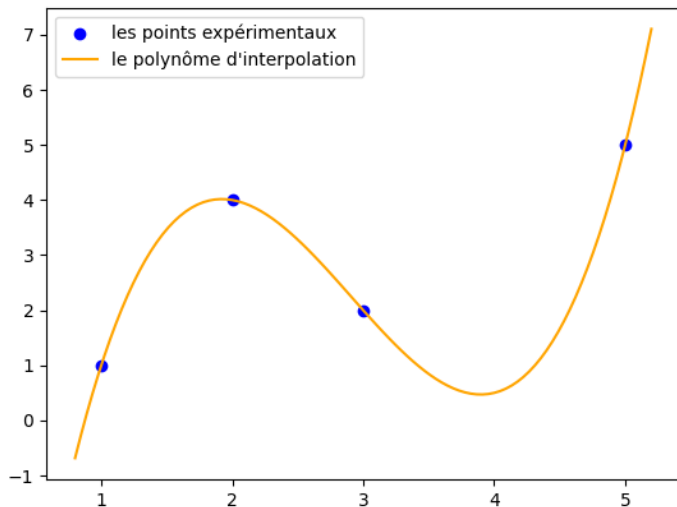
d'abscisses distinctes deux-à-deux, il existe un unique polynôme $P(z)$ de degré au plus n , dont le graphe passe par chacun des points. Ce polynôme est appelé polynôme d'interpolation.

i	0	1	2	3
x_i	1	2	3	5
y_i	1	4	2	5

Le polynôme d'interpolation est

$$P(z) = -\frac{25}{2} + \frac{247}{12}z - 8z^2 + \frac{11}{12}z^3.$$

Graphiquement



- En algèbre, l'interpolation polynomiale est un non problème
- En algorithmique numérique, les algorithmes prennent des précautions pour éviter des propagations catastrophiques d'erreurs d'arrondis (non à la base des monômes !)
- Dans les applications, on a souvent besoin de propriétés supplémentaires (exemple d'une fonction de survie qui doit être décroissante et comprise entre 0 et 1). Besoin d'interpolation polynomiale mais faite de façon intelligente : les splines.

En algèbre, c'est un non problème

On attribue à $P(z)$ des coefficients inconnus

$$P(z) = a_0 + a_1 z + \cdots + a_n z^n$$

On écrit les conditions que doivent vérifier les a_i

$$\begin{aligned} y_0 &= a_0 + a_1 x_0 + \cdots + a_n x_0^n, \\ y_1 &= a_0 + a_1 x_1 + \cdots + a_n x_1^n, \\ &\vdots \\ y_n &= a_0 + a_1 x_n + \cdots + a_n x_n^n. \end{aligned}$$

En algèbre, c'est un non problème

On attribue à $P(z)$ des coefficients inconnus

$$P(z) = a_0 + a_1 z + \cdots + a_n z^n$$

On écrit les conditions que doivent vérifier les a_i

$$\underbrace{\begin{pmatrix} 1 & x_0 & \cdots & x_0^n \\ 1 & x_1 & \cdots & x_1^n \\ \vdots & & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^n \end{pmatrix}}_A \underbrace{\begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix}}_v = \underbrace{\begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}}_b .$$

Thm *La matrice de Vandermonde A est inversible si les x_i sont distincts deux-à-deux.*

En algorithmique numérique, c'est moins simple

Idée *La méthode par la matrice de Vandermonde cherche à décomposer $P(z)$ dans la base des monômes. Ce n'est pas une bonne idée.*

Les algorithmes numériques évaluent $P(z)$ sans calculer ses coefficients :

- La formule de Lagrange (1800)
- L'algorithme de Neville (1900)
- Les différences divisées de Newton (1700)

L'algorithme de Neville (1900)

i	0	1	2	3
x_i	1	2	3	5
y_i	1	4	2	5

($n = 3$)

Idée 1 On définit $P_{i,j}$ comme le polynôme d'interpolation dont le graphe passe par les points d'indices $i, i-1, \dots, i-j$.

i	x_i	j			
		0	1	2	3
0	1	1			
1	2	4	$3z - 2$		
2	3	2	$-2z + 8$	$-\frac{5}{2}z^2 + \frac{21}{2}z - 7$	
3	5	5	$\frac{3}{2}z - \frac{5}{2}$	$\frac{7}{6}z^2 - \frac{47}{6}z + 15$	$\frac{11}{12}z^3 - 8z^2 + \frac{247}{12}z - \frac{25}{2}$

L'algorithme de Neville (1900)

i	0	1	2	3
x_i	1	2	3	5
y_i	1	4	2	5

$(n = 3)$

Idée 1 On définit $P_{i,j}$ comme le polynôme d'interpolation dont le graphe passe par les points d'indices $i, i-1, \dots, i-j$.

Idée 2 Il y a une formule pour construire le tableau de polynômes colonne par colonne

$$\begin{aligned} P_{i,0} &= y_i & (0 \leq i \leq n), \\ P_{i,j} &= \frac{(x_i - z) P_{i-1,j-1} + (z - x_{i-j}) P_{i,j-1}}{x_i - x_{i-j}} & (1 \leq j \leq n, \quad j \leq i \leq n). \end{aligned}$$

L'algorithme de Neville (1900)

i	0	1	2	3
x_i	1	2	3	5
y_i	1	4	2	5

$(n = 3)$

Idée 1 On définit $P_{i,j}$ comme le polynôme d'interpolation dont le graphe passe par les points d'indices $i, i-1, \dots, i-j$.

Idée 2 Il y a une formule pour construire le tableau de polynômes colonne par colonne

$$\begin{aligned} P_{i,0} &= y_i & (0 \leq i \leq n), \\ P_{i,j} &= \frac{(x_i - z) P_{i-1,j-1} + (z - x_{i-j}) P_{i,j-1}}{x_i - x_{i-j}} & (1 \leq j \leq n, \quad j \leq i \leq n). \end{aligned}$$

Idée 3 Si on donne une valeur numérique \bar{z} à z , la formule construit un tableau de nombres et on obtient la valeur numérique $P(\bar{z})$

Les différences divisées de Newton (1700)

i	0	1	2	3
x_i	1	2	3	5
y_i	1	4	2	5

$(n = 3)$

Rappel Le tableau P de l'algorithme de Neville

i	x_i	j			
		0	1	2	3
0	1	1			
1	2	4	$3z - 2$		
2	3	2	$-2z + 8$	$-\frac{5}{2}z^2 + \frac{21}{2}z - 7$	
3	5	5	$\frac{3}{2}z - \frac{5}{2}$	$\frac{7}{6}z^2 - \frac{47}{6}z + 15$	$\frac{11}{12}z^3 - 8z^2 + \frac{247}{12}z - \frac{25}{2}$

Les différences divisées de Newton (1700)

i	0	1	2	3
x_i	1	2	3	5
y_i	1	4	2	5

($n = 3$)

Idée 1 Ne calculer que les coefficients dominants $c_{i,j}$ des polynômes

i	x_i	j			
		0	1	2	3
0	1	1			
1	2	4	$3z - 2$		
2	3	2	$-2z + 8$	$-\frac{5}{2}z^2 + \frac{21}{2}z - 7$	
3	5	5	$\frac{3}{2}z - \frac{5}{2}$	$\frac{7}{6}z^2 - \frac{47}{6}z + 15$	$\frac{11}{12}z^3 - 8z^2 + \frac{247}{12}z - \frac{25}{2}$

Les différences divisées de Newton (1700)

i	0	1	2	3
x_i	1	2	3	5
y_i	1	4	2	5

$(n = 3)$

Idée 1 Ne calculer que les coefficients dominants $c_{i,j}$ des polynômes

i	x_i	j			
		0	1	2	3
0	1	1			
1	2	4	3		
2	3	2	-2	$-\frac{5}{2}$	
3	5	5	$\frac{3}{2}$	$\frac{7}{6}$	$\frac{11}{12}$

Les différences divisées de Newton (1700)

i	0	1	2	3
x_i	1	2	3	5
y_i	1	4	2	5

$(n = 3)$

Idée 1 Ne calculer que les coefficients dominants $c_{i,j}$ des polynômes

$$\begin{aligned}c_{i,0} &= y_i & (0 \leq i \leq n), \\c_{i,j} &= \frac{c_{i,j-1} - c_{i-1,j-1}}{x_i - x_{i-j}} & (1 \leq j \leq n, \quad j \leq i \leq n).\end{aligned}$$

Les différences divisées de Newton (1700)

i	0	1	2	3
x_i	1	2	3	5
y_i	1	4	2	5

$(n = 3)$

Idée 1 Ne calculer que les coefficients dominants $c_{i,j}$ des polynômes

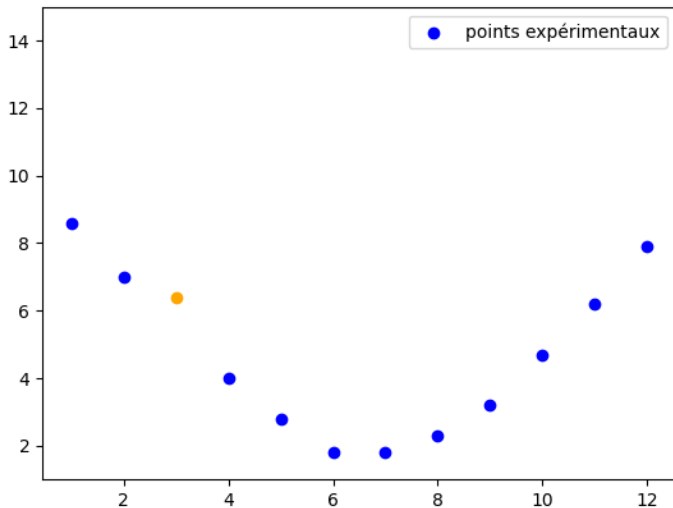
$$\begin{aligned}c_{i,0} &= y_i & (0 \leq i \leq n), \\c_{i,j} &= \frac{c_{i,j-1} - c_{i-1,j-1}}{x_i - x_{i-j}} & (1 \leq j \leq n, \quad j \leq i \leq n).\end{aligned}$$

Idée 2 Il existe une formule pour le polynôme d'interpolation

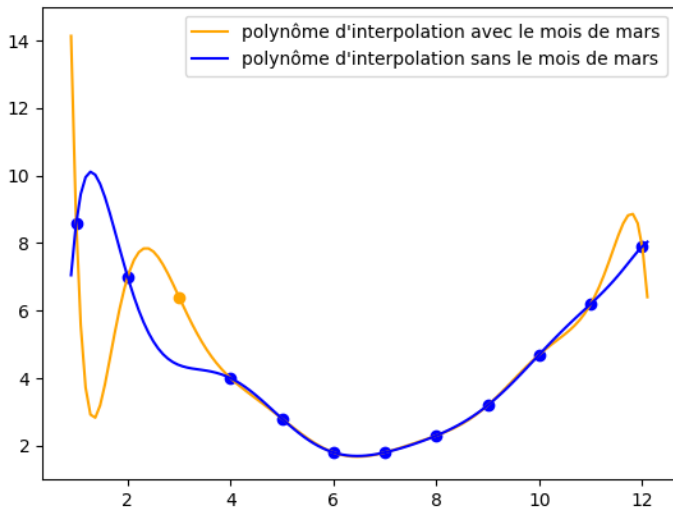
$$\begin{aligned}P_{n,n}(z) &= c_{0,0} + c_{1,1}(z - x_0) + c_{2,2}(z - x_0)(z - x_1) + \cdots + \\&\quad c_{n,n}(z - x_0)(z - x_1) \cdots (z - x_{n-1}).\end{aligned}$$

Idée 3 Le tableau des $c_{i,j}$ n'est calculé qu'une fois. Une variante du schéma de Horner-Ruffini permet d'évaluer $P_{n,n}(z)$

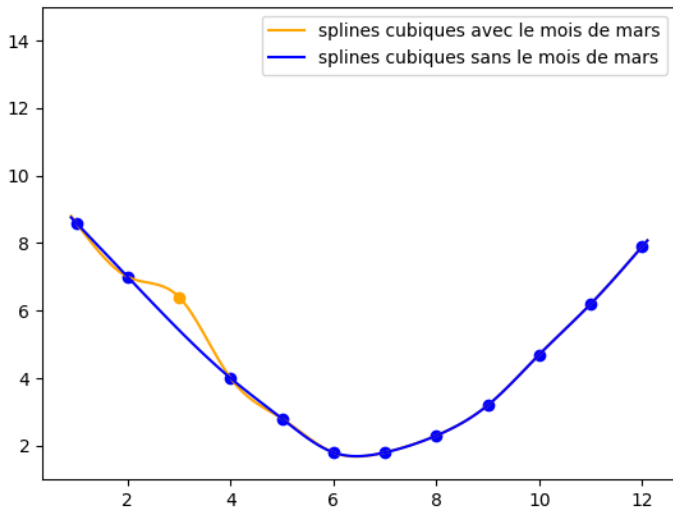
Les splines



Les splines



Les splines



Les splines : définitions

On suppose donnés $n + 1$ points

$$\begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, \begin{pmatrix} x_1 \\ y_1 \end{pmatrix}, \dots, \begin{pmatrix} x_n \\ y_n \end{pmatrix}.$$

Les abscisses sont appelés les *nœuds*.

On les suppose ordonnées par ordre croissant

$$a = x_0 < x_1 < \dots < x_n = b.$$

Les *nœuds* x_1, \dots, x_{n-1} sont appelés *nœuds intérieurs*.

Les *nœuds* $a = x_0$ et $x_n = b$ sont appelés *extrémités*.

Def Une *spline cubique* est une fonction $s(x)$ deux fois dérivable sur l'intervalle $[a, b]$, qui se réduit à un polynôme de degré 3 sur chaque sous-intervalle $[x_i, x_{i+1}]$ de $[a, b]$.

Les splines : définitions

On suppose donnés $n + 1$ points

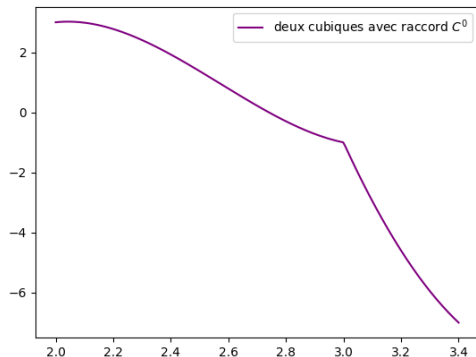
$$\begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, \begin{pmatrix} x_1 \\ y_1 \end{pmatrix}, \dots, \begin{pmatrix} x_n \\ y_n \end{pmatrix}.$$

Une spline cubique est une fonction définie par morceaux.

Les raccords entre les morceaux sont C^2 .

$$s(z) = \begin{cases} s_0(z) &= a_0 + b_0(z - x_0) + c_0(z - x_0)^2 + d_0(z - x_0)^3, & (z \leq x_1), \\ s_1(z) &= a_1 + b_1(z - x_1) + c_1(z - x_1)^2 + d_1(z - x_1)^3, & (x_1 \leq z \leq x_2), \\ &\vdots \\ s_i(z) &= a_i + b_i(z - x_i) + c_i(z - x_i)^2 + d_i(z - x_i)^3, & (x_i \leq z \leq x_{i+1}), \\ &\vdots \\ s_{n-1}(z) &= a_{n-1} + b_{n-1}(z - x_{n-1}) + c_{n-1}(z - x_{n-1})^2 + \\ &\quad + d_{n-1}(z - x_{n-1})^3, & (x_{n-1} \leq z). \end{cases}$$

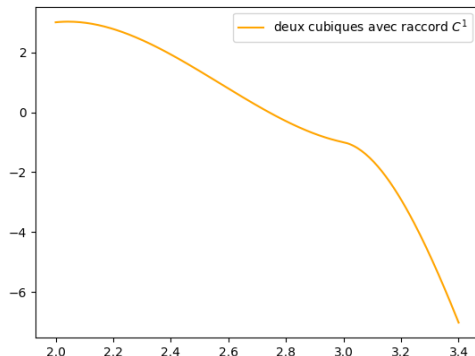
Notion de raccord C^2



Le raccord en $(x_1, y_1) = (3, -1)$ est C^0 mais pas C^1 :

$$s_0(x_1) = s_1(x_1).$$

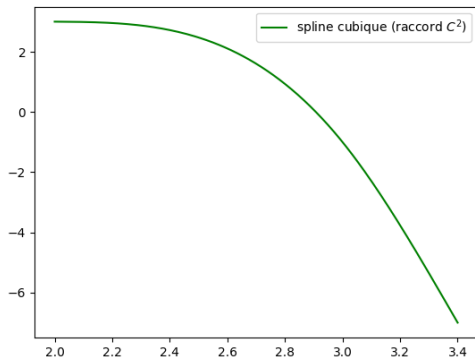
Notion de raccord C^2



Le raccord en $(x_1, y_1) = (3, -1)$ est C^1 mais pas C^2 :

$$s_0(x_1) = s_1(x_1), \quad s'_0(x_1) = s'_1(x_1).$$

Notion de raccord C^2



Le raccord en $(x_1, y_1) = (3, -1)$ est C^2 :

$$s_0(x_1) = s_1(x_1), \quad s'_0(x_1) = s'_1(x_1), \quad s''_0(x_1) = s''_1(x_1).$$

Conditions sur les coefficients

Il y a $n + 1$ nœuds et n cubiques.

Chaque cubique dépend de 4 coefficients à déterminer :

$$s_i(z) = a_i + b_i (z - x_i) + c_i (z - x_i)^2 + d_i (z - x_i)^3.$$

Il y a donc $4n$ inconnues. On cherche $4n$ équations.

Raccords C^0 Chaque cubique s_i passe par x_i et x_{i+1} :

$$\left. \begin{array}{l} s_i(x_i) = y_i \\ s_i(x_{i+1}) = y_{i+1} \end{array} \right\} (0 \leq i \leq n-1) \quad \text{2 n équations}$$

Conditions sur les coefficients

Il y a $n + 1$ nœuds et n cubiques.

Chaque cubique dépend de 4 coefficients à déterminer :

$$s_i(z) = a_i + b_i (z - x_i) + c_i (z - x_i)^2 + d_i (z - x_i)^3.$$

Il y a donc $4n$ inconnues. On cherche $4n$ équations.

Raccords C^1 En chacun des **$n-1$** nœuds intérieurs x_i , les dérivées des cubiques s_{i-1} et s_i sont égales :

$$s'_{i-1}(x_i) = s'_i(x_i) \quad (1 \leq i \leq n-1) \quad \text{ **$n-1$ équations**}$$

Conditions sur les coefficients

Il y a $n + 1$ nœuds et n cubiques.

Chaque cubique dépend de 4 coefficients à déterminer :

$$s_i(z) = a_i + b_i (z - x_i) + c_i (z - x_i)^2 + d_i (z - x_i)^3.$$

Il y a donc $4n$ inconnues. On cherche $4n$ équations.

Raccords C^2 En chacun des **$n-1$** nœuds intérieurs x_i , les dérivées secondes des cubiques s_{i-1} et s_i sont égales :

$$s''_{i-1}(x_i) = s''_i(x_i) \quad (1 \leq i \leq n-1) \quad \text{ **$n-1$ équations**}$$

Conditions sur les coefficients

Il y a $n + 1$ nœuds et n cubiques.

Chaque cubique dépend de 4 coefficients à déterminer :

$$s_i(z) = a_i + b_i (z - x_i) + c_i (z - x_i)^2 + d_i (z - x_i)^3.$$

Il y a donc $4 n$ inconnues. On cherche $4 n$ équations.

Au total

	$2 n$	(raccords C^0)
+	$n - 1$	(raccords C^1)
+	$n - 1$	(raccords C^2)
=	$4 n - 2$	équations

Il reste 2 équations qu'on peut choisir librement.

Conditions sur les coefficients

Il y a $n + 1$ nœuds et n cubiques.

Chaque cubique dépend de 4 coefficients à déterminer :

$$s_i(z) = a_i + b_i (z - x_i) + c_i (z - x_i)^2 + d_i (z - x_i)^3.$$

Il y a donc $4n$ inconnues. On cherche $4n$ équations.

Aux $4n - 2$ équations obligatoires pour une spline on peut rajouter

$$\left. \begin{array}{l} s_0''(x_0) = 0 \\ s_{n-1}''(x_n) = 0 \end{array} \right\} \quad (\text{la spline est dite naturelle})$$

Conditions sur les coefficients

Il y a $n + 1$ nœuds et n cubiques.

Chaque cubique dépend de 4 coefficients à déterminer :

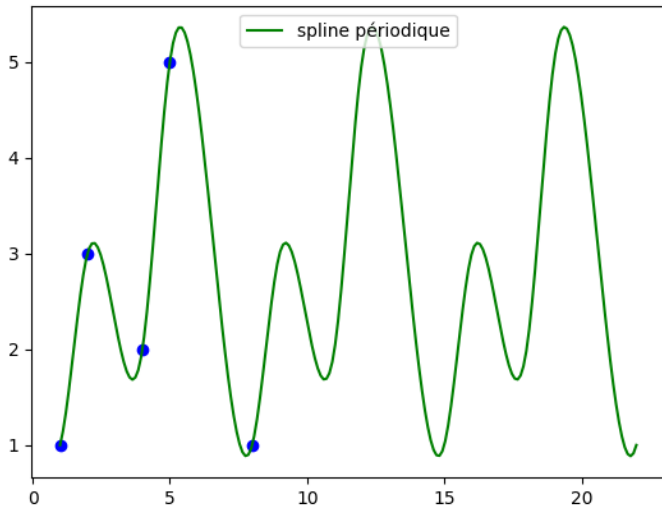
$$s_i(z) = a_i + b_i (z - x_i) + c_i (z - x_i)^2 + d_i (z - x_i)^3.$$

Il y a donc $4n$ inconnues. On cherche $4n$ équations.

Aux $4n - 2$ équations obligatoires pour une spline on peut rajouter, sous réserve que $y_0 = y_n$,

$$\left. \begin{array}{l} s'_0(x_0) = s'_{n-1}(x_n) \\ s''_0(x_0) = s''_{n-1}(x_n) \end{array} \right\} \quad (\text{la spline est dite } \text{périodique})$$

Une spline périodique



Une spline lissante naturelle

