

# Introduction à l'infographie

CIR3 - 2023/2024



- 1 Infographie
- 2 CGAO
- 3 Courbes polynomiales
- 4 Courbes de Bézier
- 5 Courbes B-Splines
- 6 NURBS

# Qu'est-ce que l'infographie ?

*The Adventures of André and Wally B. (1984)*

## Les années 1970

**INFOGRAPHIE / Computer graphics**

Création d'images numériques assistée par ordinateur

# Introduction

Les années 1960-1990

CAO = Conception Assistée par Ordinateur  
(CAD = Computer Aided Design)

Idée : décrire la forme d'un objet pour le représenter sur l'ordinateur

# Introduction

1<sup>er</sup> travaux :

**1959** *Pierre Bézier* - Renault

**1961** *Paul De Casteljau* - Citroen

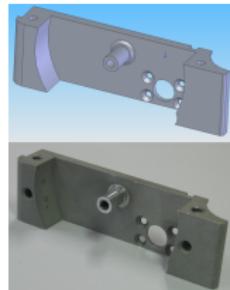
Commercialisation avec la naissance des langages de programmation  
(C, Pascal ...)

**1977** *CATIA* (Computer Aided Three dimensional Interactive Application) - IBM, Dassault Systèmes

*Topsolid* - Missler Software

**1982** *AutoCAD* - Autodesk

**1993** *SolidWorks* - Dassault Systèmes



# Introduction

1974 (Barnhill et Riesenfeld)

CGAO = Conception Géométrique Assistée par Ordinateur  
(CAGD = Computer Aided Geometric Design)

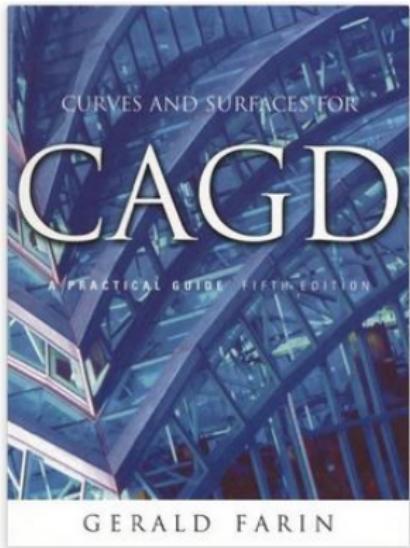
## Tâches de la CGAO :

- représentation
  - approximation de
  - construction
- ① courbes      ② surfaces      sur ordinateur

## Domaines d'utilisation :

Industrie, automobile, aéronautique, construction mécanique, milieu médical, robotique, cinéma ...

# Bibliographie



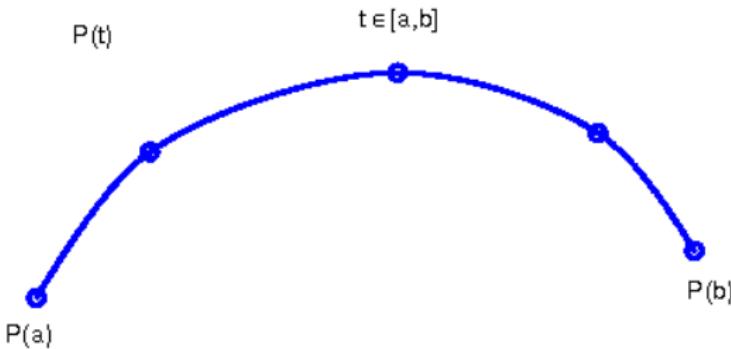
***Curves and surfaces for CAGD***  
***A practical guide***

Gerald Farin

# Représentation paramétrique : description

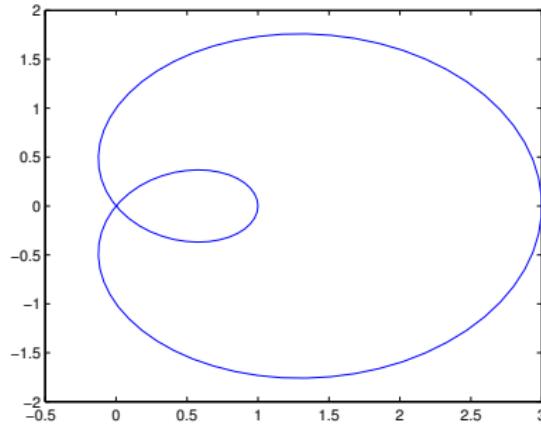
Courbe paramétrique :

$$\mathbf{P}(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix} ; \quad t \in [a, b] \in \mathbb{R}$$



## Représentation paramétrique : exemple

$$\mathbf{P}(t) = \begin{pmatrix} \cos(t)(2\cos(t) - 1) \\ \sin(t)(2\cos(t) - 1) \end{pmatrix} ; \quad t \in [0, 2\pi]$$



# Représentation paramétrique : pourquoi ?

Paramétrisation n'est pas unique (changement de paramètre) :

- propriétés intrinsèques liées à la géométrie (tangente, longueur d'arc)
- propriétés liée à la paramétrisation (continuité "paramétrique")

Pour aider les dessinateurs :

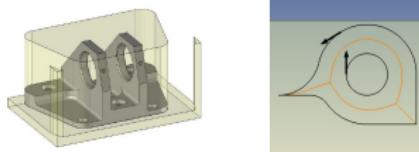
Système de modélisation :

intuitif, flexible, efficient, stable

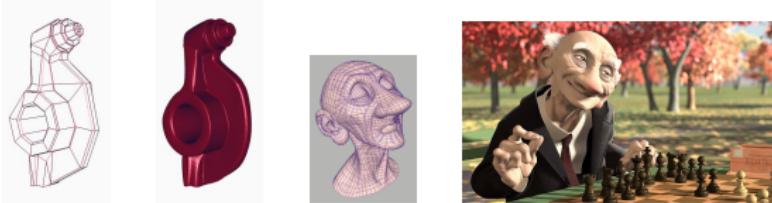
# Courbes paramétriques

Les plus connues et utilisées :

- **mesh polygonale** → objets pas lissés, rugueux
- **spline/NURBS** → Design industriel et trajectoires des machines outils  
(*C(F)AO, Conception (et Fabrication) Assistée par Ordinateur*)



- **courbes/surfaces de subdivision** → Animation 3D

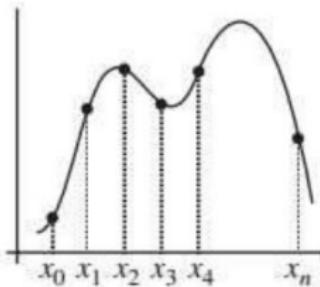


# Problème

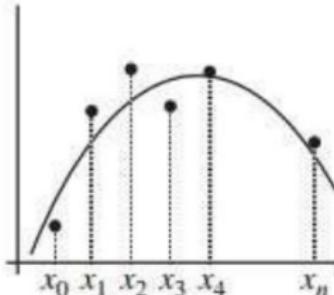
Soient :

$x_0, \dots, x_n$  points

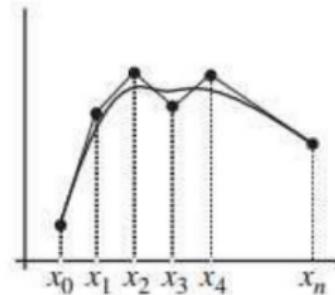
} Polynôme  $P(x)$  de degré  $n$   
tel que  $P(x_i) = y_i, i = 0, \dots, n$



Interpolation



Approximation



Modélisation

## Définition

Polynôme d'interpolation de degré  $n$  :

$$\mathbf{P}(x) = \sum_{i=0}^n a_i Q_i(x)$$

où

- $Q_i = \{\alpha_0(t), \dots, \alpha_n(t)\}$  base de l'espace vectoriel des polynômes de degré  $\leq n$
- $a_i$  coefficients réels

# Caractéristiques de la base

Objectif :

$Q_i$  donnés  $\Rightarrow$  déterminer  $a_i$ , pour  $i = 0, \dots, n$

Caractéristiques de la base polynomiale  $Q_i$ :

- interprétation géométrique des coefficients
- algorithme stable et rapide

# Différents choix des bases

## INTERPOLATION

- Base canonique (*Polynôme interpolant*)
- Base de Lagrange (*Polynôme de Lagrange*)
- Base de Newton (*Polynôme de Newton*)

## Interpolation par morceaux

- Linéaire
- Polynôme cubique d'Hermite
- Splines

## APPROXIMATION

- La méthode de moindres carrés

## MODÉLISATION

- Base de Bernstein (*Courbes de Béziers*)
- Modélisation par morceaux
- Base B-spline (*Courbes B-splines*)
  - Base B-spline rationnelle (*NURBS*)

## Base canonique

$$Q_i = \{1, x, x^2, \dots, x^n\} \Rightarrow P(x) = a_0 + a_1x + \dots + a_nx^n$$

## *Polynôme interpolant* de degré $\leq n$

Système linéaire de  $(n + 1)$  équations à  $(n + 1)$  inconnues

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

M = matrice de Vandermonde      inconnues

## Base canonique

Si les  $x_i$  sont distincts deux à deux  $\Rightarrow \det(M) \neq 0$

$\Rightarrow \exists!$  polynôme de degré  $\leq n$

Inconvénients de la base canonique :

- interprétation géométrique des coefficients inexistant
- complexité algorithmique élevé (résolution du système coûteuse)
- instabilité numérique de la matrice de Vandermonde

# Base de Lagrange

Objectif :

$$a_i \text{ donnés} \Rightarrow \text{déterminer } Q_i, \text{ pour } i = 0, \dots, n$$

On considère :

- coefficients  $a_i \rightsquigarrow y_i$
- base polynomiale  $Q_i \rightsquigarrow L_i = \{L_0, L_1, \dots, L_n\}$  **Base de Lagrange**

## Base de Lagrange

Le polynôme passe par les  $n + 1$  points.

On impose la condition :

$$L_i(x_j) = \begin{cases} 1 & \text{si } i=j \\ 0 & \text{sinon} \end{cases} \quad \text{pour } 0 \leq i, j \leq n$$

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

$$\Rightarrow P(x) = \sum_{i=0}^n y_i L_i(x)$$

*Polynôme de Lagrange*

## Base de Lagrange

Inconvénients de la base de Lagrange :

- pas efficace du point de vue pratique : si on ajoute un point d'interpolation (re-calcule de  $n + 2$  polynômes)
- Phénomène de Runge : augmentation du nombre de points  $\Rightarrow$  forte oscillation du polynôme

## Base de Newton

Soit  $\{\omega_0, \omega_1, \dots, \omega_n\}$  une famille de polynômes, où

$$\omega_0(x) = 1,$$

$$\omega_k(x) = \prod_{i=0}^{k-1} (x - x_i) = (x - x_{k-1})\omega_{k-1}(x) \quad \forall k = 1, \dots, n$$

$$\Rightarrow P(x) = \sum_{i=0}^n \alpha_i \omega_i(x)$$

*Polynôme de Newton*

## Base de Newton

Comment on calcule les  $\alpha_i$  ?

$$P(x_0) = \sum_{i=0}^n \alpha_i \omega_i(x_0) = \alpha_0 \Rightarrow \alpha_0 = y_0$$

$$P(x_1) = \sum_{i=0}^n \alpha_i \omega_i(x_1) = \alpha_0 + \alpha_1(x_1 - x_0) \Rightarrow \alpha_1 = \frac{y_1 - y_0}{x_1 - x_0}$$

$$P(x_2) = \sum_{i=0}^n \alpha_i \omega_i(x_2) = \alpha_0 + \alpha_1(x_2 - x_0) + \alpha_2(x_2 - x_0)(x_2 - x_1)$$

$$\Rightarrow \alpha_2 = \frac{y_2 - \alpha_0 - \alpha_1(x_2 - x_0)}{(x_1 - x_0)} = \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}$$

## Base de Newton

### Différences divisées

Soit  $\{(x_i, y_i)\}_{i=0}^n = \mathbf{x}_0, \dots, \mathbf{x}_n$  un ensemble de  $n + 1$  points distincts.

- La différence divisée d'ordre 1 de  $x_{i-1}$  et  $x_i$  est

$$f[x_{i-1}, x_i] = \frac{y_i - y_{i-1}}{x_i - x_{i-1}}$$

- La différence divisée d'ordre  $n$  des  $n + 1$  points  $x_0, \dots, x_n$  est définie par récurrence :

$$f[x_0, \dots, x_n] = \frac{f[x_1, \dots, x_n] - f[x_0, \dots, x_{n-1}]}{x_n - x_0}$$

# Base de Newton

Pour expliciter le processus récursif :

$i$	$x_i$	$y_i$	$f[x_{i-1}, x_i]$	$f[x_{i-2}, x_{i-1}, x_i]$	$f[x_{i-3}, x_{i-2}, x_{i-1}, x_i]$	$f[x_{i-4}, x_{i-3}, x_{i-2}, x_{i-1}, x_i]$	$\dots$
0	$x_0$	$y_0$					
1	$x_1$	$y_1$	$f[x_0, x_1]$				
2	$x_2$	$y_2$	$f[x_1, x_2]$	$f[x_0, x_1, x_2]$			
3	$x_3$	$y_3$	$f[x_2, x_3]$	$f[x_1, x_2, x_3]$	$f[x_0, x_1, x_2, x_3]$		
4	$x_4$	$y_4$	$f[x_3, x_4]$	$f[x_2, x_3, x_4]$	$f[x_1, x_2, x_3, x_4]$	$f[x_0, x_1, x_2, x_3, x_4]$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$

$$\Rightarrow P(x) = \sum_{i=0}^n \omega_i f[x_0, \dots, x_i]$$

*Polynôme de Newton*

## Inconvénients de l'interpolation polynomiale

En général :

- avec beaucoup de données  $\Rightarrow$  problème d'oscillation
- problème mal conditionné (si on change de peu les données, les coefficients du polynôme varient beaucoup)
- Si on diminue la distance entre les *noeuds*  $x_i$  (en augmentant leur nombre) on obtient des courbes qui ne convergent pas uniformément vers  $f$

$\Rightarrow$  SOLUTION : **Interpolation par morceaux ou par SPLINES**

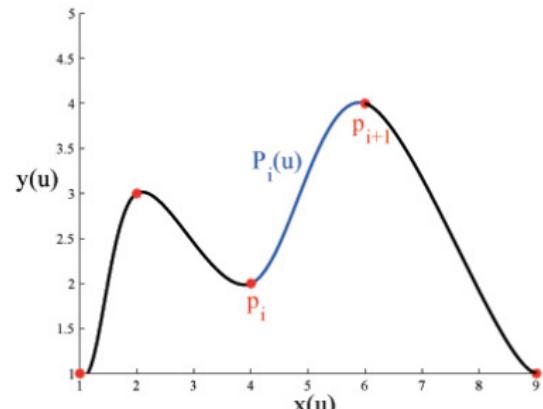
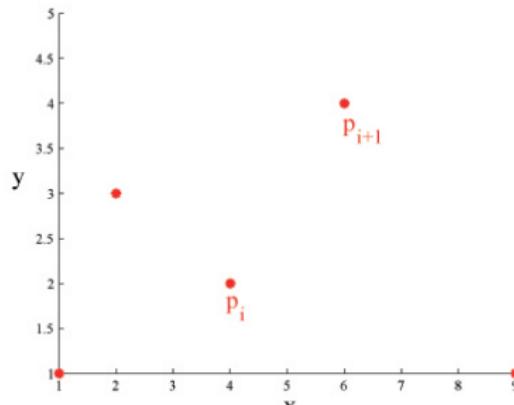
## Problème

Soient :

- $p_1, \dots, p_n$  points
- $t_1, \dots, t_n$

$$t \in [t_i, t_{i+1}] \text{ et } u \in [0, 1] \implies u = \frac{t - t_i}{t_{i+1} - t_i}$$

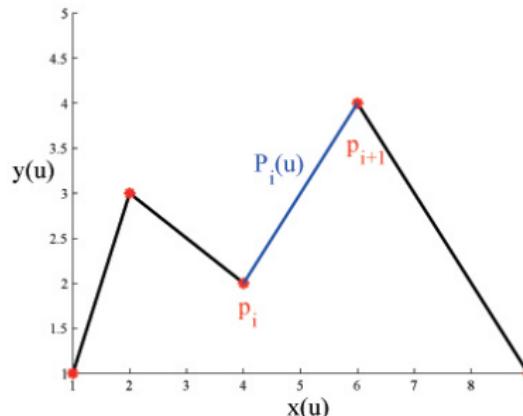
}  $n - 1$  courbes  $P_i(u)$   
qui interpolent  $p_i$  et  $p_{i+1}$



## Différentes techniques

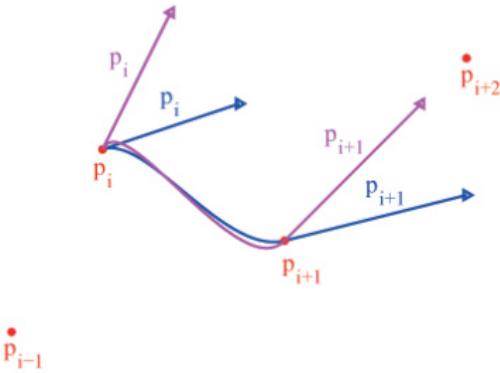
- *Interpolation linéaire* (courbe de continuité  $\mathcal{C}^0$  si  $\mathbf{P}_i(1) = \mathbf{P}_{i+1}(0)$ )

$$\mathbf{P}_i(u) = [ \begin{array}{cc} u & 1 \end{array} ] \cdot [ \begin{array}{cc} -1 & 1 \\ 1 & 0 \end{array} ] \cdot [ \begin{array}{c} \mathbf{p}_i \\ \mathbf{p}_{i+1} \end{array} ]$$



- *Courbes cubiques d'Hermite* (courbe de continuité  $\mathcal{C}^1$  si  $\mathcal{C}^0$  et  $\mathbf{P}'_i(1) = \mathbf{P}'_{i+1}(0)$ )

$$\mathbf{P}_i(u) = [ \begin{array}{cccc} u^3 & u^2 & u & 1 \end{array} ] \cdot \left[ \begin{array}{cccc} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{array} \right] \cdot \left[ \begin{array}{c} \mathbf{p}_i \\ \mathbf{p}_{i+1} \\ \mathbf{p}'_i \\ \mathbf{p}'_{i+1} \end{array} \right]$$



### ● Catmull-Rom:

$$\mathbf{p}'_i = \frac{1}{2}(\mathbf{p}_{i+1} - \mathbf{p}_{i-1}) \text{ et } \mathbf{p}'_{i+1} = \frac{1}{2}(\mathbf{p}_{i+2} - \mathbf{p}_i)$$

### ● TCB curve:

$$\mathbf{p}'_i = \frac{(1-\tau_i)(1+c_i)(1+b_i)}{2} \cdot (\mathbf{p}_i - \mathbf{p}_{i-1}) + \frac{(1-\tau_i)(1-c_i)(1-b_i)}{2} (\mathbf{p}_{i+1} - \mathbf{p}_i)$$

$$\mathbf{p}'_{i+1} = \frac{(1-\tau_{i+1})(1-c_{i+1})(1+b_{i+1})}{2} \cdot (\mathbf{p}_{i+1} - \mathbf{p}_i) + \frac{(1-\tau_{i+1})(1+c_{i+1})(1-b_{i+1})}{2} (\mathbf{p}_{i+2} - \mathbf{p}_{i+1})$$

## EXERCICES

- ① On considère  $x, y \in \mathbb{R}^4$  donnés par :  $x = [-2, 0, 1, 2]$  et  $y = [4, 0, 0, 4]$ .  
 Parmi les polynômes suivants, lequel est le polynôme d'interpolation  $P$  aux points  $x, y$ ? Justifiez votre réponse.

$$\begin{aligned}
 ① \quad P_1(X) &= X^4 - \frac{2}{3}X^3 - 3X^2 + \frac{8}{3} \\
 ② \quad P_2(X) &= \frac{4}{3}X^2 - \frac{4}{3} \\
 ③ \quad P_3(X) &= \frac{1}{3}X^3 + X^2 - \frac{4}{3}X
 \end{aligned}$$

- ② Calculer, si il existe, le polynôme  $P_3$  de  $\mathbb{R}_3[X]$  tel que :

$$P(-1) = 1 \quad P'(-1) = 1 \quad P'(1) = 2 \quad P(2) = 1$$

③ Calculer le polynôme d'interpolation de la fonction

$$f(x) = \sin(x) \text{ en } x_i = \frac{\pi}{2} i \text{ avec } i = 0, 1, 2$$

avec les trois bases : canonique, Lagrange et Newton.

On cherche le polynôme  $P_2$  de degré 2 tel que  $P_2(x_i) = \sin(x_i)$  pour  $i = 0, 1, 2$ .

Et si on veut ajouter un point  $x = \frac{3\pi}{2}$ , quelle méthode est préférable utiliser ?

Pourquoi ?

- ④
- ① Déterminer l'expression de la base de Newton de  $\mathbb{R}_5[X]$  associée aux nœuds 1,2,3,4,5,6.
  - ② Montrer qu'il s'agit bien d'une base.
  - ③ Donner, dans cette base, l'expression du polynôme  $P \in \mathbb{R}_5[X]$  tel que

$$P(1) = P(6) = 1, \quad P(2) = P(3) = P(4) = P(5) = 0$$

## Introduction : courbes composites

Courbes simples de degré 2 ou 3 avec raccord de jonction :

- *continuité de classe  $\mathcal{G}^1$*  (continuité géométrique)

Au raccord, les vecteurs dérivée ont la même direction mais pas forcement la même norme. La vitesse peut changer. On a la condition :  $\overrightarrow{\mathbf{p}_n \mathbf{p}_{n+1}} = k \overrightarrow{\mathbf{q}_0 \mathbf{q}_1}$

- *continuité de classe  $\mathcal{C}^1$*

Au raccord, la direction et la norme du vecteur dérivée sont continues. Il n'y a pas changement de vitesse. On a la condition :  $\overrightarrow{\mathbf{p}_n \mathbf{p}_{n+1}} = \overrightarrow{\mathbf{q}_0 \mathbf{q}_1}$

## Introduction : vers une spline

Idée :

- on utilise plusieurs polynômes de bas degré  $m$  sur chaque intervalle  $[t_i, t_{i+1})$  ;
- on recolle ces polynômes pour définir une fonction  $P$  sur l'intervalle tout entier ;
- on ajoute des conditions de régularité  $\vec{u}$  (continuité, dérivabilité, ...) à la fonction  $P$ .

Souci : augmentation degré des polynômes  $\Rightarrow$  Trouver bon compromis

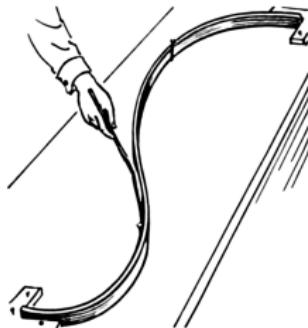
$\Rightarrow P_i^m \in [t_i, t_{i+1})$  ESPACE DES POLYNOMES PAR MORCEAUX

Inconvénient : espace avec une dimension trop lourde!

On fixe  $\vec{t}, \vec{u}, m (= k - 1)$   $\Rightarrow$  "dimension espace =  $n \cdot k - \vec{u}$ "

## Définition

"Le mot anglais *spline* a malencontreusement prévalu, alors qu'il se traduit en français par les substantifs *latte* ou *baguette*." (P.Bézier)



*Cerce* : règle ou latte plate, permettant aux dessinateurs de plans de tracer des courbes.

Cette latte est contrainte en flexion pour adopter la forme voulue.

## Définition

*The Oxford English Dictionary* 1989 donne les définitions suivantes pour le mot *spline* :

"A long narrow, and relatively thin piece or strip of wood, metal, etc.; a slat. A flexible strip of wood or hard rubber used by draftsmen in laying out broad sweeping curves, especially in railroad work."



Le terme spline vient de la construction navale: longues, fines et flexibles bandes de bois, de plastique ou métallique. Elles étaient maintenues en place avec des poids en plomb ("ducks") servant de points de contrôle pour la courbe.

## Espace des splines polynomiales

L' **espace des splines polynomiales** est formé par des polynômes par morceaux qui se raccordent avec la continuité la plus élevée :

$$S_{\vec{t},k} \in C^{k-2}, \text{ où } k = m + 1 \text{ est l'ordre}$$

⇒ Bas degré + contrôle locale

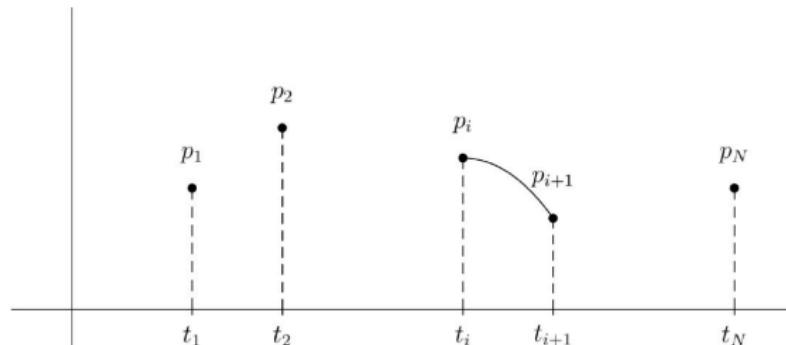
## Spline cubique

L'interpolation de  $N$  points donnés

$$P_1(t_1, p_1), P_2(t_2, p_2), \dots, P_N(t_N, p_N)$$

est réalisée par une courbe polynomiale par morceaux de degré 3.

La continuité  $\mathcal{C}^2$  est imposée.



## Spline cubique

- Sur chaque segment  $[t_i, t_{i+1}]$  la fonction spline correspond à un polynôme  $f_{i,i+1}$  de degré 3.
- En chaque  $t_i$  pour  $i = 2 \dots N - 1$

$$\begin{aligned}
 f_{i-1,i}(t_i) &= f_{i,i+1}(t_i) \\
 f'_{i-1,i}(t_i) &= f'_{i,i+1}(t_i) \\
 f''_{i-1,i}(t_i) &= f''_{i,i+1}(t_i) = k_i
 \end{aligned}$$

- Aux extrémités  $t_1$  et  $t_N$

$$\begin{aligned}
 f''_{1,2}(t_1) &= 0 = k_1 \\
 f''_{N-1,N}(t_N) &= 0 = k_N
 \end{aligned}$$

## Spline cubique

Pour  $i = 1 \dots N - 1$  le polynôme  $f''_{i,i+1}$  est de degré 1

$$f''_{i,i+1}(t) = k_i \frac{t - t_{i+1}}{t_i - t_{i+1}} + k_{i+1} \frac{t - t_i}{t_{i+1} - t_i}$$

En prenant une primitive :

$$f'_{i,i+1}(t) = \frac{k_i}{t_i - t_{i+1}} \frac{(t - t_{i+1})^2}{2} + \frac{k_{i+1}}{t_{i+1} - t_i} \frac{(t - t_i)^2}{2} + (\text{Constante})_i$$

Et en intégrant à nouveau :

$$f_{i,i+1}(t) = \frac{k_i}{t_i - t_{i+1}} \frac{(t - t_{i+1})^3}{6} + \frac{k_{i+1}}{t_{i+1} - t_i} \frac{(t - t_i)^3}{6} + (\text{PolynomeDegree1})_i$$

## Spline cubique

En écrivant le  $(PolynomeDegre1)_i$  dans la base  $\{t - t_{i+1}, t_i - t\}$

$$A_i(t - t_{i+1}) + B_i(t_i - t)$$

et en utilisant les conditions  $f_{i,i+1}(t_i) = p_i$  et  $f_{i,i+1}(t_{i+1}) = p_{i+1}$  :

$$A_i = \frac{p_i}{t_i - t_{i+1}} - \frac{k_i}{6}(t_i - t_{i+1}) \quad \text{et} \quad B_i = \frac{p_{i+1}}{t_i - t_{i+1}} - \frac{k_{i+1}}{6}(t_i - t_{i+1})$$

Soit, pour  $i = 1 \dots N - 1$

$$\begin{aligned} f_{i,i+1}(t) &= \frac{k_i}{6} \left( \frac{(t - t_{i+1})^3}{t_i - t_{i+1}} - (t_i - t_{i+1})(t - t_{i+1}) \right) + \frac{k_{i+1}}{6} \left( \frac{(t - t_i)^3}{t_{i+1} - t_i} - (t_i - t_{i+1})(t - t_i) \right) \\ &\quad + \frac{p_i(t - t_{i+1}) - p_{i+1}(t - t_i)}{t_i - t_{i+1}} \end{aligned}$$

## Spline cubique

L'étape suivante est d'utiliser les conditions de continuité des dérivées premières :

$$f'_{i-1,i}(t_i) = f'_{i,i+1}(t_i)$$

Pour  $i = 2 \dots N - 1$  on a donc les  $N - 2$  équations à  $N - 2$  inconnues  $k_2, \dots, k_{N-1}$  :

$$\frac{k_i}{3}(t_i - t_{i+1}) + \frac{k_{i+1}}{6}(t_i - t_{i+1}) + \frac{p_i - p_{i+1}}{t_i - t_{i+1}} = -\frac{k_{i-1}}{6}(t_{i-1} - t_i) + \frac{k_i}{3}(t_i - t_{i-1}) + \frac{p_{i-1} - p_i}{t_{i-1} - t_i}$$

## EXERCICE

Déterminer la spline cubique pour  $N = 3$ .

Combien d'inconnues nous avons ? Et de polynômes ?

Pour  $N = 3$  on a une seule inconnue  $k_2$  qui doit vérifier :

$$\frac{k_2}{3}(t_2 - t_3) + \frac{p_2 - p_3}{t_2 - t_3} = \frac{k_2}{3}(t_2 - t_1) + \frac{p_1 - p_2}{t_1 - t_2}$$

Soit :

$$k_2 = \frac{3}{t_1 - t_3} \left( \frac{p_1 - p_2}{t_1 - t_2} - \frac{p_2 - p_3}{t_2 - t_3} \right)$$

On a deux polynômes de degré 3 :

$$f_{1,2}(t) = \frac{k_2}{6} \left( \frac{(t - t_1)^3}{t_2 - t_1} - (t_1 - t_2)(t - t_1) \right) + \frac{p_1(t - t_2) - p_2(t - t_1)}{t_1 - t_2}$$

et

$$f_{2,3}(t) = \frac{k_2}{6} \left( \frac{(t - t_3)^3}{t_2 - t_3} - (t_2 - t_3)(t - t_3) \right) + \frac{p_2(t - t_3) - p_3(t - t_2)}{t_2 - t_3}$$

# Approximation

Objectif :

On cherche le polynôme  $P(x)$  de degré  $\leq n$  qui ne passe pas exactement par les points donnés mais il ne reste proche au sens *des moindres carrés*.

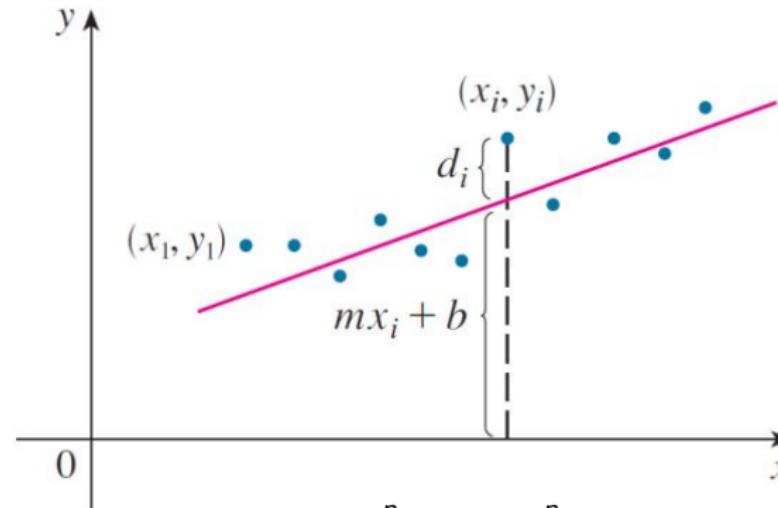
La quantité

$$(P(x_0) - y_0)^2 + (P(x_1) - y_1)^2 + \dots + (P(x_n) - y_n)^2$$

soit la plus petite possible.

$$\Rightarrow \exists! \text{ polynome de degré } \leq n$$

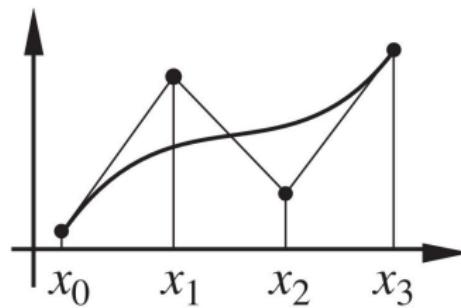
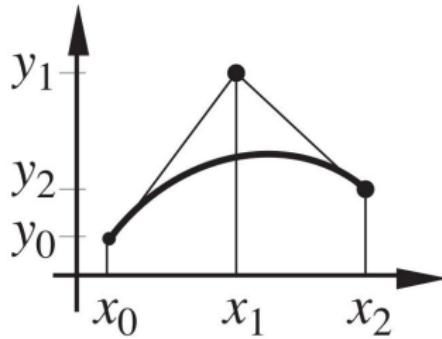
## La méthode des moindres carrés



$$\text{Minimiser : } g(m, q) = \sum_{i=0}^n d_i^2 = \sum_{i=0}^n (y_i - mx_i - q)^2$$

## Modélisation : Introduction

Trouver une courbe "douce" qui se trouve à l'intérieur d'un polygone de contrôle qui interpolate les données.



## Résumé

### Problématique :

Soient  $\mathbf{x}_1, \dots, \mathbf{x}_n$  points

$\left. \begin{array}{l} \text{Polynôme } P(x) \text{ de degré } n \\ \text{tel que } P(\mathbf{x}_i) = y_i, \quad i = 1, \dots, n \end{array} \right\}$

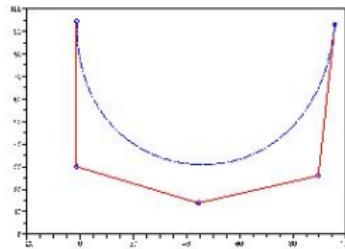
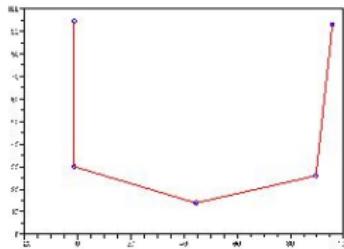
### 3 approches :

- *Interpolation*  $\Rightarrow$  polynôme (Matrice de Vandermonde, Lagrange, Newton) ou par morceaux (linéaire, cubique d'Hermite, splines)
- *Approximation*  $\Rightarrow$  moindres carrés
- *Modélisation*  $\Rightarrow$  courbes de Bézier et B-spline

## Historique

*Pierre Bézier* (1910-1999), ingénieur chez Renault

Idée : exprimer une courbe comme combinaison linéaire de points (points de contrôles)



## Historique

[1980] lettre stockée sous la forme d'une grille 8 x 8 (noire ou blanche, 0 ou 1).



[1988] John Warnock chez Adobe : courbe de Bézier à la base du système PostScript (vectorielle)



## Définition

### Courbe de Bézier

Soient  $\mathbf{p}_0, \dots, \mathbf{p}_n$  points (*de contrôle*), une **courbe de Bézier**  $\mathbf{P}(t)$  est définie comme

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

où  $B_i^n(t)$  sont les fonctions de base définies par les :

### Polynômes de Bernstein (de degré $n$ )

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}, \quad i = 0, \dots, n; \quad t \in [0, 1]$$

# Polynômes de Bernstein

Exemples :

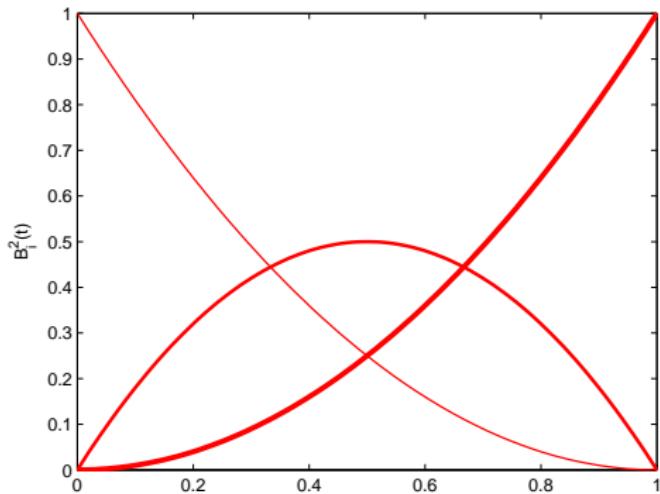
Si  $n = 1$  :  $B_0^1(t) = (1 - t)$ ,  $B_1^1(t) = t$

Si  $n = 2$  :

$$B_0^2(t) = (1 - t)^2$$

$$B_1^2(t) = 2t(1 - t)$$

$$B_2^2(t) = t^2$$



# Polynômes de Bernstein

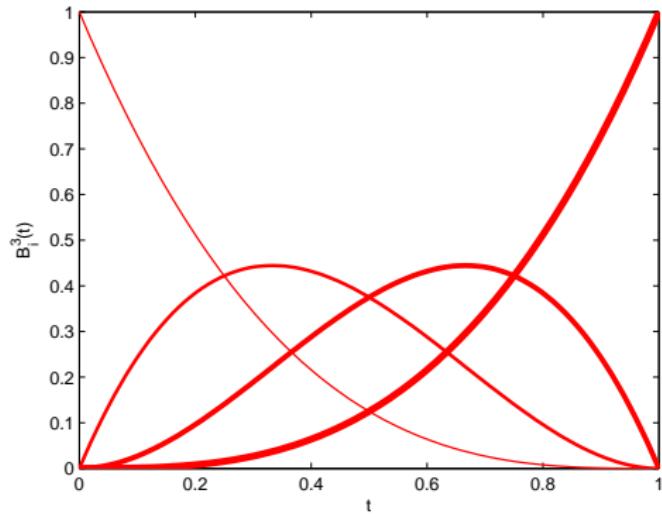
Si  $n = 3$

$$B_0^3(t) = (1 - t)^3$$

$$B_1^3(t) = 3t(1 - t)^2$$

$$B_2^3(t) = 3t^2(1 - t)$$

$$B_3^3(t) = t^3$$



## Propriétés des polynômes de Bernstein

- ❶ *Positivité* :  $B_i^n(t) \geq 0 \quad \forall i = 0, \dots, n$
- ❷ *Partition de l'unité* :  $\sum_{i=0}^n B_i^n(t) = 1, \quad \forall t \in [0, 1]$
- ❸ *Symétrie* :  $B_i^n(t) = B_{n-i}^n(1 - t)$
- ❹ *Formule de récurrence* :

$$\begin{aligned} B_i^n(t) &= (1 - t) B_i^{n-1}(t) + t B_{i-1}^{n-1}(t) \\ B_i^n(t) &= 0 \quad \text{pour } i < 0, \quad i > n \\ B_0^n(t) &= 1 \quad \text{pour } i < 0 \end{aligned}$$

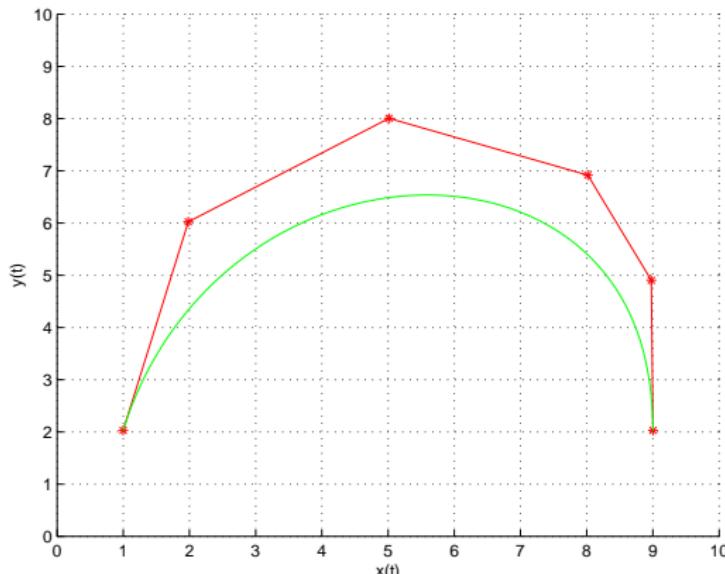
- ❺ *Dérivée* :

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

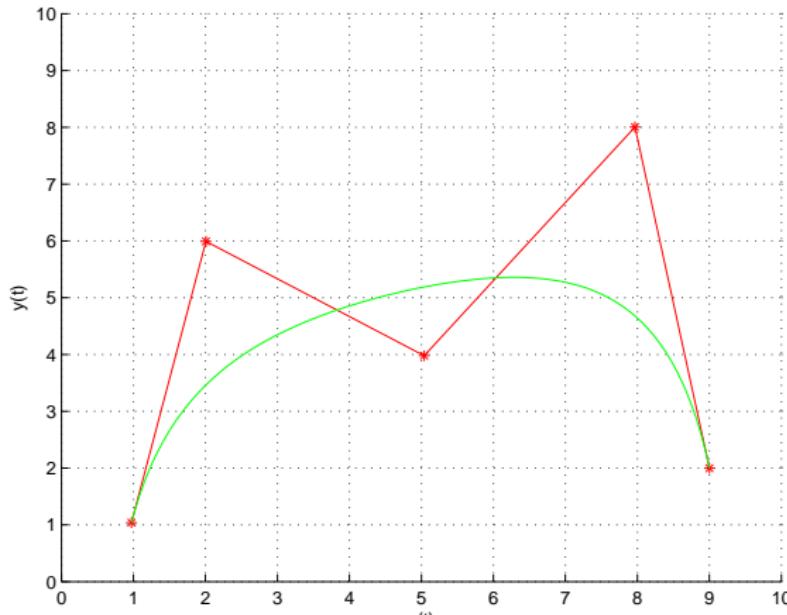
- ❻  $B_i^n(t)$  possède un *maximum* en  $t = \frac{i}{n}$

## Propriétés de la courbe de Bézier

Les points de contrôle  $p_0, \dots, p_n$  constituent le *polygone de contrôle*.



## Propriétés de la courbe de Bézier



## Propriétés de la courbe de Bézier

### ① *Interpolation du premier et dernier point de contrôle :*

- pour  $t = 0$ ,  $P(0) = \mathbf{p}_0$
- pour  $t = 1$ ,  $P(1) = \mathbf{p}_n$

### ② *Tangence au premier et dernier segment du polygone de contrôle:*

- pour  $t = 0$ ,  $P(t)$  est tangent à  $\overline{\mathbf{p}_0\mathbf{p}_1}$  :  $P'(0) = n(\mathbf{p}_1 - \mathbf{p}_0)$
- pour  $t = 1$ ,  $P(t)$  est tangent à  $\overline{\mathbf{p}_{n-1}\mathbf{p}_n}$  :  $P'(1) = n(\mathbf{p}_n - \mathbf{p}_{n-1})$

### ③ *Enveloppe convexe (Convexe hull):*

La courbe se trouve à l'intérieur de l'enveloppe convexe des points de contrôle. C'est le plus petit ensemble convexe parmi ceux qui contiennent les points de contrôle.

### ④ *Invariance par transformations affines*

## Propriétés de la courbe de Bézier

### 5 *Variation diminishing* :

Le nombre  $k$  de points d'intersection de la courbe  $P(t)$  avec une droite arbitraire est inférieur ou égal au nombre de points d'intersection  $m$  du polygone de contrôle avec la même droite ( $k \leq m$ ).

- Si le polygone de contrôle ne change pas sa concavité (convexe ou concave) alors la courbe ne change pas sa concavité.  
Attention : le vice-versa n'est pas garantie!  
Si le polygone de contrôle change de concavité, la courbe de Bézier peut être convexe.
- La courbe peut avoir au maximum le même nombre d'inflexions du polygone de contrôle.
- Si le polygone est monotone, alors la courbe est monotone.

## EXERCICES

- ➊ Démontrer les propriétés suivantes des polynômes de Bernstein :
  - Formule de récurrence :  $B_i^n(t) = (1 - t) B_i^{n-1}(t) + t B_{i-1}^{n-1}(t)$
  - Dérivée :  $(B_i^n(t))' = n (B_{i-1}^{n-1}(t) - B_i^{n-1}(t))$
- ➋ Soit  $\mathbf{P}(t) = (t, 2t - 2t^2)$  une courbe de Bézier.
  - ➌ Trouver ses points de contrôle.
  - ➍ Donner les points de contrôle de la courbe élevée au degré 3.

## Algorithme de De Casteljau

L'algorithme de Casteljau est un *algorithme récursif* trouvé par Paul de Casteljau pour approximer efficacement les polynômes écrit dans la base de Bernstein.

Il permet de calculer pour une valeur du paramètre  $t^*$  le correspondant point sur la courbe de Bézier, par des interpolations linéaires.

# Algorithme de De Casteljau

## Théorème

Soit  $(\mathbf{p}_i)_{0 \leq i \leq n}$  la suite des  $n + 1$  points de contrôle de  $\mathbf{P}(t)$  une courbe de Bézier.

Soit, pour tout  $t$ , la suite des points  $\mathbf{p}_j^k$ , avec  $0 < k \leq n$  et  $0 \leq j \leq n - k$ , définis par

$$\begin{aligned}\mathbf{p}_j^0 &= \mathbf{p}_j \\ \mathbf{p}_j^k &= (1 - t) \mathbf{p}_j^{k-1} + t \mathbf{p}_{j+1}^{k-1}.\end{aligned}$$

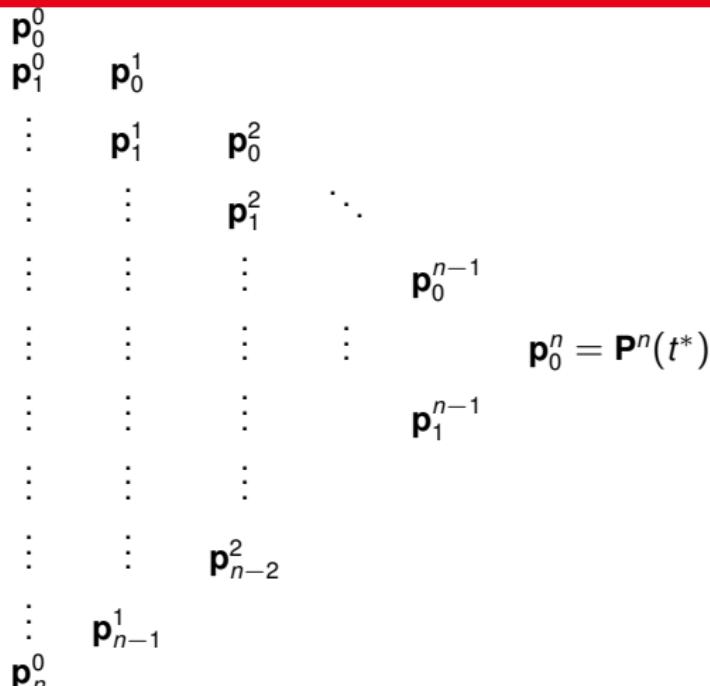
Alors on a pour tout  $t^*$  tel que  $t^* \in [0, 1]$ ,  $\mathbf{p}_0^n = \mathbf{P}(t^*)$ , où  $\mathbf{P}(t^*)$  est le point sur la courbe  $P(t)$  correspondant à  $t^*$ .

# Algorithme de De Casteljau

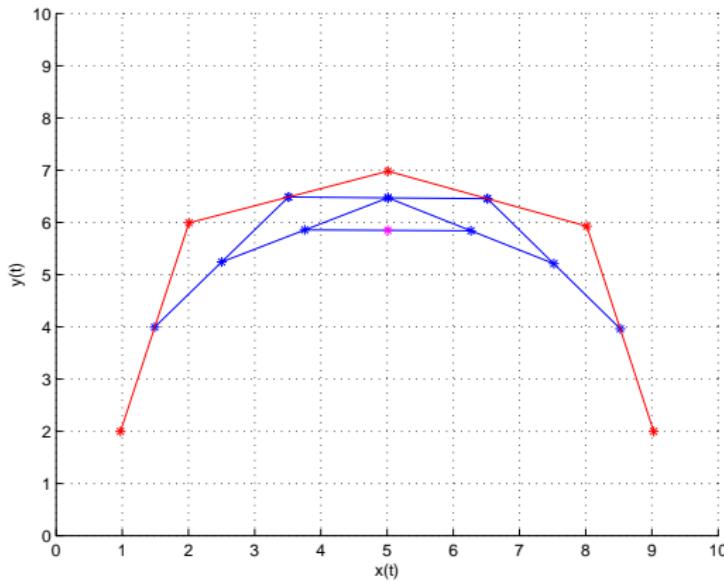
Proof.

$$\begin{aligned}
 \mathbf{p}_0^n &= P(t^*) = \sum_{i=0}^n \mathbf{p}_i^0 B_i^n(t^*), \quad \text{pour } t^* \in [0, 1] \\
 &= \sum_{i=0}^{n-1} \mathbf{p}_i^0 (1-t^*) B_i^{n-1}(t^*) + \sum_{i=1}^n \mathbf{p}_i^0 t^* B_{i-1}^{n-1}(t^*) \\
 &= \sum_{i=0}^{n-1} ((1-t^*) \mathbf{p}_i^0 + t^* \mathbf{p}_{i+1}^0) B_i^{n-1}(t^*) = \sum_{i=0}^{n-1} \mathbf{p}_i^1 B_i^{n-1}(t^*) \\
 &= \sum_{i=0}^{n-2} \mathbf{p}_i^1 (1-t^*) B_i^{n-2}(t^*) + \sum_{i=1}^{n-1} \mathbf{p}_i^1 t^* B_{i-1}^{n-2}(t^*) = \sum_{i=0}^{n-2} \mathbf{p}_i^2 B_i^{n-2}(t^*) \\
 \implies n \text{ fois} &= \sum_{i=0}^1 \mathbf{p}_i^{n-1} B_i^1(t^*) = \mathbf{p}_0^n B_0^0(t^*) = \mathbf{p}_0^n
 \end{aligned}$$

## Algorithme de De Casteljau



# Algorithme de De Casteljau



## Inconvénient : contrôle global

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

- **Méthode globale** : dès qu'on déplace un point de contrôle, la courbe de Bézier est totalement modifiée.
- Tous les points de contrôle participent à la construction de la courbe.

## Subdivision $\Rightarrow$ contrôle local

### Théorème

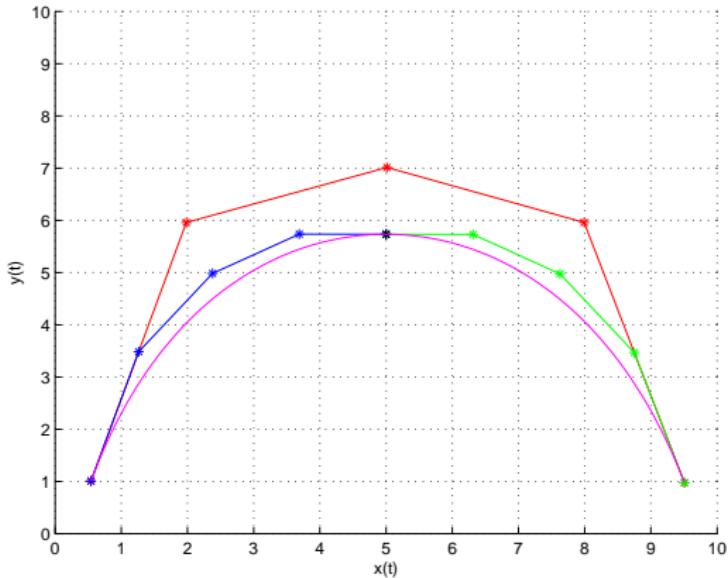
Soit  $\mathbf{P}^n(t)$  la courbe de Bézier défini par les points de contrôle  $(\mathbf{p}_i^0)_{0 \leq i \leq n}$  et soit  $t^* \in [0, 1]$ .

Si on considère le tableau de points de contrôle défini par l'algorithme de De Casteljau pour déterminer le point  $\mathbf{P}^n(t^*)$ , alors la courbe  $\mathbf{P}^n(t)$  est divisée par  $t^*$  en deux parties :

$$\mathbf{P}^n(t) = \begin{cases} \mathbf{P}_1^n\left(\frac{t}{t^*}\right) & \text{pour } t \in [0, t^*] \\ \mathbf{P}_2^n\left(\frac{t-t^*}{1-t^*}\right) & \text{pour } t \in [t^*, 1] \end{cases} \text{ où}$$

- $\mathbf{P}_1^n\left(\frac{t}{t^*}\right)$  est défini par  $(\mathbf{p}_0^0, \mathbf{p}_0^1, \dots, \mathbf{p}_0^n)$
- $\mathbf{P}_2^n\left(\frac{t-t^*}{1-t^*}\right)$  est défini par  $(\mathbf{p}_0^n, \mathbf{p}_1^{n-1}, \dots, \mathbf{p}_n^0)$

# Subdivision



## Remarques

- Plus de flexibilité VS souci de continuité (point en commun)
- Polygonale converge à la courbe

## Élévation du degré

Une courbe  $\mathbf{P}(t)$  de degré  $n$  est aussi une courbe de degré  $n + 1$ , particulière. Donc il existe  $n + 2$  points tels que

$$\forall t \quad \mathbf{P}(t) = \sum_{i=0}^n \mathbf{p}_i B_i^n(t) = \sum_{i=0}^{n+1} \mathbf{q}_i B_i^{n+1}(t)$$

En multipliant à gauche par  $(t + (1 - t))$ , nous obtenons

$$\sum_{i=0}^n \mathbf{p}_i \binom{n}{i} (t^i (1-t)^{n-i+1}) + t^{i+1} (1-t)^{n-i} = \sum_{i=0}^{n+1} \mathbf{q}_i \binom{n+1}{i} t^i (1-t)^{n-i+1}$$

en comparant les coefficients de  $t^i (1-t)^{n-i+1}$  de deux côtés, nous obtenons

$$\mathbf{q}_i \binom{n+1}{i} = \mathbf{p}_i \binom{n}{i} + \mathbf{p}_{i-1} \binom{n}{i-1}$$

ce qui est équivalent à

$$\mathbf{q}_0 = \mathbf{p}_0, \quad \mathbf{q}_{n+1} = \mathbf{p}_n, \quad \mathbf{q}_i = \frac{i}{n+1} \mathbf{p}_{i-1} + \left(1 - \frac{i}{n+1}\right) \mathbf{p}_i, \text{ pour } 1 \leq i \leq n$$

# Élévation de degré

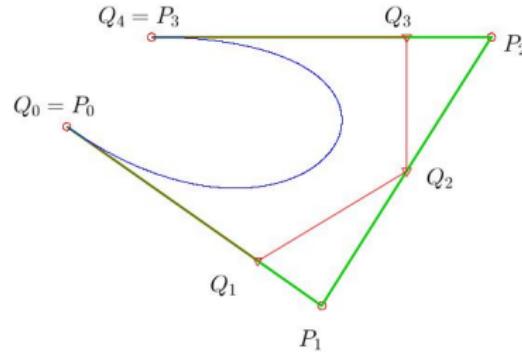


Figure: Élévation du degré 3 au degré 4

# EXERCICE

Soit  $C_1$  la courbe de Bézier définie par les points  $P_0(1, 1)$ ,  $P_1(4, 4)$ ,  $P_2(1, 4)$  et  $P_3(4, 1)$ .

- ➊ Tracer le polygone de contrôle de la courbe.
- ➋ Quel est le degré de cette courbe de Bézier ?
- ➌ Donner l'équation de la courbe de Bézier approximant ces points.
- ➍ Déterminer et placer les différents points calculés par l'algorithme de De Casteljau pour  $t = \frac{1}{3}$ .
- ➎ Dessiner l'allure de la courbe  $C_1$ .
- ➏ Dessiner l'allure de la courbe  $C_2$  définie par  $P_0$ ,  $P_2$ ,  $P_1$  et  $P_3$ .
- ➐ Dessiner l'allure de la courbe  $C_3$  définie par  $P_0$ ,  $P_2$ ,  $P_3$  et  $P_1$ .
- ➑ Donner l'équation de la courbe qui interpole les points  $P_0$  et  $P_3$  et  $T_0(9, 9)$  et  $T_3(9, -9)$  qui correspondent aux tangentes en ces deux points. Quel est le degré de ce polynôme ?  
*Suggestion :* Écrivez un polynôme générique et calculez les coefficients de ce polynôme utilisant les 4 contraintes connues (2 points extrêmes et 2 tangentes).
- ➒ Comment s'appelle le polynôme déterminé ?
- ➓ A-t-on la même courbe ? Pourquoi ?

## Définition

### Courbe de Bézier rationnelles

Soient  $\mathbf{p}_0, \dots, \mathbf{p}_n$  points de contrôle, une **courbe de Bézier rationnelle**  $\bar{\mathbf{P}}(t)$  est définie comme

$$\bar{\mathbf{P}}(t) = \sum_{i=0}^n \mathbf{p}_i R_i^n(t)$$

où  $R_i^n(t)$  sont les fonctions de base rationnelles définie par les :

### Polynômes de Bernstein rationnels (de degré $n$ )

$$R_i^n(t) = \frac{\omega_i B_i^n(t)}{\sum_{j=0}^n \omega_j B_j^n(t)}, \quad t \in [0, 1]$$

et  $\omega_i \in \mathbb{R}$  sont appelés **poids**.

# Propriétés

- Dénominateur  $\neq 0$  et  $\omega_j > 0$
- Même propriétés des polynômes de Bernstein (combinaison linéaire)
- Même propriétés d'une courbe de Bézier
- Paramètre de forme
- Courbes coniques (cercles, ellipses, hyperboles ...)

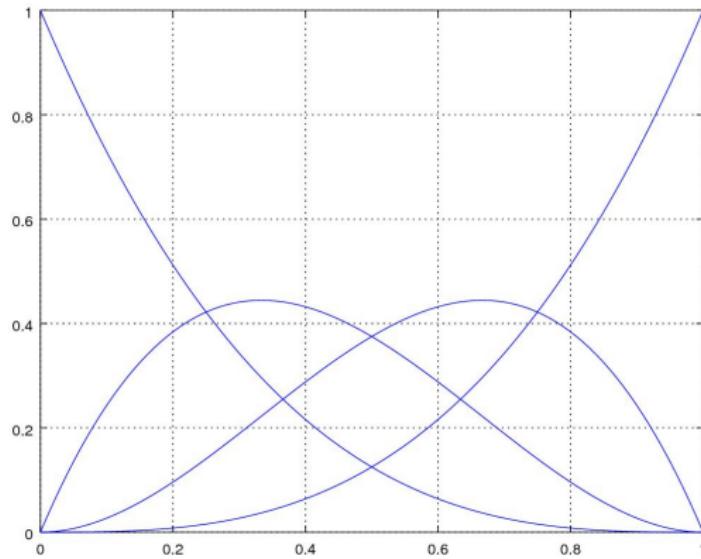
## Polynômes de Bernstein rationnelles avec n=3

$$\omega_0 = 1$$

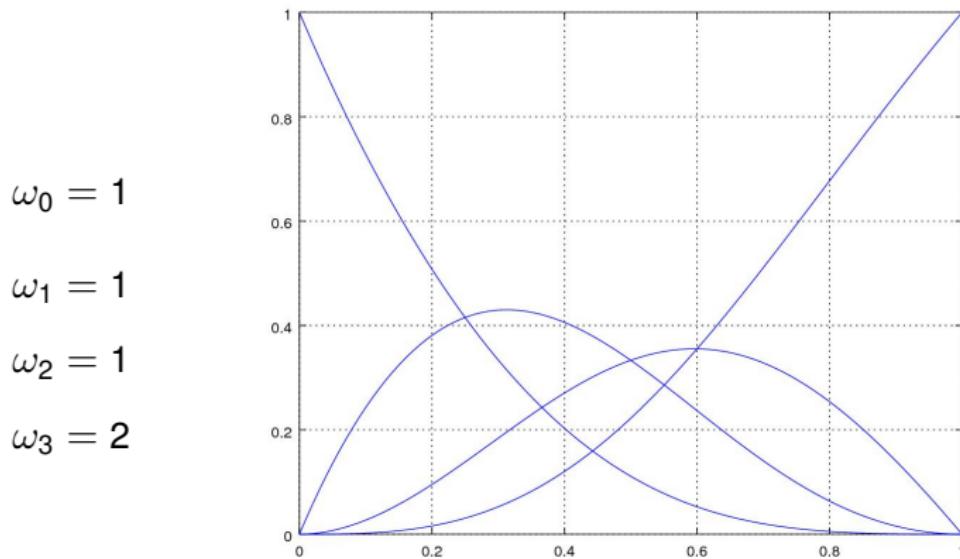
$$\omega_1 = 1$$

$$\omega_2 = 1$$

$$\omega_3 = 1$$



## Polynômes de Bernstein rationnelles avec n=3



## Courbes de Bézier rationnelles

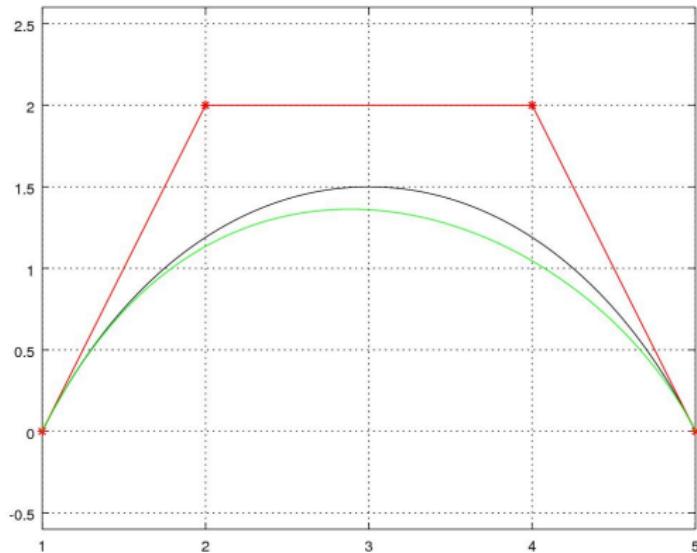
En noir avec

$$\omega_0 = \omega_1 = \omega_2 = \omega_3 = 1$$

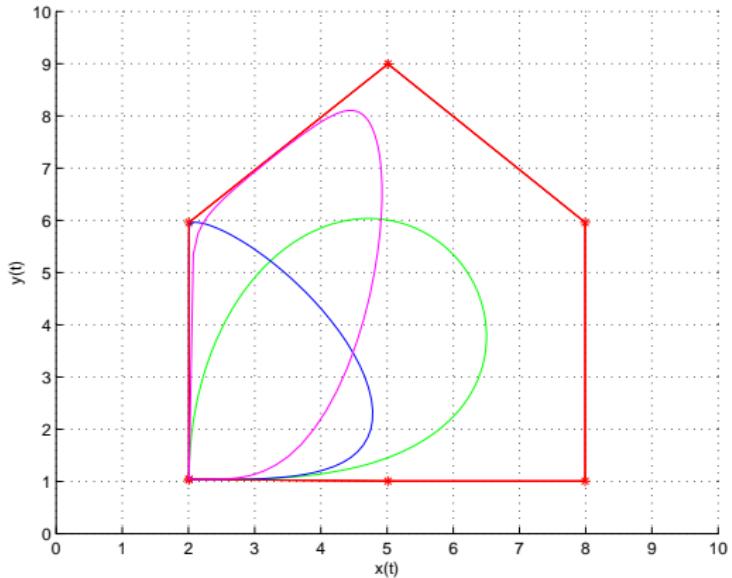
En vert avec

$$\omega_0 = \omega_1 = \omega_2 = 1$$

et  $\omega_3 = 2$



## Exemple



# EXERCICE

Soit  $s \in \mathbb{R}$ ,  $0 < s < 1$ , on souhaite modéliser l'arc de cercle suivant

$$C = \left\{ \begin{pmatrix} x \\ y \end{pmatrix}, -s \leq x \leq s, y = f(x) = \sqrt{1 - x^2} \right\}$$

par une courbe de Bézier rationnelle de degré 2, c'est-à-dire trouver les points  $\mathbf{p}_i$  et le poids  $\omega_1$  tels que :

$$C = \left\{ \mathbf{P}(t) = \frac{\sum_{i=0}^2 \omega_i \mathbf{p}_i B_i^2(t)}{\sum_{i=0}^2 \omega_i B_i^2(t)}, \quad t \in [0, 1] \right\} \text{ et } \omega_0 = \omega_2 = 1$$

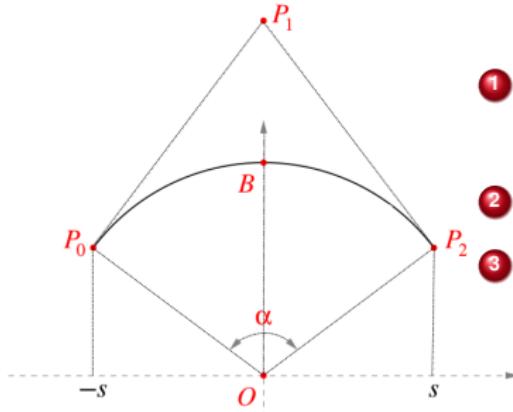
# EXERCICE

La figure ci-dessous représente l'arc de cercle  $C$  en gras, les trois points de contrôle  $\mathbf{p}_0$ ,  $\mathbf{p}_1$ ,  $\mathbf{p}_2$ ,

$$B = \mathbf{P}(1/2) = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \text{ milieu de l'arc de cercle et l'origine } O = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

L'ouverture de l'arc de cercle est l'angle  $\alpha$  strictement compris entre 0 et  $\pi$  avec :

$$\alpha = 2 \arcsin s \Leftrightarrow s = \sin\left(\frac{\alpha}{2}\right) \quad \text{et} \quad c = \sqrt{1 - s^2} = \cos\left(\frac{\alpha}{2}\right)$$



- ➊ Montrer que le point de contrôle  $\mathbf{p}_1$  est égal à  $(0, b)$  avec  $b = \frac{1}{c} = \frac{1}{\sqrt{1-s^2}}$ . Suggestion : écrire la droite tangente à la courbe  $C$  au point d'abscisse  $x_0$ .
- ➋ En déduire la valeur du poids  $\omega_1$ , en sachant que  $B_0^2(t) = (1-t)^2$ ,  $B_1^2(t) = 2t(1-t)$  et  $B_2^2(t) = t^2$ .
- ➌ Expliquer qu'on peut modéliser un cercle entier à l'aide de trois Bézier rationnelles de degré 2 mais qu'on ne peut pas modéliser un cercle entier uniquement avec deux Bézier rationnelles de degré 2.

# Introduction

Du point de vue mathématique, une courbe est définie par une polygonale qui lie un ensemble de points (points de contrôle) et par un schéma d'interpolation (ou approximation) qui associe à cette figure géométrique une équation.

⇒ Courbes de Bézier

## Avantages :

- propriétés des polynômes de Bernstein
- interprétation géométrique des coefficients
- algorithme stable et rapide

## Inconvénients :

- 1 degré de la courbe fixé par le nombre de points de contrôle
- 2 contrôle global de la courbe (polynômes de Bernstein définies globalement)

Système : intuitif, efficient, stable, mais pas flexible

## Vecteur nœud

Introduction d'un paramètre supplémentaire :

On appelle **vecteur nœud** toute suite finie de réels  $(t_0, \dots, t_n)$  tels que  $t_i \leq t_{i+1}$ .

Exemple :

- 0,1,2,3,4,5, suite d'entiers consécutifs
- 0,0,1,2,3,3, suite d'entiers croissante au sens large avec des valeurs nodales multiples
- 0.2,0.3,0.4,0.52,0.6, suite strictement croissante de réels

## Vecteur nœud

Le nœud  $t_i$  est **simple** si  $t_{i-1} < t_i < t_{i+1}$ , il est dit **multiple d'ordre  $p$**  si  $t_{i-1} < t_i = t_{i+2} = \dots = t_{i+p-1} < t_{i+p}$ .

Le vecteur nœud  $(t_0, \dots, t_n)$  est dit **non périodique** si la première valeur nodale et la dernière valeur nodale sont répétées  $p$  fois.

Lorsque les valeurs nodales non répétées sont également espacées, le vecteur nœud est dit **uniforme**.

Le vecteur nœud est dit **non uniforme** si les valeurs répétées ne sont pas régulièrement espacées.

## Fonctions B-splines

Étant donné un vecteur de nœuds  $(t_0, \dots, t_n)$ , on appelle **fonctions B-splines** associée à ce vecteur nœud les fonctions  $N_i^m$  pour  $0 \leq m \leq n - 1$  et  $0 \leq i \leq n - m - 1$  définies par la formule de récurrence :

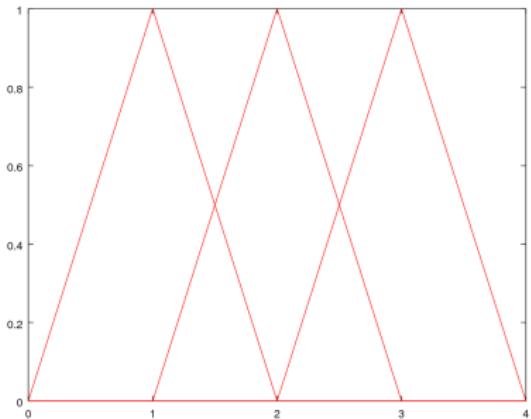
$$N_i^m(t) = \frac{t - t_i}{t_{i+m} - t_i} N_i^{m-1}(t) + \frac{t_{i+m+1} - t}{t_{i+m+1} - t_{i+1}} N_{i+1}^{m-1}(t)$$

avec

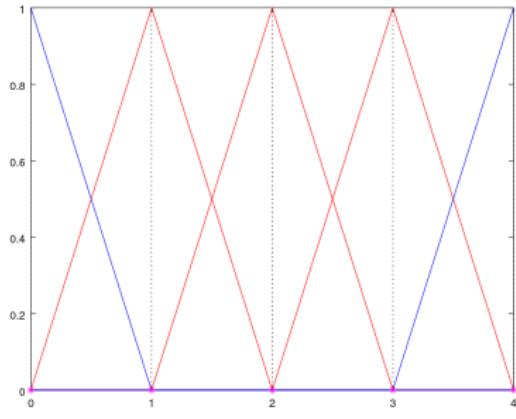
$$N_i^0(t) = \begin{cases} 1 & \text{si } t_i \leq t < t_{i+1} \\ 0 & \text{sinon} \end{cases}$$

où l'on convient si l'un des dénominateurs s'annule, d'annuler le terme correspondant.

## B-spline de degré 1 (ordre 2) avec $t = [0, 1, 2, 3, 4]$

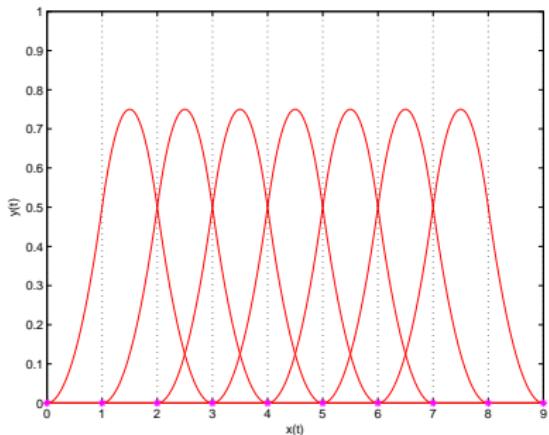


B-spline incomplète

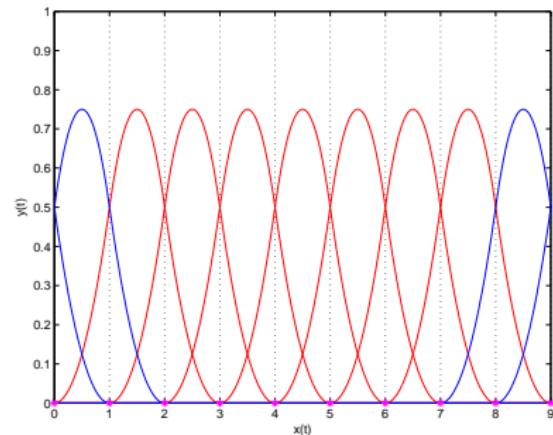


B-spline complète avec  
1 fonction à gauche et  
1 fonction à droite en plus

# B-spline de degré 2 ( $k=3$ ) avec $t = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]$



B-spline incomplète



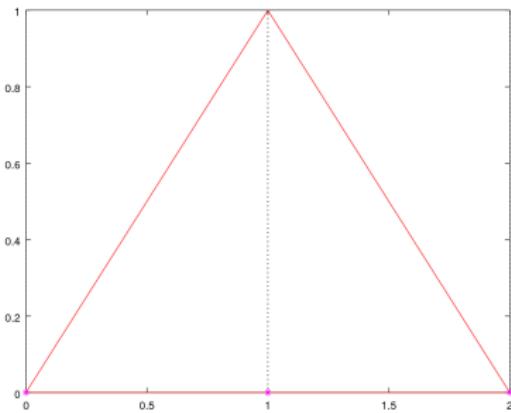
B-spline complète avec  
 $k - 1$  fonctions à gauche et  
 $k - 1$  fonctions à droite

## Propriétés des fonctions de base B-spline :

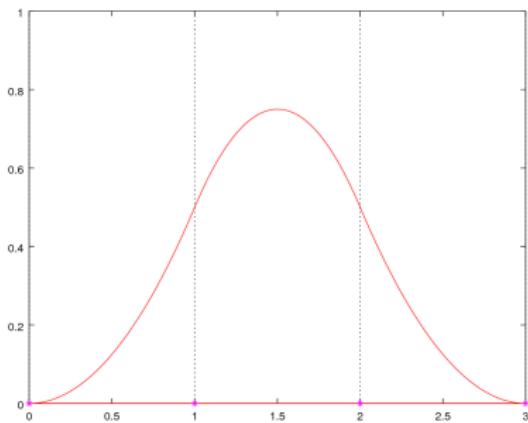
Chaque fonction  $N_i^m$  :

- couvre  $k$  intervalles
- est positive ou nulle :  $N_i^m \geq 0$
- partition de l'unité :  $\sum_{i=0}^n N_i^m(t) = 1$

# Recouvrement de $k$ intervalles

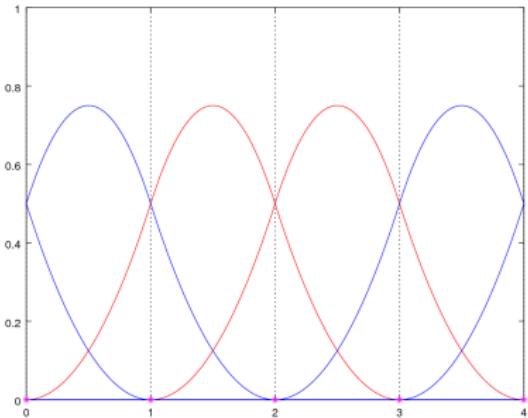


$k = 2$



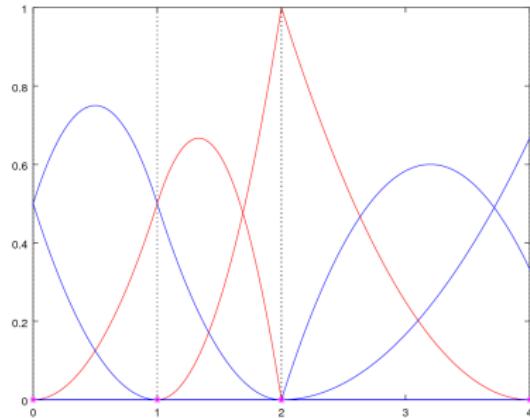
$k = 3$

# Nœuds multiples



$$k = 3$$

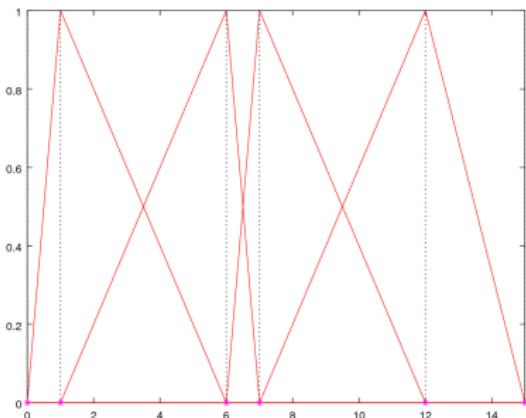
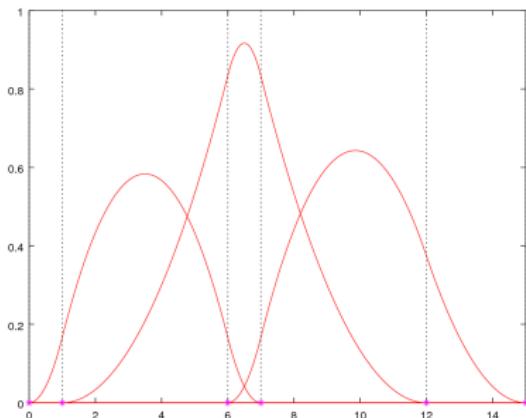
$$t = (0 \ 1 \ 2 \ 3 \ 4)$$



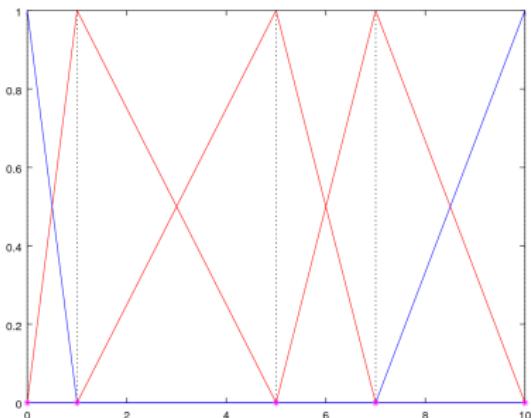
$$k = 3$$

$$t = (0 \ 1 \ 2 \ 2 \ 4)$$

# Nœuds non uniformes

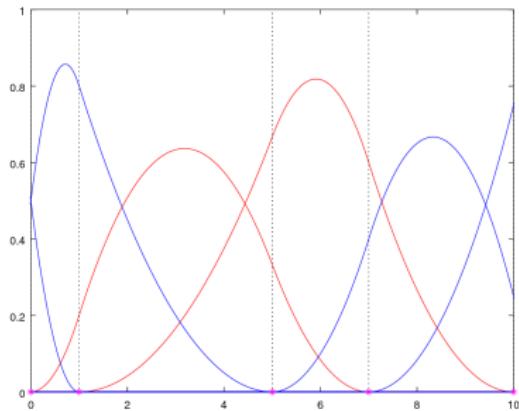
 $k = 2$  $t = (0 \ 1 \ 6 \ 7 \ 12 \ 15)$  $k = 3$  $t = (0 \ 1 \ 6 \ 7 \ 12 \ 15)$

## Nœuds non uniformes



$k = 2$

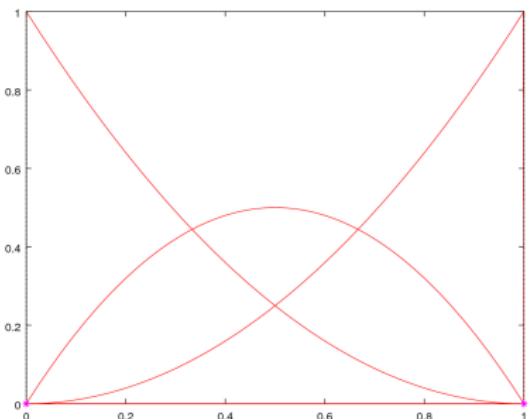
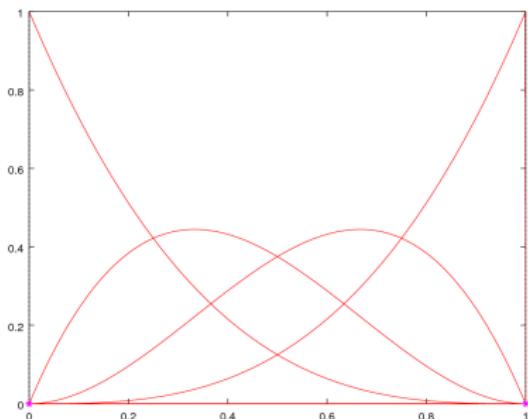
$t = (0 \ 1 \ 5 \ 7 \ 10)$



$k = 3$

$t = (0 \ 1 \ 5 \ 7 \ 10)$

# Nœuds non périodiques

 $k = 3$  $t = (0\ 0\ 0\ 1\ 1\ 1)$  $k = 4$  $t = (0\ 0\ 0\ 0\ 1\ 1\ 1\ 1)$

## Définition : courbe B-spline

Étant donné un vecteur de nœuds  $(t_0, \dots, t_{n+k})$ , une **courbe B-spline** de degré  $m$  ou d'ordre  $k (= m + 1)$  est une courbe paramétrique définie comme combinaison linéaire des points de contrôle  $\mathbf{p}_0, \dots, \mathbf{p}_n$  et des fonctions B-splines :

$$\mathbf{S}(t) = \sum_{i=0}^n \mathbf{p}_i N_i^m(t)$$

Remarque : La courbe de Bézier associée à  $n + 1$  points de contrôle est la courbe B-spline de degré  $n$  avec comme nœuds les points  $t_0 = t_1 = \dots = t_n = 0$  et  $t_{n+1} = t_{n+2} = \dots = t_{2n+1} = 1$

## EXERCICE

Soit une courbe B-spline définie par 5 points de contrôles  $\mathbf{p}_0(2, 2)$ ,  $\mathbf{p}_1(3, 4)$ ,  $\mathbf{p}_2(5, 1)$ ,  $\mathbf{p}_3(7, 3)$  et  $\mathbf{p}_4(9, 2)$  d'ordre 3.

- ➊ Calculer le vecteur de nœuds non périodique (nœuds de multiplicité  $k$  aux extrémités) de cette courbe.
- ➋ Calculer la fonction de base  $N_{2,2}$ .

## Propriétés des courbes B-spline

Elles ont les mêmes caractéristiques des courbes de Bézier :

- enveloppe convexe
- variation diminishing
- invariance par transformations affines



- ① nombre de segments de la courbe
- ② discontinuité
- ③ type de partition des noeuds

FLEXIBILITÉ

⇒ CONTRÔLE LOCALE

# 1) $n - k + 2$ morceaux de courbe

Soient

- $\mathbf{p}_0, \dots, \mathbf{p}_n$  points de contrôle ;
- $k$  l'ordre de la courbe

pour construire la courbe B-spline, on a besoin de

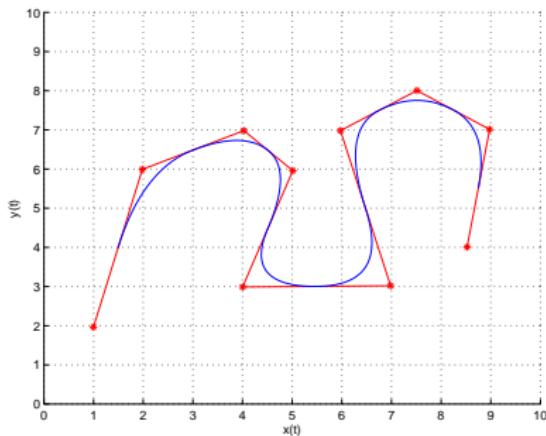
- $n + 1$  fonctions de base B-spline  $N_i^m$
- $t = [t_0, \dots, t_{n+k}]$  avec  $n + k + 1$  nœuds

Dans chaque intervalle de la partition on a  $k$  fonctions de base. Donc la courbe sur cet intervalle est combinaison linéaire de  $k$  points de contrôle avec les correspondantes fonctions de base.

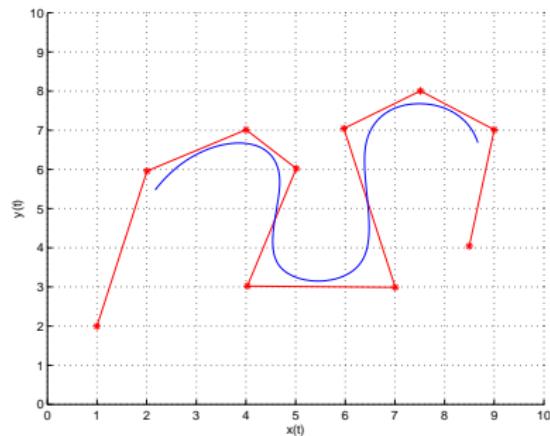
Les fonctions de base B-spline sont définies en  $[t_{k-1}, t_{n+1}] \Rightarrow n \geq m$

Pour  $n + 1$  points de contrôle  $\Rightarrow n - k + 2$  morceaux de courbe

# 1) $n - k + 2$ morceaux de courbe

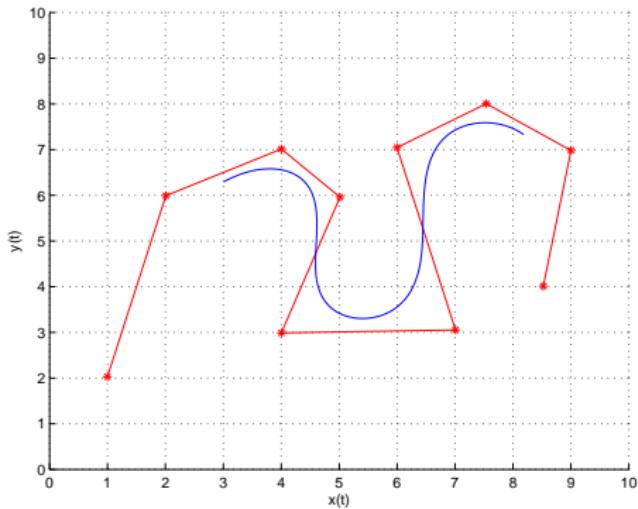


$$k = 3$$



$$k = 4$$

# 1) $n - k + 2$ morceaux de courbe



$$k = 5$$

## 2) Discontinuité



$$t_0 \leq t_1 \leq \cdots \leq t_n$$

Nœuds multiples : **nœud avec multiplicité  $p$**

$\implies -p$  degrés de continuité

Points de contrôle multiples : **points de contrôle avec multiplicité  $p$**

$\implies -p$  degrés de continuité

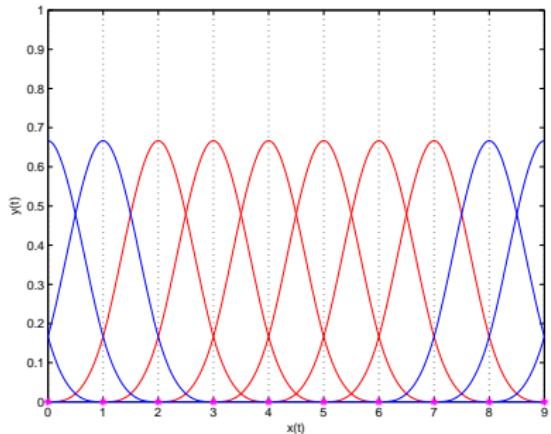
(Degré de multiplicité := nombre de fois que le nœud apparaît dans la séquence)

## 2) Discontinuité

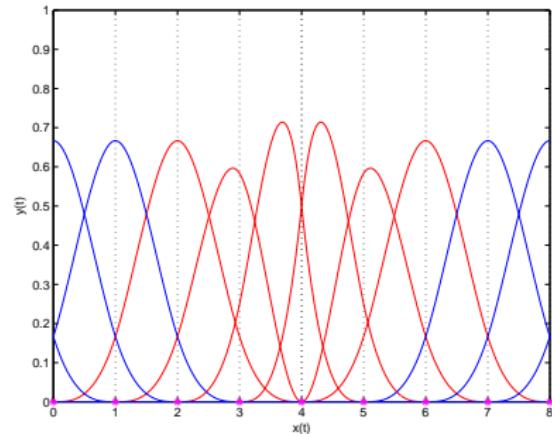
*En pratique on utilise :*

- points de contrôle multiples (avec multiplicité  $p \geq 1$ )
- nœuds intérieurs simples (avec multiplicité  $p = 1$ )
- nœuds au bord multiples (avec multiplicité  $p = k$ )

## 2) Discontinuité

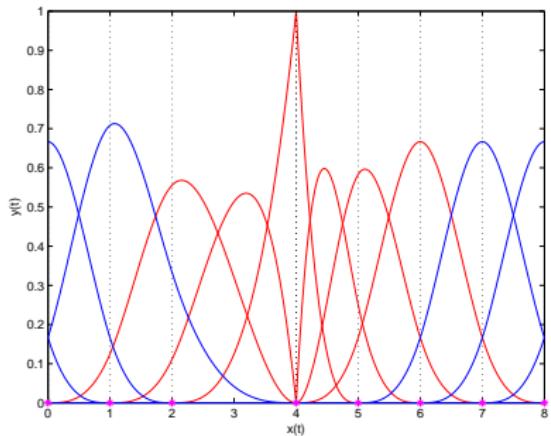


B-spline avec partition uniforme

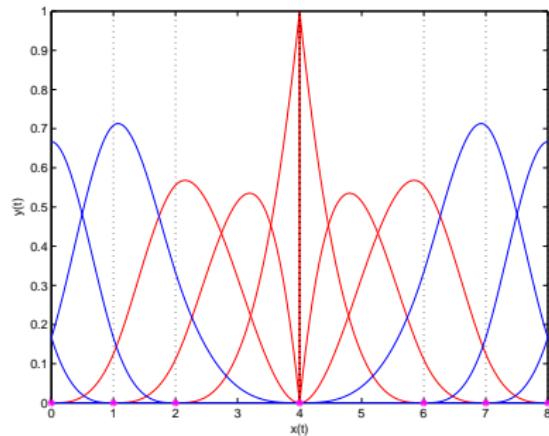


B-spline avec 1 nœud avec  
multiplicité 1

## 2) Discontinuité

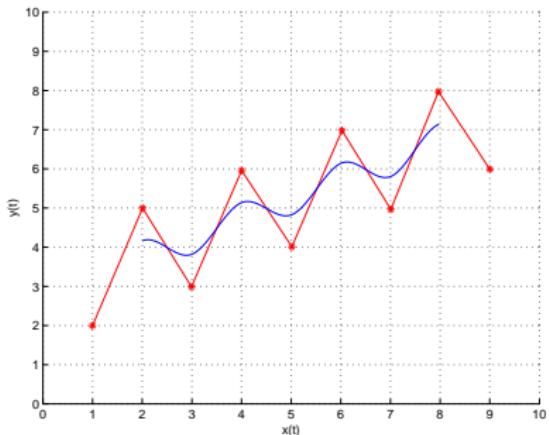


B-spline avec 1 nœud avec  
multiplicité 2

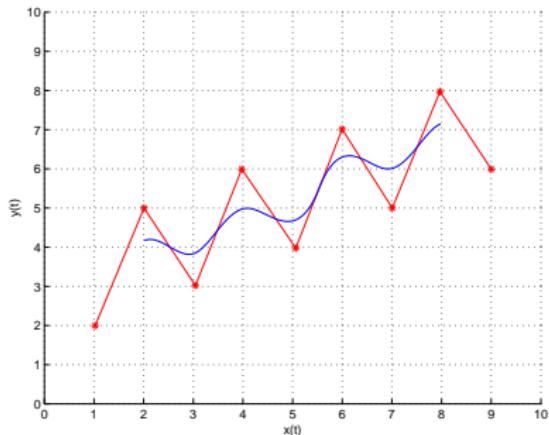


B-spline avec 1 nœud avec  
multiplicité 3

## 2) Discontinuité

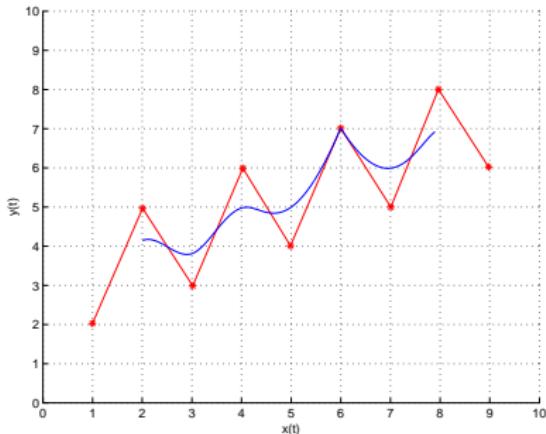


Courbe avec partition uniforme

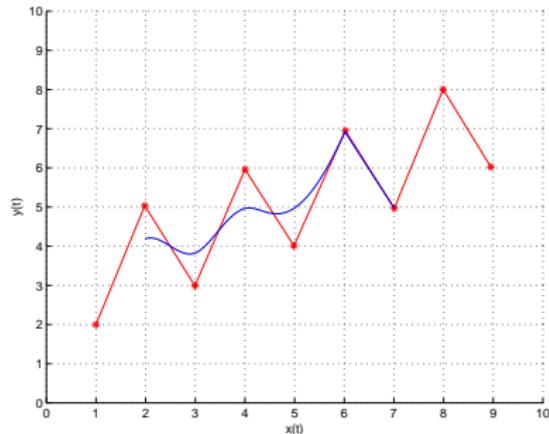


Courbe avec 1 nœud avec  
multiplicité 1

## 2) Discontinuité

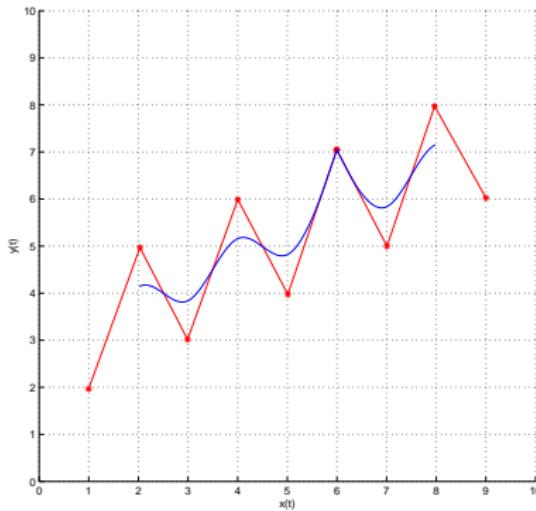


Courbe avec 1 noeud avec  
multiplicité 2



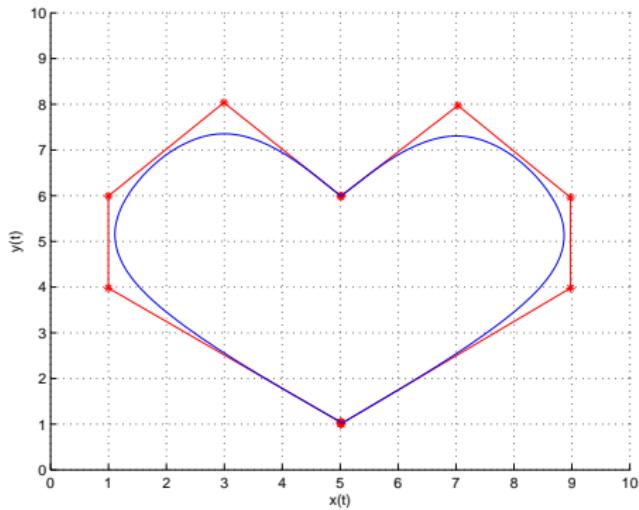
Courbe 1 nœud avec  
multiplicité 3

## 2) Discontinuité



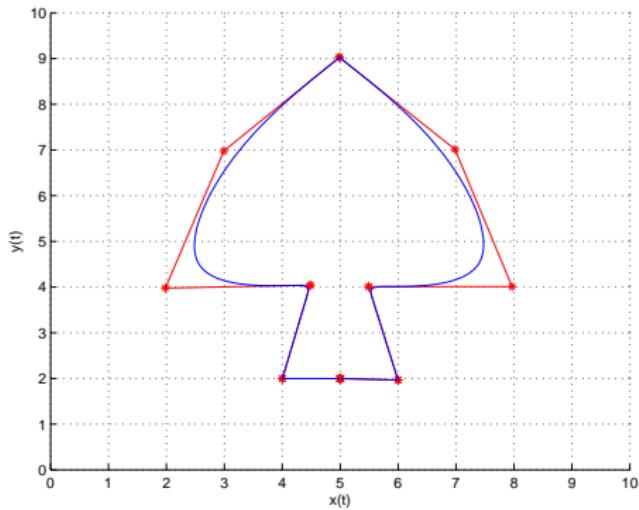
Courbe avec le 6<sup>eme</sup> point de contrôle avec multiplicité 3

## 2) Discontinuité



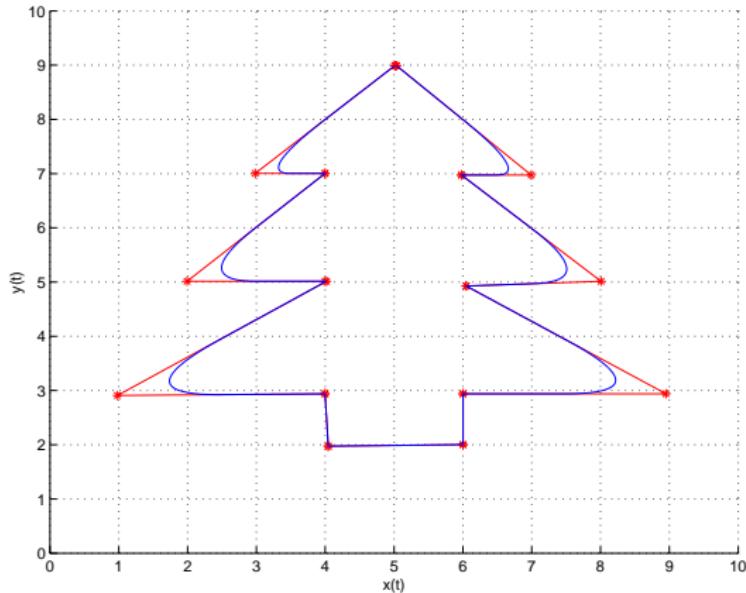
Courbe d'ordre  $k = 4$  et points de contrôle multiples

## 2) Discontinuité



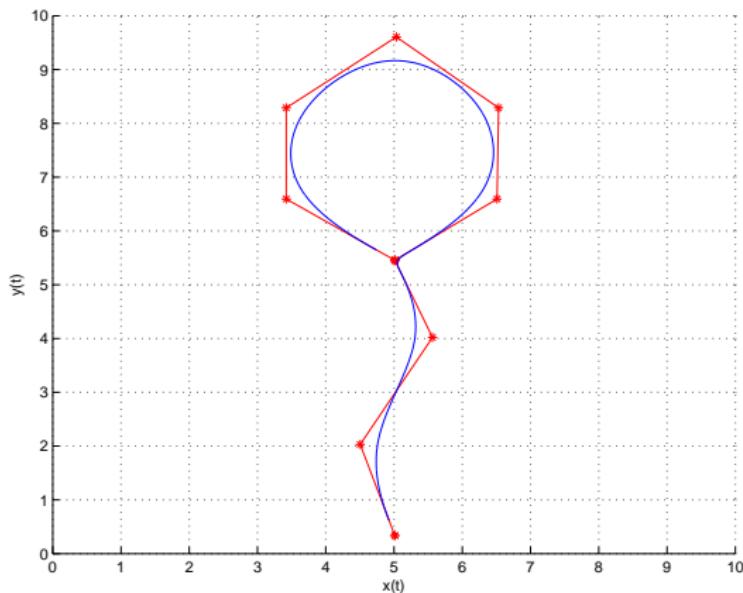
Courbe d'ordre  $k = 4$  et points de contrôle multiples

## 2) Discontinuité



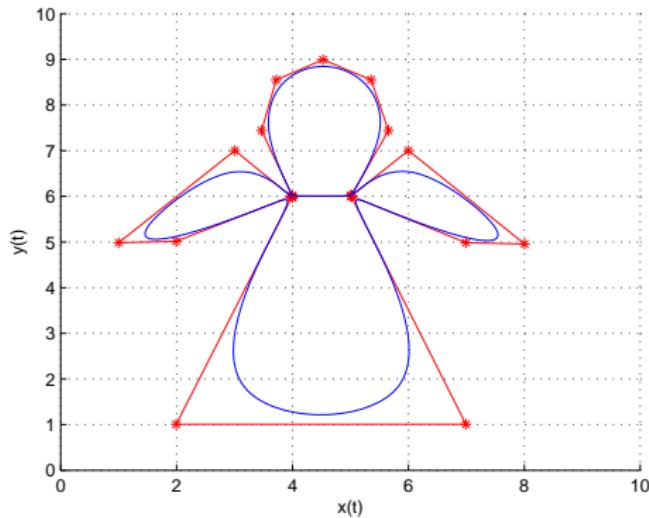
Courbe d'ordre  $k = 4$  et points de contrôle multiples

## 2) Discontinuité



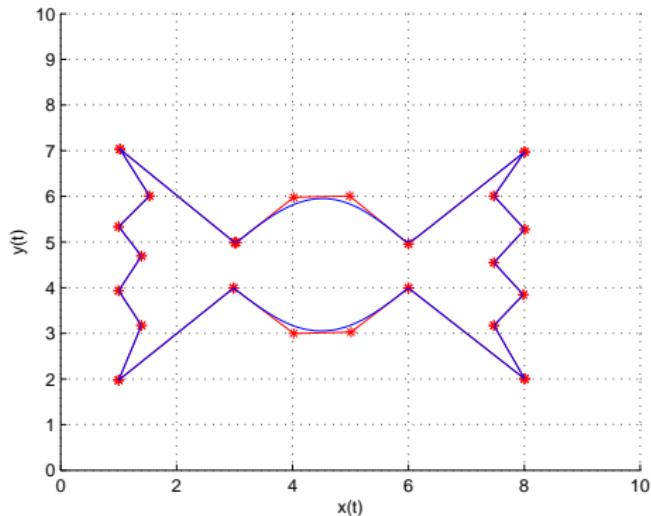
Courbe d'ordre  $k = 4$  et points de contrôle multiples

## 2) Discontinuité



Courbe d'ordre  $k = 4$  et points de contrôle multiples

## 2) Discontinuité



Courbe d'ordre  $k = 4$  et points de contrôle multiples

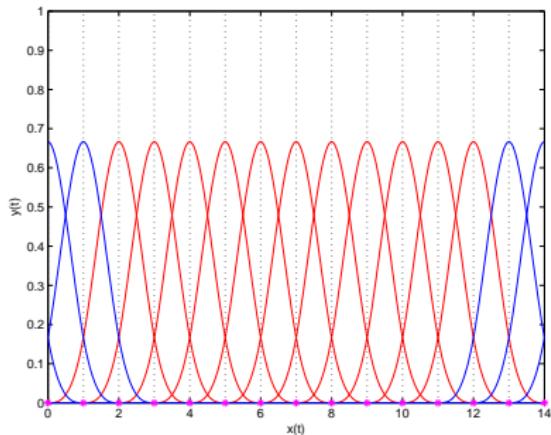
### 3) Type de partition

La partition du vecteur  $t$  des nœuds influence la forme de la courbe.

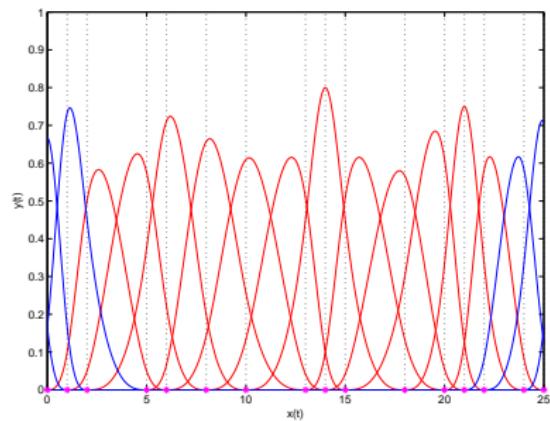
nœuds équidistants  $\implies$  Algorithme stable et efficient

(les fonctions de base sont calculées par translation)

### 3) Type de partition

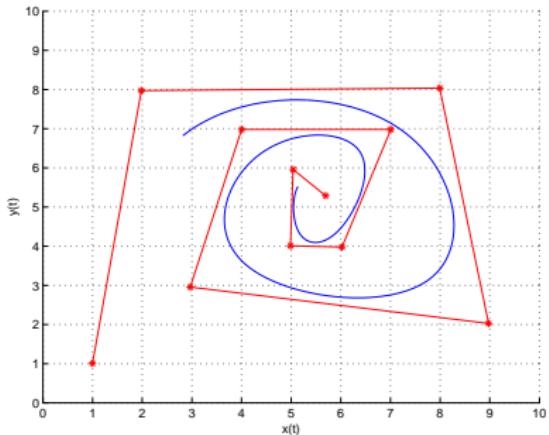


B-spline avec partition uniforme

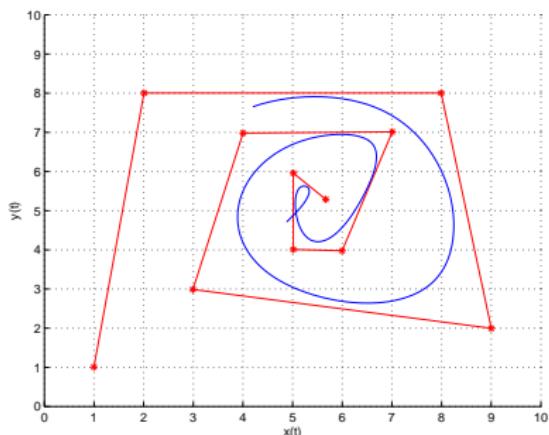


B-spline avec partition pas uniforme

### 3) Type de partition



Courbe avec partition uniforme



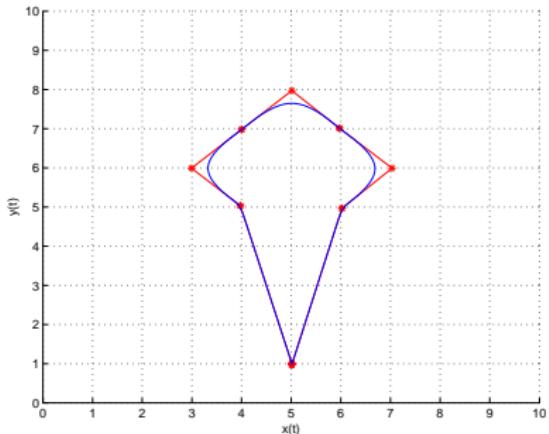
Courbe avec partition pas uniforme

## Remarques

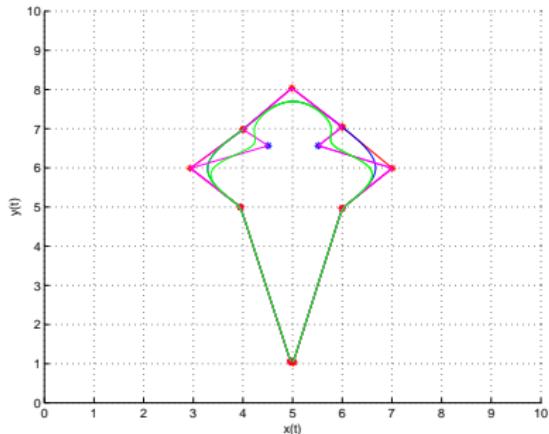
### REMARQUES:

- les courbes B-splines sont la généralisation des courbes de Bézier.
- une courbe B-spline d'ordre  $k$  avec les nœuds ou les points de contrôle au bord de multiplicité  $k$  est une courbe de Bézier.
- **CONTRÔLE LOCALE** : si on déplace  $\mathbf{p}_i$ , on ne modifie que la partie de la courbe pour laquelle  $t \in [t_i, t_{i+k}]$ .

# Contrôle locale



Courbe avec points de contrôle multiples



Courbe obtenue en déplaçant les points de contrôle multiples

## Algorithme de De Boor

L'algorithme de De Boor est un *algorithme récursif* très similaire à l'algorithme de De Casteljau.

Il permet de calculer pour une valeur du paramètre  $t^*$  le correspondant point sur la courbe B-spline, par des interpolations linéaires.

La grande différence est que cette construction agit que sur  $k$  points de contrôle (ceux qui participe à la construction du morceau de la courbe) et qu'elle utilise un paramètre  $\alpha$  construit récursivement à partir des nœuds.

## Algorithme de De Boor

Données :

- $\mathbf{p}_0, \dots, \mathbf{p}_n$
- ordre  $k$
- $[t_0, \dots, t_{n+k}]$
- paramètre  $t^*$

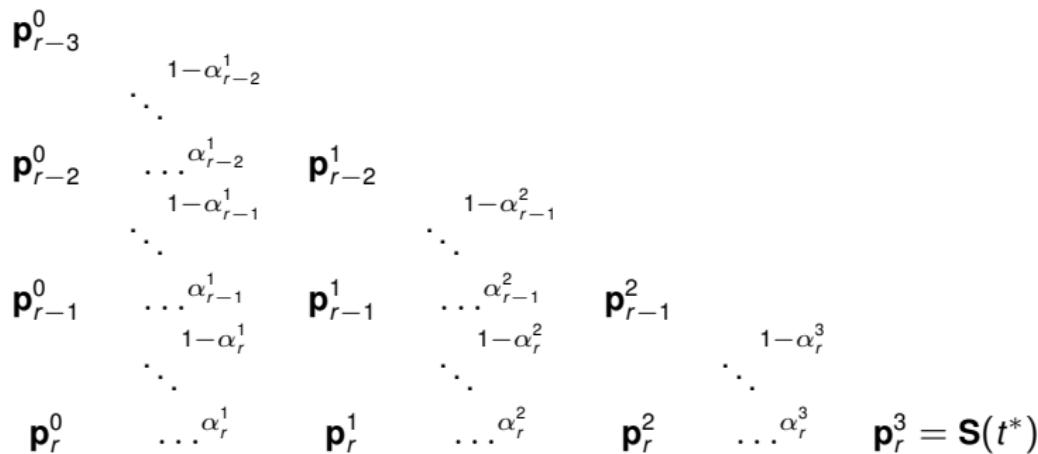
Les pas de l'algorithme :

- il trouve  $r$  tel que  $t^* \in [t_r, t_{r+1})$
- il calcule  $\alpha_i^j$  et  $\mathbf{p}_i^j$
- il calcule le point  $\mathbf{S}(t^*)$  correspondant à  $t^*$

$$\alpha_i^j = \frac{t^* - t_i}{t_{i+k-j} - t_i} \quad \text{et} \quad \mathbf{p}_i^j = (1 - \alpha_i^j) \mathbf{p}_{i-1}^{j-1} + \alpha_i^j \mathbf{p}_i^{j-1}$$

# Algorithme de De Boor

Exemple pour une courbe d'ordre  $k = 4$  :



## Subdivision

Subdivision : meilleur contrôle local

Modification de  $[t_0, \dots, t_{n+k}]$  et donc des fonctions de base :

- Algorithme de De Boor : on ajoute un nœud
- Algorithme de Oslo : on peut ajouter plusieurs nœuds

## Subdivision : Algorithme de De Boor

On ajoute que 1 noeud  $t^*$

de  $T = [t_0, \dots, t_{k-1}, \dots, t_{n+1}, \dots, t_{n+k}]$  à  
 $W = [\omega_0 = t_0, \dots, \omega_{k-1} = t_{k-1}, \omega_k = t^*, \dots, \dots, \omega_{n+k+1}]$

$$\implies \mathbf{S}(t) = \sum_{i=0}^n \mathbf{p}_i N_i^k = \sum_{i=0}^{n+1} \mathbf{p}_i^* N_i^{*k}$$

Trouver les nouveaux  $\mathbf{p}_i^*$  ? Algorithme de De Boor

## Subdivision : Algorithme de De Boor

Exemple pour une courbe d'ordre  $k = 4$  :

$$\begin{array}{ccccccccc} p_0^0 & & p_1^1 & & & & & & \\ p_1^0 & & p_1^1 & & & & & & \\ p_2^0 & & p_2^1 & & p_2^2 & & & & \\ p_3^0 & & p_3^1 & & p_3^2 & & p_3^3 & = & S(t^*) \\ p_4^0 & & & & & & & & \end{array}$$

## Algorithme de Oslo

On ajoute  $m$  nœuds

de  $T = [t_0, \dots, t_{k-1}, \dots, t_{n+1}, \dots, t_{n+k}]$  à  
 $W = [\omega_0, \dots, \omega_{k-1}, \omega_{n+m+1}, \dots, \dots, \omega_{n+k+m}]$

$$\implies \mathbf{S}(t) = \sum_{i=0}^n \mathbf{p}_i N_i^k = \sum_{i=0}^{n+m} \mathbf{p}_i^* N_i^{*k}$$

Comment trouver les nouveaux  $\mathbf{p}_i^*$  ? Algorithme de Oslo

## Algorithme de Oslo

L'espace des B-splines associé à la partition  $W$ ,  $S_{\omega,k}$  contient l'espace des B-splines associé à la partition  $T$ ,  $S_{t,k} : T \subset W$

Chaque élément de la base  $N_i^k$  est combinaison linéaire de la base  $N_j^{*k}$  :

$$N_i^k(t) = \sum_{j=0}^{n+m} \alpha_{i,k}^j N_j^{*k}(t)$$

On veut exprimer  $\mathbf{S}(t)$  en fonction de  $N_j^{*k}$  (changement de base) :

$$\mathbf{S}(t) = \sum_{i=0}^n \mathbf{p}_i N_i^k = \sum_{j=0}^{n+m} \mathbf{p}_j^* N_j^{*k}$$

## Algorithme de Oslo

$$\begin{aligned}
 \mathbf{s}(t) &= \sum_{i=0}^n \mathbf{p}_i N_i^k \\
 &= \sum_{i=0}^n \mathbf{p}_i \sum_{j=0}^{n+m} \alpha_{i,k}^j N_j^{*k}(t) = \sum_{j=0}^{n+m} \sum_{i=0}^n \alpha_{i,k}^j \mathbf{p}_i N_j^{*k}(t) \\
 &= \sum_{j=0}^{n+m} \mathbf{p}_j^* N_j^{*k}
 \end{aligned}$$

$\Rightarrow \mathbf{p}_j^* = \sum_{i=0}^n \alpha_{i,k}^j \mathbf{p}_i$  sont les nouveaux points de contrôle

pour  $j = 0, \dots, n + m$

## Algorithme de Oslo

Les coefficients du changement de base  $\alpha_{i,k}^j$  pour  $r = 2, \dots, k$  et pour  $i = 0, \dots, n$  sont

$$\alpha_{i,r}^j = \frac{\omega_{j+r-1} - t_i}{t_{i+r-1} - t_i} \alpha_{i,r-1}^j + \frac{t_{i+r} - \omega_{j+r-1}}{t_{i+r} - t_{i+1}} \alpha_{i+1,r-1}^j$$

où  $\alpha_{i,1}^j = \begin{cases} 1 & \text{si } t_i \leq \omega_j < t_{i+1} \\ 0 & \text{sinon} \end{cases}$

Les coefficients  $\alpha_{i,k}^j$  s'appellent SPLINES DISCRÈTES.

## Algorithme de Oslo

Propriétés des  $\alpha_{i,k}^j$ :

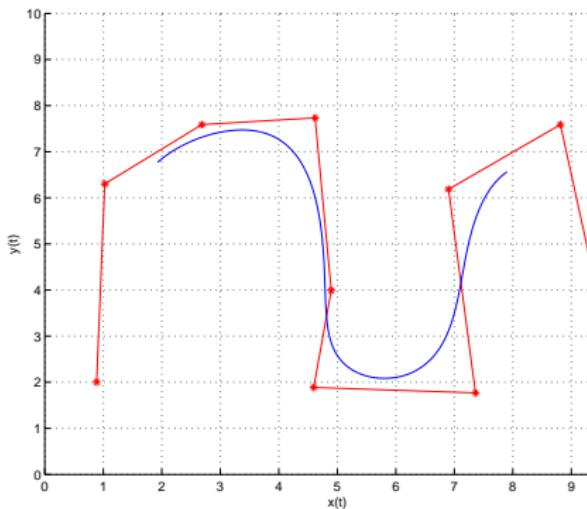
- positivité :  $\alpha_{i,k}^j \geq 0 \quad \forall i, j, k$
- Soit  $\delta$  tel que  $t_\delta \leq \omega_j \leq t_{\delta+1}$  et  $j$  fixé, alors  $\alpha_{i,k}^j = 0$   
 $\forall i \notin [\delta - (k - 1), \dots, \delta]$ . Donc elle a au plus  $k$  coefficients différents de 0 pour chaque  $j$ .
- partition de l'unité :  $\sum_{i=0}^n \alpha_{i,k}^j = 1, \quad \forall j$

## Algorithme de Oslo

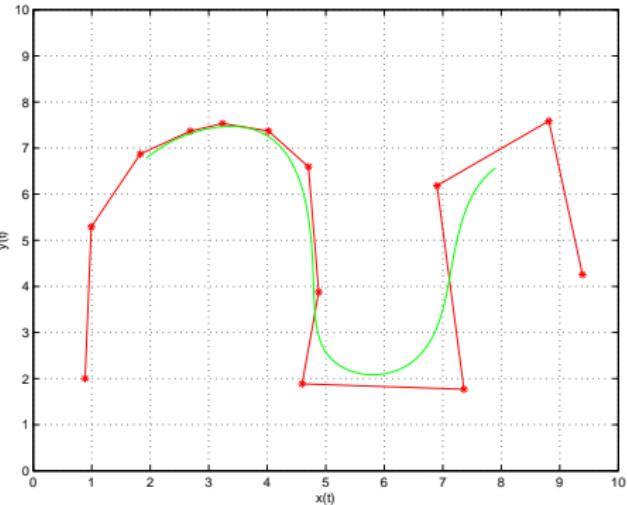
Propriétés des B-splines après la subdivision d'Oslo :

- Enveloppe convexe :  $\mathbf{p}_j^*$  forment une combinaison linéaire convexe des  $k$  points de contrôle  $\mathbf{p}_i$ .
- Le processus de subdivision est locale :  $\mathbf{p}_j$  est combinaison linéaire de  $k$  points de contrôle  $\mathbf{p}_i \in [\mathbf{p}_{\delta-k+1}, \dots, \mathbf{p}_\delta]$  correspondant aux  $N_i^k$ . Son support contient le nouveau nœud  $t_{j-k+1} \leq \omega_j \leq t_\delta$ .

## Algorithme de Oslo

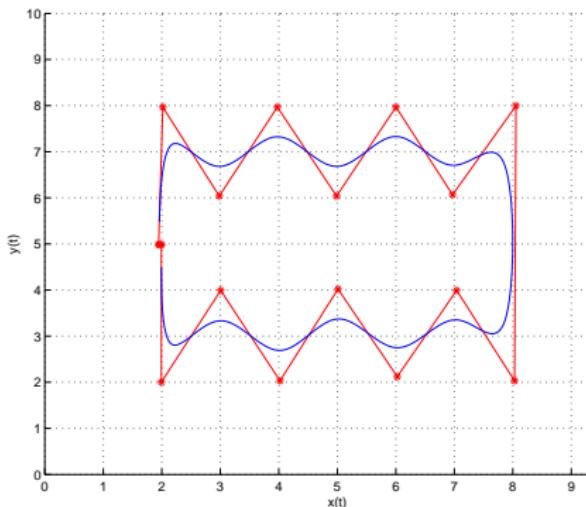


Courbe B-spline

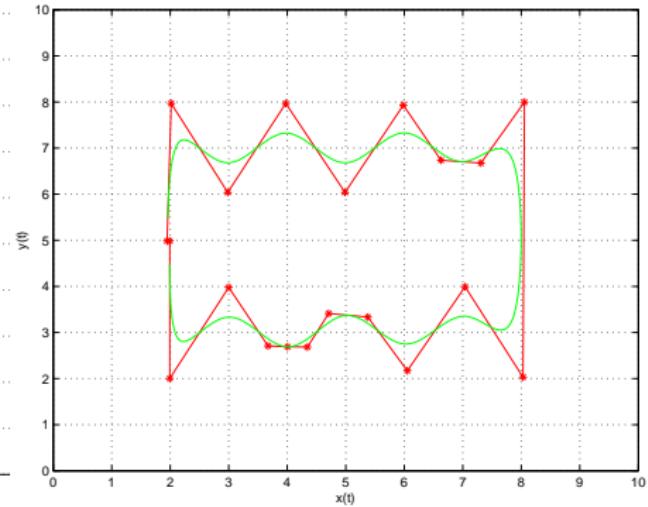


Courbe B-spline avec 3 nœuds  
en plus

## Algorithme de Oslo

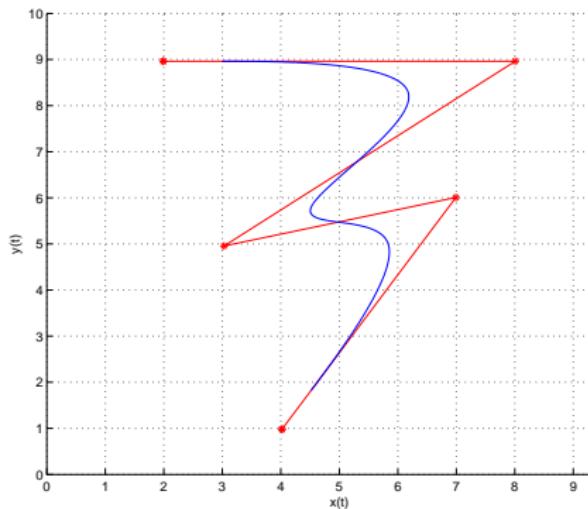


Courbe B-spline

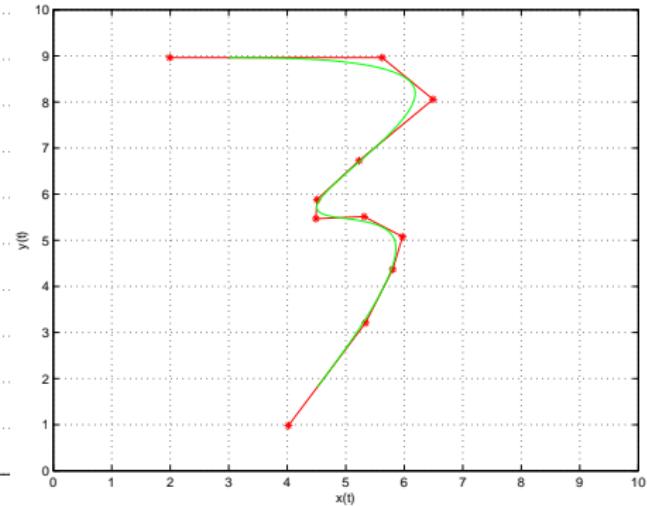


Courbe B-spline avec 4 noeuds en plus

## Algorithme de Oslo



Courbe B-spline



Courbe B-spline avec 6 points  
de contrôle en plus

## EXERCICE

Soit une courbe B-spline définie par 5 points de contrôle  $P_0(2, 2)$ ,  $P_1(3, 4)$ ,  $P_2(5, 1)$ ,  $P_3(7, 3)$  et  $P_4(9, 2)$  d'ordre 3.

- ① Donner le vecteur de nœuds non périodique (nœuds de multiplicité  $k$  aux extrémités) de cette courbe.
- ② Calculer la fonction de base  $N_2^2$ , en sachant que

$$N_i^m(t) = \frac{t - t_i}{t_{i+m} - t_i} N_i^{m-1}(t) + \frac{t_{i+m+1} - t}{t_{i+m+1} - t_{i+1}} N_{i+1}^m(t)$$

- ③ Le point de contrôle  $P_3(7, 3)$  est déplacé du vecteur  $[1, 2]$ . Quelle portion de la courbe est affectée par cette translation si  $k = 2$  ?
- ④ Pour quelle valeur de  $k$  cette B-spline serait équivalent à une courbe de Bézier ?  
Déterminer son vecteur des nœuds.
- ⑤ Si le 2<sup>e</sup> point de cette B-spline est dédoublé :
  - a) cela aura-t-il un effet sur la forme de la courbe ?
  - b) cela aura-t-il un effet sur la forme de fonctions de base ?

## EXERCICE

Construire un vecteur de nœuds et de points de contrôle (le moins possible) dans le plan de sorte que la B-spline de degré 3 associée passe par les points  $M = (1, 0)$  et  $N = (3, 0)$  avec en ces deux points une tangente dirigée par  $(1, 1)$ .

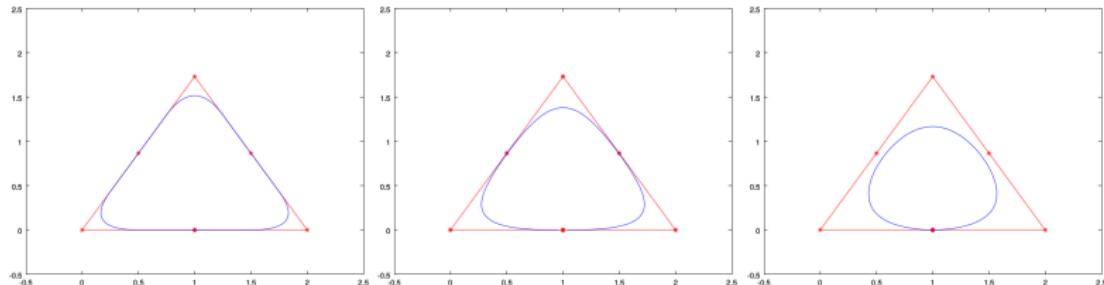
## Introduction

Peut-on représenter en manière exacte un cercle avec une courbe spline ?

NON!

On peut représenter que des courbes polynomiales, mais pas des coniques (cercles, ellipses, paraboles, hyperboles)!

# Introduction



$k = 3$

$k = 5$

$k = 10$

$t$  uniforme

## Définition

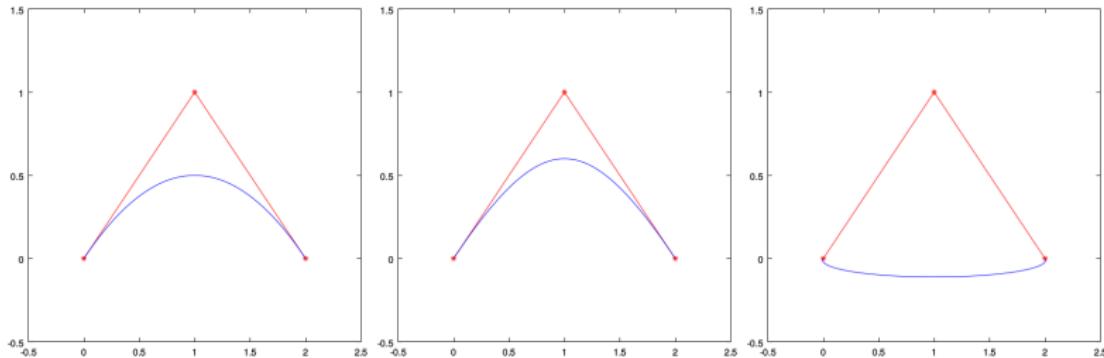
### NURBS

Soient  $\mathbf{p}_0, \dots, \mathbf{p}_n$  points de contrôle, une courbe B-spline rationnelle (NURBS, Non-Uniform Rational Basis Splines) est définie comme

$$\bar{\mathbf{s}}(t) = \frac{\sum_{i=0}^n \mathbf{p}_i \omega_i N_i^m(t)}{\sum_{i=0}^n \omega_i N_i^m(t)} = \sum_{i=0}^n \mathbf{p}_i \bar{R}_i^m$$

où  $\omega_i \in \mathbb{R}$  sont appelés **poids** et  $\bar{R}_i^m$  sont les fonctions B-splines rationnelles.

## Paramètre de forme



$$\omega_1 = 1$$

$$\omega_1 > 1$$

$$\omega_1 < 0$$

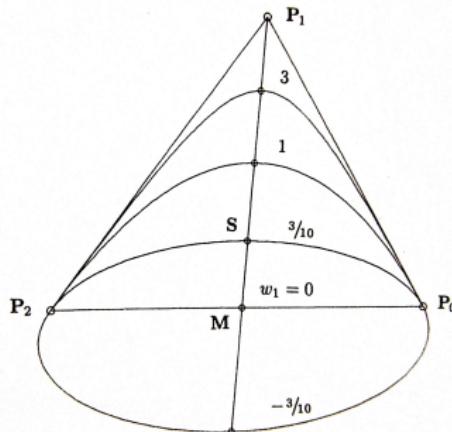
## Coniques

NURBS de degré 2 ouverte avec trois points de contrôle et  $\omega_0 = \omega_2 = 1$  :

$$\begin{aligned}\bar{\mathbf{S}}(t) &= \frac{\mathbf{p}_0 N_0^2(t) + \mathbf{p}_1 \omega_1 N_1^2(t) + \mathbf{p}_2 N_2^2(t)}{N_0^2(t) + \omega_1 N_1^2(t) + N_2^2(t)} \\ &= \frac{(1-t)^2 \mathbf{p}_0 + 2\omega_1 t(1-t) \mathbf{p}_1 + t^2 \mathbf{p}_2}{(1-t)^2 + 2\omega_1 t(1-t) + t^2}\end{aligned}$$

# Coniques

Pour  $t = 0.5$  on a :



$$\begin{aligned}\mathbf{S} = \bar{\mathbf{S}}(0.5) &= \frac{(\mathbf{p}_0 + \mathbf{p}_2)/2}{1 + \omega_1} + \frac{\omega_1 \mathbf{p}_1}{1 + \omega_1} \\ &= \frac{1}{1 + \omega_1} \mathbf{M} + \frac{\omega_1}{1 + \omega_1} \mathbf{p}_1 = (1-s)\mathbf{M} + s\mathbf{p}_1\end{aligned}$$

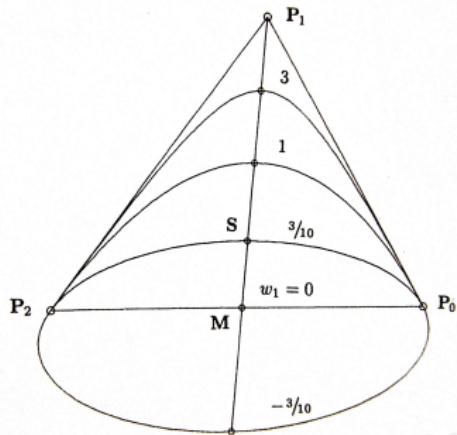
où

$$s = \frac{\omega_1}{1 + \omega_1} \quad \text{et} \quad \omega_1 = \frac{s}{1 - s}$$

$s$  est un nouveau paramètre qui donne l'interpolation linéaire entre  $\mathbf{M}$  et  $\mathbf{p}_1$ .

Le point **S** est appelé **shoulder point** de la conique

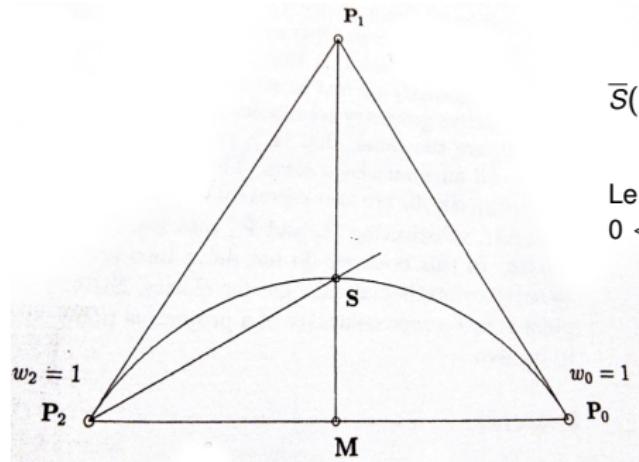
# Coniques



$w_1$  détermine le type de conique :

- $w_1 < 1$  : ellipse
- $w_1 = 1$  : parabole
- $w_1 > 1$  : hyperbole

## Arc de cercle



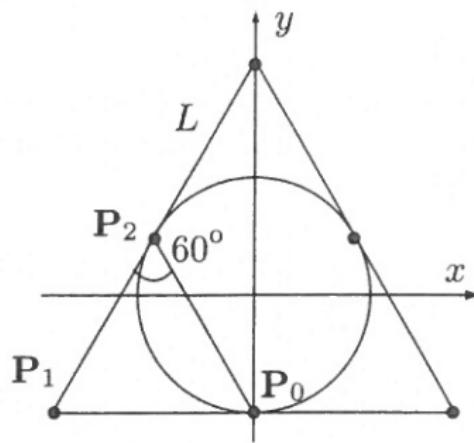
$$\bar{S}(t) = \frac{(1-t)^2 \mathbf{p}_0 + 2\omega_1 t(1-t) \mathbf{p}_1 + t^2 \mathbf{p}_2}{(1-t)^2 + 2\omega_1 t(1-t) + t^2}$$

Le triangle est isocèle avec  $\mathbf{p}_0\mathbf{p}_1 = \mathbf{p}_1\mathbf{p}_2$  et  $0 < \omega_1 < 1$ .

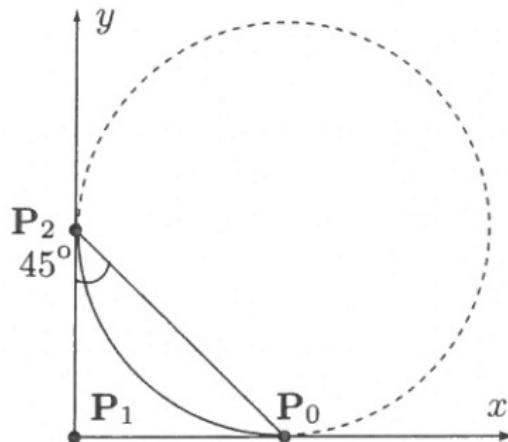
$$\omega_1 = \frac{\mathbf{M}\mathbf{S}}{\mathbf{S}\mathbf{p}_1} = \frac{\mathbf{M}\mathbf{p}_2}{\mathbf{p}_2\mathbf{p}_1} = \cos(\theta)$$

avec  $\theta = \widehat{\mathbf{p}_1\mathbf{p}_2\mathbf{M}}$

## Arc de cercle

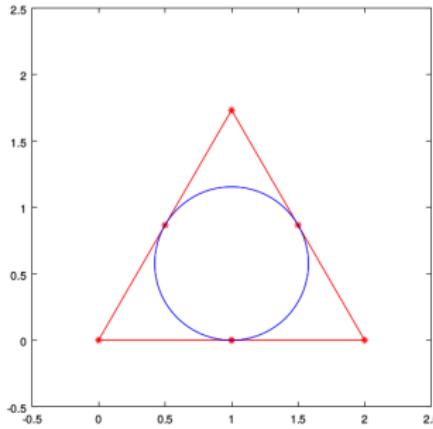


$$\theta = 60^\circ$$



$$\theta = 45^\circ$$

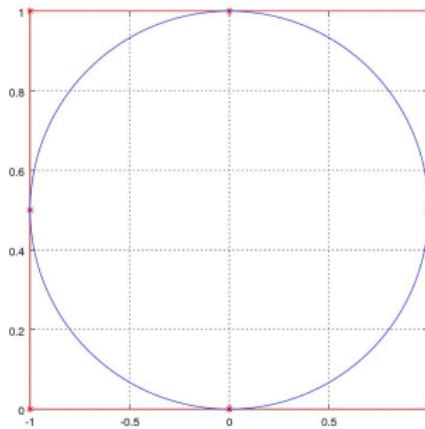
## Cercle : NURBS



$$\mathbf{p} = \{(1, 0), (0, 0), \left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right), (1, \sqrt{3}), \left(\frac{3}{2}, \frac{\sqrt{3}}{2}\right), (2, 0), (1, 0)\}$$

$$k = 3, \quad t = (0, 0, 0, \frac{1}{3}, \frac{1}{3}, \frac{2}{3}, \frac{2}{3}, 1, 1, 1), \quad \omega = (1, \frac{1}{2}, 1, \frac{1}{2}, 1, \frac{1}{2}, 1)$$

## Cercle : NURBS



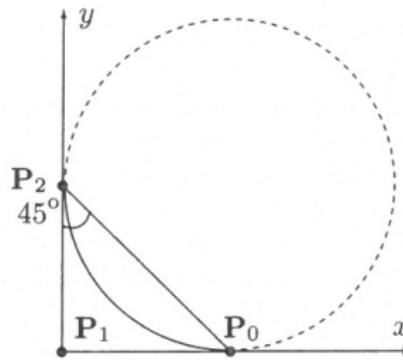
$$\mathbf{p} = \{(0,0), (-1,0), \left(-1, \frac{1}{2}\right), (-1,1), (0,1), (1,1), \left(1, \frac{1}{2}\right), (1,0), (0,0)\}$$

$$k = 3, \quad t = (0, 0, 0, \frac{1}{4}, \frac{1}{2}, \frac{1}{2}, \frac{3}{4}, \frac{3}{4}, 1, 1, 1), \quad \omega = (1, \frac{\sqrt{2}}{2}, 1, \frac{\sqrt{2}}{2}, 1, \frac{\sqrt{2}}{2}, 1, \frac{\sqrt{2}}{2}, 1)$$



## EXERCICE 1

Donner l'expression de la courbe B-spline rationnelle représentée sur la figure ci-dessous.  
Elle est de degré 2 avec vecteur des nœuds  $t = [0, 0, 0, 1, 1, 1]$  et ses points de contrôle sont  $\mathbf{p}_0 = (1, 0)r$ ,  $\mathbf{p}_1 = (0, 0)$  et  $\mathbf{p}_2 = (0, 1)r$ , avec  $r$  le rayon du cercle.



## EXERCICE 2

Nous voulons construire une NURBS de degré 3 avec 5 points de contrôle.

Définir les éléments qui permettent de la construire.