

LICENCE 3 DE MATHÉMATIQUES  
UFR BOURGOGNE FRANCHE-COMTÉ

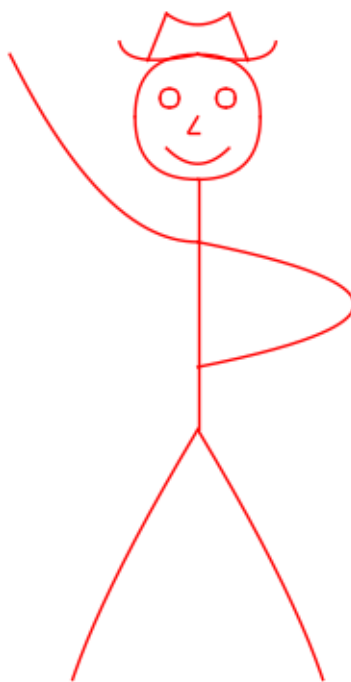
---

# Courbes de Bézier

## et B-splines

FOSSEZ PAULINE - LAUMONT ESTELLE  
VUILLEMOT KILLIAN

---



*Année 2018-2019*

# Table des matières

I	Généralités sur les courbes de Bézier . . . . .	2
	I. 1) Les polynômes de Bernstein . . . . .	2
	I. 2) Courbes de Bézier . . . . .	7
	I. 3) Algorithme de De Casteljau . . . . .	12
II	B-splines . . . . .	17
	II. 1) Fonctions B-splines . . . . .	17
	II. 2) Courbes B-splines . . . . .	19

### Introduction

Dans ce rapport, nous allons aborder le problème de la modélisation informatique des courbes. Nous allons pour cela voir deux approches différentes. La première approche sera celle des courbes de Bézier qui constituera la majeure partie de notre développement. Puis nous verrons une seconde méthode, celle des B-splines. Les courbes de Bézier ont été découvertes et utilisées par Pierre Bézier et Paul de Casteljau. Les deux ingénieurs (P. Bézier chez Renault et P. de Casteljau chez Citroën) cherchaient tous deux à réaliser des courbes par l'informatique. Il était, dans un contexte de développement du numérique et de l'informatique, nécessaire de réaliser des modélisations de courbes par informatique, cela représentant un gain de temps et d'argent par rapport à des réalisations de maquettes. Les courbes de Bézier ont été une révolution dans ce domaine. En effet, elles se sont avérées plus rapides, plus efficaces et plus souples que les précédentes méthodes déjà existantes. Ces modélisations servaient notamment pour les conceptions de carrosserie ; par la suite elles furent ensuite utilisées dans différents domaines. De nos jours, les courbes de Bézier sont toujours l'un des principaux outils de la plupart des logiciels de dessin vectoriel et sont également la base de la police Post-Script. Afin d'approfondir un peu le sujet, nous parlerons ensuite d'une généralisation des courbes de Bézier : les B-splines. Ce type de modélisation est quant à lui apparu dans les années 80, notamment dans l'aviation (chez Boeing). Nous verrons alors les avantages de ce type de modélisation par rapport aux courbes de Bézier.

## I Généralités sur les courbes de Bézier

Dans un premier temps, nous allons aborder la notion de courbes de Bézier. Tout d'abord, nous définirons donc quelques notions qui nous seront utiles tout au long de cette partie. Puis, nous verrons la méthode des courbes paramétrées de Pierre Bézier avant de nous intéresser à l'algorithme de Paul De Casteljau.

### I. 1) Les polynômes de Bernstein

**Rappel I.1.** Le coefficient binomial  $i$  parmi  $n$ , noté  $\binom{n}{i}$  ou parfois  $C_n^i$  est défini par :

$$\begin{cases} \frac{n!}{i!(n-i)!} & \text{si } i \in \llbracket 0, n \rrbracket \\ 0 & \text{sinon} \end{cases}$$

## I. GÉNÉRALITÉS SUR LES COURBES DE BÉZIER

---

**Définition I.1.** Soient  $n \in \mathbb{N}$ ,  $i \leq n$  et  $t \in [0, 1]$ . On appelle polynômes de Bernstein d'ordre  $n$  les polynômes définis par :

$$B_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i}$$

Pour tout  $n \in \mathbb{N}$ , on obtiendra une famille de  $n + 1$  polynômes de Bernstein d'ordre  $n$ .

**Remarque.** Parfois, lorsque  $n$  est fixé, on trouve la notation  $B_i(t)$ .

Nous allons maintenant déterminer les représentations graphiques de polynômes de Bernstein d'ordre 1, 2, 3 et 4. Cela permettra une compréhension et une visualisation plus aisées de cette notion.

Soient  $B_{i,n}(t)$  des polynômes de Bernstein d'ordre  $n$  avec  $t \in [0, 1]$ . En posant  $n = 1$ , on va obtenir deux polynômes de Bernstein à l'ordre 1 :  $B_{0,1}(t)$  et  $B_{1,1}(t)$ . Ces polynômes étant de degré 1, on obtient des portions de droites. On peut alors représenter ces deux polynômes graphiquement, comme sur la figure 1 page 3.

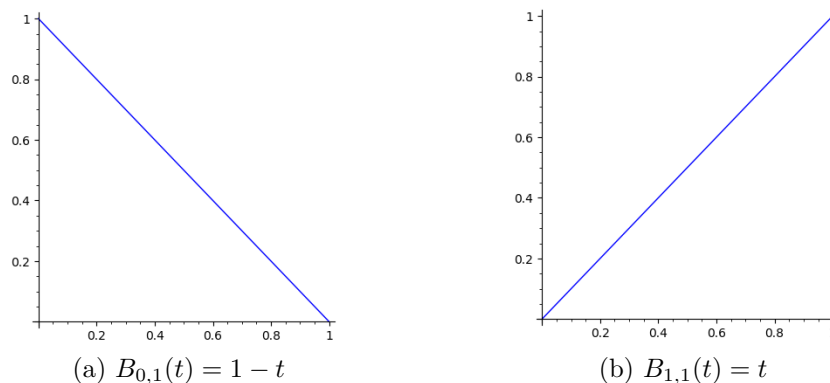


FIGURE 1 – Tracés des polynômes de Bernstein d'ordre 1

On pose maintenant  $n = 2$ . On obtient alors les 3 polynômes  $B_{0,2}$ ,  $B_{1,2}$  et  $B_{2,2}$  qui sont des polynômes de Bernstein quadratiques. Ces polynômes sont de degré 2, on a donc des portions de paraboles qui sont représentées graphiquement sur la figure 2 page 4.

Nous allons maintenant représenter les polynômes d'ordres 3 et 4, c'est-à-dire les

## I. GÉNÉRALITÉS SUR LES COURBES DE BÉZIER

---

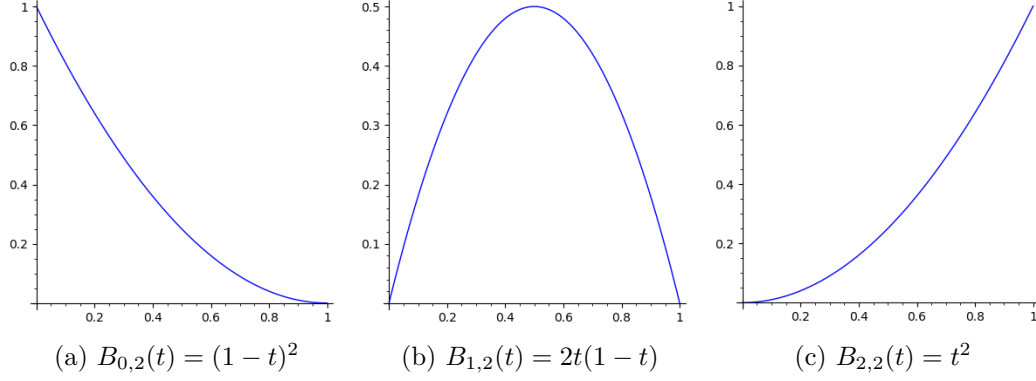


FIGURE 2 – Tracés des polynômes de Bernstein d'ordre 2

$B_{i,3}(t)$  et les  $B_{i,4}(t)$ . Ces polynômes sont respectivement de degré 3 et de degré 4. On obtient donc les représentations de la figure 3 page 4.

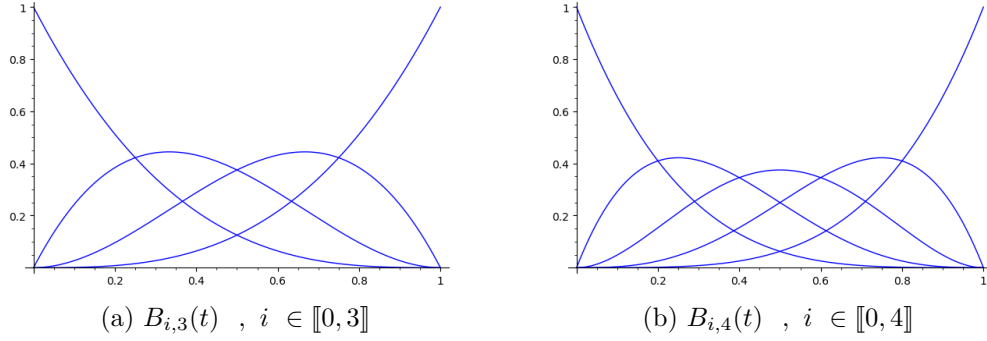


FIGURE 3 – Tracés des polynômes de Bernstein d'ordre 3 et 4

**Proposition I.1.** Soit  $n \in \mathbb{N}$ . La famille des polynômes de Bernstein d'ordre  $n$ , notée  $(B_{i,n})_{0 \leq i \leq n}$  forme une base de l'espace des polynômes de degré inférieur ou égal à  $n$ , à coefficients réels,  $\mathbb{R}_n[X]$ .

*Démonstration.* Soient  $n \in \mathbb{N}$  et  $i \in \llbracket 0, n \rrbracket$ . On pose  $B_{i,n}(X) = \binom{n}{i} X^i (1-X)^{n-i}$ . Montrons que  $(B_{i,n})_{0 \leq i \leq n}$  est une famille libre de  $\mathbb{R}_n[X]$ . On montre cela par récurrence. Posons pour cela :

## I. GÉNÉRALITÉS SUR LES COURBES DE BÉZIER

---

$\mathcal{P}(n) : " \forall k \leq n, (B_{i,k}) \text{ est une famille libre de } \mathbb{R}_k[X] "$

*Initialisation* :  $n = 0$ .

Si  $n = 0$ , la famille de polynômes est constituée seulement de  $B_{0,0}$ . Or, on a :

$$B_{0,0}(X) = \binom{0}{0} X^0 (1-X)^0 = 1$$

Donc au rang  $n = 0$ , la famille est libre et la propriété est vraie.

*Hérédité* :

On suppose maintenant la propriété vraie pour tout  $n \in \mathbb{N}$ . Montrons qu'elle l'est également au rang  $n + 1$ .

Soient  $\lambda_0, \lambda_1, \dots, \lambda_{n+1}$  tels que :

$$\lambda_0 B_{0,n+1} + \lambda_1 B_{1,n+1} + \dots + \lambda_{n+1} B_{n+1,n+1} = 0$$

On a donc :

$$\begin{aligned} & \lambda_0 B_{0,n+1}(X) + \lambda_1 B_{1,n+1}(X) + \dots + \lambda_{n+1} B_{n+1,n+1}(X) = 0 \\ \Leftrightarrow & \lambda_0 (1-X)^{n+1} + \lambda_1 \binom{n+1}{1} X(1-X)^n + \dots + \lambda_{n+1} \binom{n+1}{n+1} X^{n+1} = 0 \\ \Leftrightarrow & \lambda_0 (1-X)^{n+1} + \lambda_1 (n+1) X(1-X)^n + \dots + \lambda_{n+1} X^{n+1} = 0 \end{aligned}$$

On peut alors évaluer ce polynôme en  $X = 0$  puis en  $X = 1$ . On obtient alors successivement  $\lambda_0 = 0$  puis  $\lambda_{n+1} = 0$ .

On obtient donc une nouvelle équation :

$$\lambda_1 (n+1) X(1-X)^n + \dots + \lambda_n X^n (1-X) = 0$$

On suppose alors  $X$  différent de 0 et de 1. On peut donc simplifier cette équation par  $X(1-X)$ .

Alors, on a :

$$\begin{aligned} & \lambda_1 (n+1) (1-X)^{n-1} + \dots + \lambda_n (n+1) X^{n-1} = 0 \\ \Leftrightarrow & \lambda_1 (n+1) B_{0,n-1} + \dots + \lambda_n (n+1) B_{n-1,n-1} = 0 \end{aligned}$$

Or, d'après notre hypothèse de récurrence,  $\mathcal{P}(n-1)$  est vraie, donc la famille  $(B_{i,n-1})_{0 \leq i \leq n-1}$  est libre. Alors :  $\lambda_1 = \dots = \lambda_n = 0$ .

## I. GÉNÉRALITÉS SUR LES COURBES DE BÉZIER

---

Enfin,  $\mathcal{P}(n+1)$  est vraie, donc  $(B_{i,n+1})_{0 \leq i \leq n+1}$  est libre.

Alors,  $\mathcal{P}(n)$  est vraie pour tout  $n \in \mathbb{N}$ .

De plus, la famille  $(B_{i,n})_{0 \leq i \leq n}$  est une famille de  $n+1$  polynômes. Or,  $\mathbb{R}_n[X]$  est de dimension  $n+1$ . Donc on a bien une base de  $\mathbb{R}_n[X]$ .  $\square$

**Proposition I.2.** Soit  $(B_{i,n})_{i \leq n}$  une famille de polynômes de Bernstein d'ordre  $n$ .

Soit  $t \in [0, 1]$ . On a alors  $B_{i,n}(t) \geq 0$  et  $\sum_{i=0}^n B_{i,n}(t) = 1$

*Démonstration.* On suppose  $n$  fixé.

On sait que  $t \in [0, 1]$ . D'où :

$$t^i \geq 0 \quad \text{et} \quad (1-t)^{n-i} \geq 0$$

Or,  $\binom{n}{i} \geq 0$ . Donc on a bien  $B_{i,n}(t) \geq 0$ , pour  $t \in [0, 1]$ .

De plus :

$$B_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i}$$

D'où, en reconnaissant la formule du binôme de Newton :

$$\begin{aligned} \sum_{i=0}^n B_{i,n}(t) &= \sum_{i=0}^n \binom{n}{i} t^i (1-t)^{n-i} \\ &= (t + 1 - t)^n \\ &= 1 \end{aligned}$$

$\square$

**Proposition I.3.** Soit  $n \in \mathbb{N}^*$  et soit  $i \leq n$ . Alors le polynôme de Bernstein  $B_{i,n}(t)$  admet un maximum en  $t = \frac{i}{n}$ . De plus, on a :

$$\max(B_{i,n}(t)) = \binom{n}{i} \left(\frac{i}{n}\right)^i \left(1 - \frac{i}{n}\right)^{n-i}$$

*Démonstration.* On dérive  $B_{i,n}(t)$  par rapport à  $t$  :

$$\begin{aligned} (B_{i,n})'(t) &= \binom{n}{i} i t^{i-1} (1-t)^{n-1} - \binom{n}{i} t^i (n-i) (1-t)^{n-i-1} \\ &= \binom{n}{i} t^{i-1} (1-t)^{n-i-1} [i(1-t) - t(n-i)] \\ &= \binom{n}{i} t^{i-1} (1-t)^{n-i-1} (i - tn) \end{aligned}$$

## I. GÉNÉRALITÉS SUR LES COURBES DE BÉZIER

---

Donc  $(B_{i,n})'(t) = 0$  si et seulement si  $t = 0$  ou  $t = 1$  ou  $t = \frac{i}{n}$ .

On a  $B_{i,n}(0) = B_{i,n}(1) = 0$ , donc 0 et 1 correspondent au minimum de  $B_{i,n}$ .

De plus :  $i - tn \geq 0 \Leftrightarrow t \leq \frac{i}{n}$

Il s'agit donc bien d'un maximum.

Finalement, on a :

$$\max(B_{i,n}(t)) = B_{i,n}\left(\frac{i}{n}\right) = \binom{n}{i} \left(\frac{i}{n}\right)^i \left(1 - \frac{i}{n}\right)^{n-i}$$

□

**Proposition I.4.** Soit  $B_{i,n}(t)$  un polynôme de Bernstein d'ordre  $n$  avec  $i \in \llbracket 1, n-1 \rrbracket$ .

Alors :

$$B_{i,n}(t) = (1-t)B_{i,n-1}(t) + tB_{i-1,n-1}(t)$$

*Démonstration.* Avec le triangle de Pascal on a :

$$\begin{aligned} B_{i,n}(t) &= \left( \binom{n-1}{i-1} + \binom{n-1}{i} \right) t^i (1-t)^{n-i} \\ &= t \binom{n-1}{i-1} t^{i-1} (1-t)^{n-i} + (1-t) \binom{n-1}{i} t^i (1-t)^{n-i-1} \\ &= tB_{i-1,n-1}(t) + (1-t)B_{i,n-1}(t) \end{aligned}$$

□

**Remarque.** On a :

$$B_{0,n}(t) = \binom{n}{0} t^0 (1-t)^n = (1-t)^n \quad \text{et} \quad B_{n,n}(t) = \binom{n}{n} t^n (1-t)^0 = t^n$$

### I. 2) Courbes de Bézier

**Définition I.2.** La courbe de Bézier d'ordre  $n$  associée à  $n+1$  points  $A_0, A_1, \dots, A_n$  est la courbe paramétrée  $M$  de paramétrisation  $\alpha$  définie par :

$$\begin{aligned} \alpha : [0, 1] &\longrightarrow \mathbb{R}^2 \\ t &\longmapsto \sum_{i=0}^n B_{i,n}(t) A_i \end{aligned}$$



## I. GÉNÉRALITÉS SUR LES COURBES DE BÉZIER

---

$A_0$  et  $A_n$  sont les extrémités de la courbe  $M$ . Ils sont appelés points d’ancrage. La courbe ne passe généralement pas par les points  $A_1, \dots, A_{n-1}$  qui sont appelés points de contrôle.

**Définition I.3.** Soit  $M$  une courbe de Bézier d’ordre  $n$  contrôlée par des points  $A_0, \dots, A_n$  tels que  $A_i = (x_i, y_i)$  pour  $i \in \llbracket 0, n \rrbracket$  et de paramétrisation  $\alpha$ . On a alors :

$$\alpha(t) = \begin{cases} x(t) = \sum_{i=0}^n B_i(t)x_i \\ y(t) = \sum_{i=0}^n B_i(t)y_i \end{cases}$$

Et on a donc  $\alpha(t) = (x(t), y(t))$ .

**Remarque.** Les courbes de Bézier étant définies par des points de contrôle, la modification de chaque point entraîne une modification globale de la courbe et non pas locale.

Voici par exemple une courbe de Bézier tracée avec SageMath, [4a](#), puis une autre après modification d’un seul point de contrôle, [4b](#).

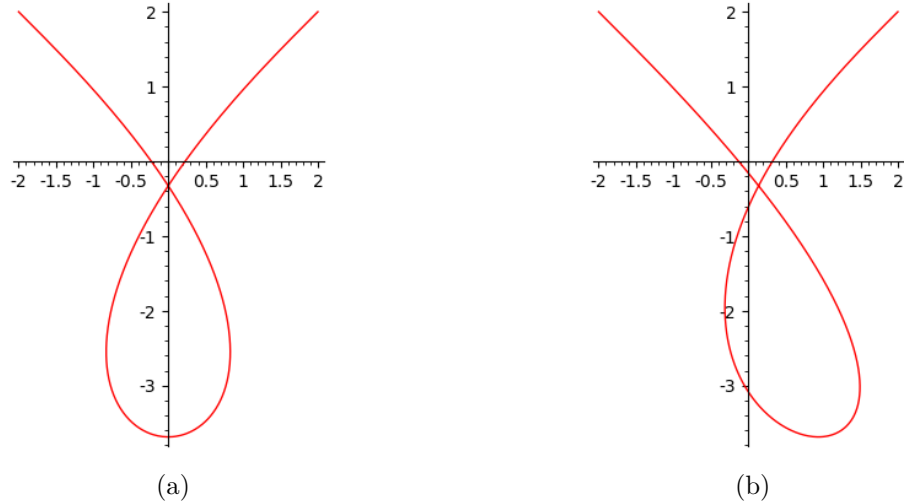


FIGURE 4 – Tracés de courbes de Bézier d’ordre 6

## I. GÉNÉRALITÉS SUR LES COURBES DE BÉZIER

---

**Proposition I.5.** Soit  $M$  une courbe de Bézier d'ordre  $n$  associée aux points de  $(A_0, \dots, A_n)$  paramétrée par  $\alpha$ .

La paramétrisation  $\alpha$  est alors de classe  $\mathcal{C}^\infty$ .

*Démonstration.* Soit  $M$  une courbe de Bézier d'ordre  $n$  paramétrée par  $\alpha$ . Alors,  $\alpha$  est définie par des fonctions polynomiales. Alors, les fonctions polynomiales étant de classe  $\mathcal{C}^\infty$  on a que  $\alpha$  l'est également.  $\square$

**Proposition I.6.** Soit  $M$  une courbe de Bézier à  $n + 1$  points de contrôle, nommés  $A_0, \dots, A_n$  et soit  $\alpha$  sa paramétrisation. On a alors :

$$\frac{d\alpha(t)}{dt} = n \sum_{i=0}^{n-1} (A_{i+1} - A_i) B_{i,n-1}(t)$$

*Démonstration.* On commence par montrer que  $(B_{i,n})'(t) = n(B_{i-1,n-1} - B_{i,n-1}(t))$ . En effet :

$$\begin{aligned} (B_{i,n})'(t) &= \binom{n}{i} t^{i-1} (1-t)^{n-i-1} (i-nt) \\ &= i \binom{n}{i} t^{i-1} (1-t)^{n-i-1} - n \binom{n}{i} t^i (1-t)^{n-i-1} \\ &= i \binom{n}{i} t^{i-1} (1-t)^{n-i-1} (1-t) + (i-n) \binom{n}{i} t^i (1-t)^{n-i-1} \\ &= n \binom{n-1}{i-1} t^{i-1} (1-t)^{n-i} - n \binom{n-1}{i} t^i (1-t)^{n-i-1} \\ &= n(B_{i-1,n-1} - B_{i,n-1}(t)) \end{aligned}$$

Alors on a :

$$\begin{aligned} \frac{d\alpha(t)}{dt} &= \sum_{i=0}^n (B_{i,n})'(t) A_i \\ &= \sum_{i=0}^n n(B_{i-1,n-1}(t) - B_{i,n-1}(t)) A_i \\ &= n \sum_{i=-1}^{n-1} B_{i,n-1}(t) A_{i+1} - n \sum_{i=0}^n B_{i,n-1}(t) A_i \\ &= 0 + n \sum_{i=0}^{n-1} B_{i,n-1}(t) (A_{i+1} - A_i) + 0 \end{aligned}$$

## I. GÉNÉRALITÉS SUR LES COURBES DE BÉZIER

---

□

**Proposition I.7.** Soit  $M$  une courbe de Bézier d'ordre  $n \in \mathbb{N}$ , contrôlée par les points  $A_0, \dots, A_n$ .

Le vecteur  $\overrightarrow{A_0A_1}$  est tangent à la courbe  $M$  au point de  $M$  de paramètre  $t = 0$ , c'est-à-dire en  $A_0$ . On a également que le vecteur  $\overrightarrow{A_{n-1}A_n}$  est tangent à la courbe  $M$  au point de paramètre  $t = 1$ , c'est-à-dire  $A_n$ .

*Démonstration.* Soit  $\alpha$  la paramétrisation d'une courbe de Bézier  $M$  contrôlée par  $n + 1$  points  $A_0, \dots, A_n$ .

D'après la proposition I.6 page 9, on a :

$$\begin{aligned}\alpha'(t) &= n \sum_{i=0}^{n-1} (A_{i+1} - A_i) B_{i,n-1}(t) \\ &= n \sum_{i=0}^{n-1} A_{i+1} B_{i,n-1}(t) - n \sum_{i=0}^{n-1} A_i B_{i,n-1}(t)\end{aligned}$$

Or, on a :  $B_{i,n}(0) = \begin{cases} 1 & \text{si } i = 0 \\ 0 & \text{sinon} \end{cases}$  et  $B_{i,n}(1) = \begin{cases} 1 & \text{si } i = n \\ 0 & \text{sinon} \end{cases}$

D'où :  $\alpha'(0) = nA_1 - nA_0 = n(A_1 - A_0)$  et  $\alpha'(1) = nA_n - nA_{n-1} = n(A_n - A_{n-1})$

On a donc bien le résultat voulu. □

**Remarque.** La proposition I.7 est une proposition importante, notamment pour les courbes de Bézier quadratiques. En effet, d'après cette propriété, il est très aisé de déterminer le troisième point de contrôle d'une courbe lorsque l'on connaît ses points de passage et l'équation de cette courbe. Il suffit alors de prendre comme point de contrôle le point d'intersection des deux tangentes.

**Proposition I.8.** Soit  $M$  une courbe de Bézier d'ordre  $n \in \mathbb{N}$ , contrôlée par les points  $A_0, \dots, A_n$ . La courbe  $M$  est alors contenue dans l'enveloppe convexe de ses points de contrôle.

*Démonstration.* On sait que la somme des polynômes de Bernstein vaut 1 d'après la proposition I.2 page 6. De plus, ces polynômes sont positifs ou nuls sur l'intervalle  $[0, 1]$ . Alors, l'enveloppe convexe des points de contrôle étant l'ensemble des barycentres à coefficients positifs de ces points, la courbe est contenue dans le polygone de contrôle, lui-même contenu dans l'enveloppe convexe. □

## I. GÉNÉRALITÉS SUR LES COURBES DE BÉZIER

---

**Proposition I.9.** Soit  $M$  une courbe de Bézier d'ordre  $n \in \mathbb{N}$  contrôlée par  $A_0, \dots, A_n$  et de paramétrisation  $\alpha$ .

Soit  $f$  une transformation affine telle que :

$$f(x) = Ax + b \quad , \quad A \in \mathcal{M}_n(\mathbb{R}) \quad , \quad b \in \mathbb{R}^2$$

Alors, appliquer la transformation  $f$  à la courbe  $M$  revient à appliquer la transformation à chaque point de contrôle.

*Démonstration.* Soient  $\alpha$  la paramétrisation d'une courbe de Bézier  $M$ , d'ordre  $n$  et contrôlée par  $A_0, \dots, A_n$  et  $f$  une transformation affine.

Alors :

$$\begin{aligned} f(\alpha(t)) &= f\left(\sum_{i=0}^n B_{i,n}(t)A_i\right) \\ &= A \times \sum_{i=0}^n (B_{i,n}(t)A_i) + b \\ &= \sum_{i=0}^n (B_{i,n}(t)A \times A_i) + b \\ &= \sum_{i=0}^n B_{i,n}(t)A \times A_i + b \sum_{i=0}^n B_{i,n}(t) \quad \text{car} \quad \sum_{i=0}^n B_{i,n}(t) = 1 \\ &= \sum_{i=0}^n B_{i,n}(t)(AA_i + b) \\ &= \sum_{i=0}^n B_{i,n}(t)f(A_i) \end{aligned}$$

Appliquer une transformation affine à la courbe  $M$  revient donc à appliquer cette transformation à chaque point de contrôle de la courbe.  $\square$

**Remarque.** Il existe également les courbes de Bézier rationnelles. Ces courbes sont contrôlées par des points pondérés  $A_i$ , de poids  $\omega_i$ . On définit alors ces courbes comme suit :

$$\frac{\sum_{i=0}^n B_{i,n}(t)\omega_i A_i}{\sum_{i=0}^n B_{i,n}(t)\omega_i}$$

## I. GÉNÉRALITÉS SUR LES COURBES DE BÉZIER

---

En pratique, elles sont utilisées pour obtenir des cercles ou des arcs de cercles. En effet, avec des courbes de Bézier simples, il est impossible d'obtenir des cercles, des arcs de cercles ou des coniques en général. C'est donc là qu'interviennent les courbes rationnelles.

### I. 3) Algorithme de De Casteljau

Nous allons maintenant nous intéresser à l'algorithme de Paul De Casteljau. Cet algorithme récursif permet d'approximer les polynômes de Bernstein avec très peu de calculs. Nous verrons quels intérêts il peut y avoir à utiliser cet algorithme. De nos jours, cet algorithme est beaucoup moins répandu qu'à ses débuts. En effet, l'algorithme permet au mieux un travail pixel par pixel et non pas un travail sur les tangentes à une courbe.

L'utilisation de cet algorithme repose principalement sur le fait que toute concaténation de courbes de Bézier reste une courbe de Bézier. Nous verrons qu'il en est de même pour toute restriction d'une courbe de Bézier. Il est donc important de vérifier quelques propriétés de concaténation et de restriction. Nous verrons par la suite une explication illustrée de cet algorithme.

#### Concaténation et restriction

**Proposition I.10.** Soient  $\beta, \gamma$  deux paramétrisations de courbes de Bézier  $B$  et  $C$  d'ordre  $n \in \mathbb{N}$ , avec  $t \in [0, 1]$ . On peut alors définir  $\alpha$ , une troisième paramétrisation d'une courbe de Bézier  $A$ , d'ordre  $2n$  comme suit :

- $\alpha(t) = \beta(2t), \quad \forall t \in [0, \frac{1}{2}]$
- $\alpha(t) = \gamma(2t - 1), \quad \forall t \in [\frac{1}{2}, 1]$

La courbe  $A$  est donc la concaténation des courbes  $B$  et  $C$ , notée  $B \vee C$ .

*Démonstration.* — Premièrement, montrons que :  $\alpha(1/2) = \beta(0) = \gamma(1)$ ,  $\alpha(0) = \beta(0)$  et  $\gamma(1) = \alpha(1)$

Soient :  $\beta(2t) = ((x(t), y(t)))$  et  $\gamma(2t - 1) = ((x(t), y(t)))$

Soit  $\alpha$  vérifiant les hypothèses de la proposition. Alors :

$$\alpha(0) = (x(0), y(0)) = \beta(0)$$

## I. GÉNÉRALITÉS SUR LES COURBES DE BÉZIER

---

$$\begin{aligned}\alpha(1/2) &= (x(1/2), y(1/2)) = \beta(2 \times 1/2) = \beta(1) \\ &= \gamma(2 \times 1/2 - 1) = \gamma(0)\end{aligned}$$

$$\alpha(1) = ((x(1), y(1))) = \gamma(2 \times 1 - 1) = \gamma(1)$$

— Deuxièmement, montrons que l'ordre de  $A$  est  $2n$ .

En effet, on a supposé  $B$  et  $C$  toutes deux d'ordre  $n$ . Chaque courbe comporte donc  $n + 1$  points.

Alors, comme la concaténation nécessite un point commun, on a que  $A$  est une courbe de  $2 \times (n + 1) - 1$  points. On a donc finalement une courbe de  $2n + 1$  points. C'est-à-dire, une courbe d'ordre  $2n$ .

□

**Remarque.** Ces concaténations sont donc continues. On peut malgré cela chercher à s'assurer une continuité  $\mathcal{C}^1$ . Pour cela, si le point commun de deux portions de courbes est le point  $A_i$ , on doit avoir  $A_i - A_{i-1} = A_{i+1} - A_i$ .

**Proposition I.11.** Soit  $M$  une courbe de Bézier d'ordre  $n$  et de paramétrisation  $\alpha$ . Soit  $N$  une restriction de la courbe  $M$  à  $[a, b]$  avec  $0 \leq a \leq b \leq 1$ , de paramétrisation  $\beta$ . Alors, la restriction  $N$  est une courbe de Bézier d'ordre  $n$ .

*Démonstration.* Soient  $\alpha$  une paramétrisation de courbe de Bézier d'ordre  $n \in \mathbb{N}$  contrôlée par les points  $A_0, \dots, A_n$ .

Soit  $\beta$  une restriction de  $\alpha$  à  $[t_1, t_2]$ , avec  $0 \leq t_1 \leq t_2 \leq 1$ .

Alors :

$$\beta = \sum_{i=0}^n A_i B_{i,n}(t) \quad , \quad t \in [t_1, t_2]$$

Donc  $\beta$  est bien la paramétrisation d'une courbe de Bézier d'ordre  $n$ .

□

### Première itération de l'algorithme

L'algorithme de De Casteljau a pour but d'augmenter le nombre de points de contrôle d'une courbe. En effet, ceci a pour intérêt de lisser le polygone formé par les points de contrôle, afin d'obtenir une approximation de la courbe de Bézier associée. La première itération a donc pour but de diviser notre polygone de base, défini par  $n + 1$  points, en deux nouveaux polygones tous deux définis par  $n + 1$  points. Ces deux polygones ayant un point commun, on obtiendra donc  $2n + 1$  points.

Pour cela, on débute avec  $n + 1$  points du plan, reliés par des segments formant un

## I. GÉNÉRALITÉS SUR LES COURBES DE BÉZIER

---

polygone (non fermé).

Soient  $(A_i)_{i \leq n}$  les points de contrôle.

On appelle alors  $(P_{0,i})_{i \leq n}$  notre nouvelle famille de points de contrôle, et on pose  $P_{0,i} = A_i$ . On a donc  $n + 1$  points  $P_{0,j}$ .

Il s'agit maintenant de créer d'autres points, afin d'obtenir nos futurs points de contrôle. On note alors ces nouveaux points  $P_{1,i}$ .

On pose donc pour cela :

$$\forall \ 0 \leq i \leq n - j, \forall \ 1 \leq j \leq n \quad : P_{j,i} = (1 - t)P_{j-1,i} + tP_{j-1,i+1}$$

On obtient donc  $\frac{n(n+1)}{2}$  nouveaux points.

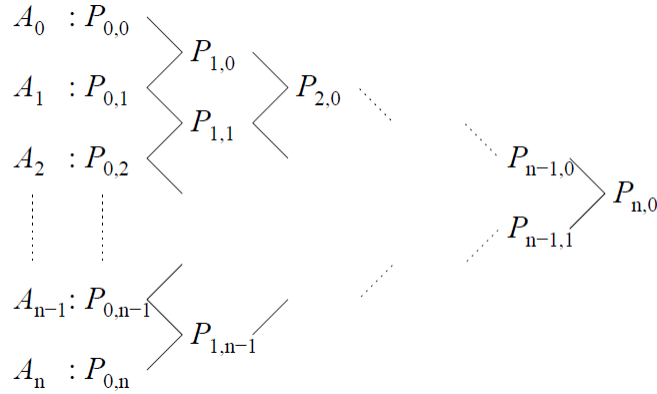


FIGURE 5 – *Diagramme représentant les points créés*

On choisit maintenant les  $2n + 1$  points.

On définit alors nos deux nouveaux polygones de contrôle comme suit :

$$B = \{P_{j,0}, 0 \leq j \leq n\} \quad C = \{P_{j,n-j}, 0 \leq j \leq n\}$$

Alors, le polygone  $B$  correspond au polygone de contrôle de notre courbe pour  $x \in [0, t]$  et le polygone  $C$  à celui pour  $x \in [t, 1]$ .

On a donc finalement obtenu deux polygones de contrôle, chacun étant composé de

## I. GÉNÉRALITÉS SUR LES COURBES DE BÉZIER

$n + 1$  points. Finalement, on a donc bien obtenu  $2n + 1$  points de contrôle.

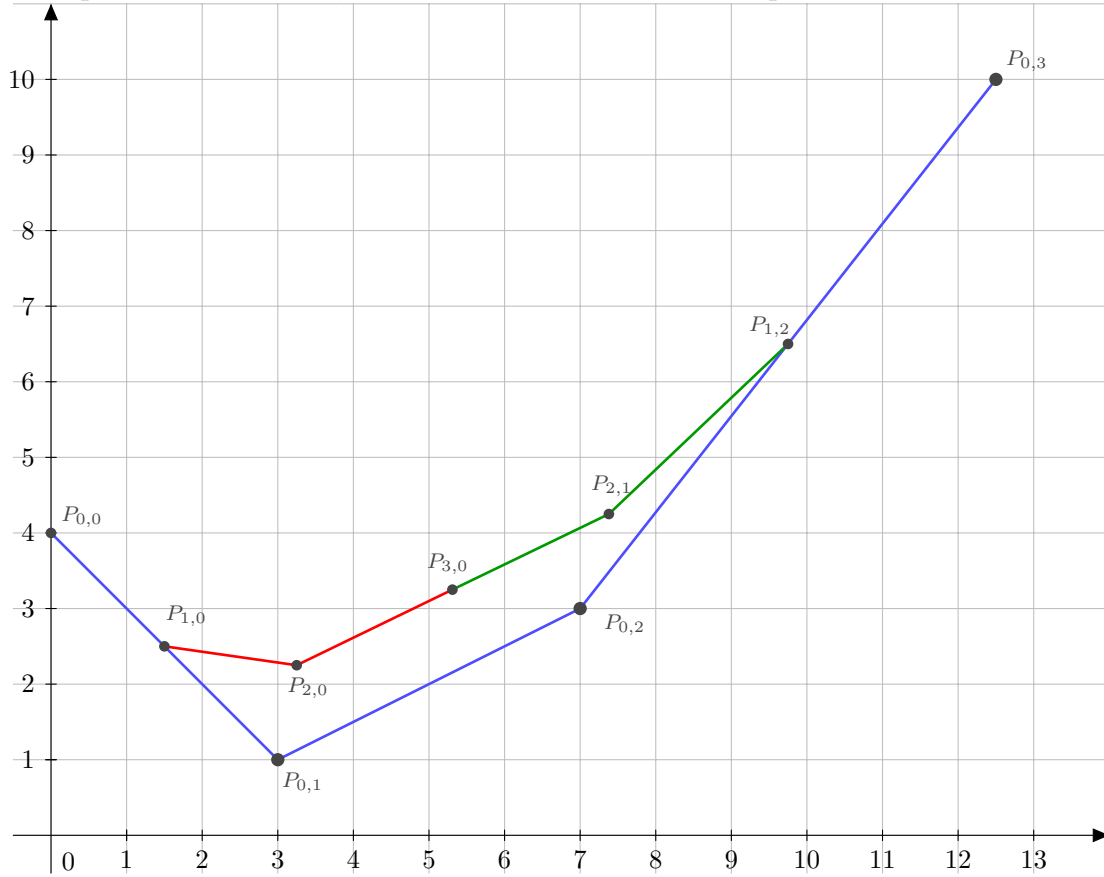


FIGURE 6 – Application manuelle de l'algorithme sous GeoGebra, avec 4 points et  $t = 0.5$ .

### Généralisation de l'algorithme

Le but est maintenant de généraliser cet algorithme. On va donc chercher à l'appliquer plusieurs fois sur un polygone de base. Cela nous permettra de nous approcher au plus près de la courbe de Bézier correspondant au polygone.

On va tout d'abord supposer que la première itération définie en page 13 a été réalisée. On suppose donc l'existence des deux polygones créés à la fin de cette première itération.

Désormais, on cherche à obtenir 2 nouveaux polygones de contrôle pour chaque polygone déjà existant. On va donc chercher à obtenir 4 polygones. Pour cela, on applique le principe précédent au premier polygone, c'est-à-dire  $B$ , puis ensuite à  $C$ .

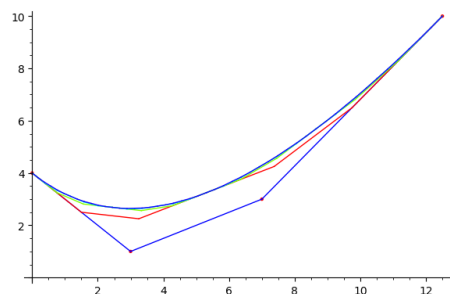


## I. GÉNÉRALITÉS SUR LES COURBES DE BÉZIER

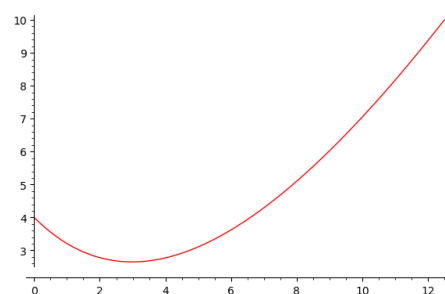
On va donc tout d'abord appliquer l'algorithme aux points  $P_{0,0}, P_{1,0}, P_{2,0}, P_{3,0}$  afin de créer deux nouveaux polygones puis aux points  $P_{3,0}, P_{2,1}, P_{1,2}, P_{0,3}$  dans le même but.

Le principe devient alors, à chaque itération, de créer deux nouveaux polygones à partir de chaque polygone existant.

On obtient à chaque itération deux fois plus de polygones qu'à l'itération précédente. Alors, on obtient donc après  $n$  itérations,  $2^n$  polygones de contrôle.



(a) Algorithme appliqué 4 fois avec SageMath, pour 4 points et  $t = 0.5$ .



(b) Courbe de Bézier correspondant aux 4 points de contrôle.

FIGURE 7 – Tracés d'une courbe avec et sans l'algorithme de De Casteljau.

**Remarque.** En général, on utilisera le cas  $t = 0.5$  pour plusieurs raisons :

- Plus simple à utiliser, que ce soit dans les calculs manuels ou informatiques ;
- Plus rapide, notamment lorsque l'algorithme est implémenté en programmation ;
- Dans la pratique, c'est le paramètre qui permet d'itérer le moins de fois pour avoir un résultat convenable. En effet, c'est le paramètre qui permet au tracé de tendre le plus rapidement vers la courbe de Bézier souhaitée.

Ainsi, nous avons donc vu différents avantages et inconvénients des courbes de Bézier. En effet, ces courbes sont faciles à manipuler (même avec un grand nombre de points) puisqu'il suffit de connaître les points de contrôle pour réaliser le tracé. Cependant, le degré d'une courbe dépend du nombre de points de contrôle et la modification d'un point entraîne une modification de la courbe dans sa majeure partie. Enfin, avec les courbes de Bézier, il est impossible d'obtenir une courbe "pointue".

Nous allons à présent aborder le thème des B-splines, qui permettront notamment de régler les inconvénients vus ci-dessus.

## II B-splines

Dans cette partie, nous allons définir rapidement les fonctions B-splines avant d'aborder quelques propriétés reliant les polynômes de Bernstein et les fonctions B-splines. Puis, nous parlerons brièvement des courbes B-splines afin de voir l'intérêt de ces dernières par rapport aux courbes de Bézier.

### II. 1) Fonctions B-splines

**Définition II.1.** Soit  $v = (v_0, v_1, \dots, v_m)$  un vecteur tel que  $v_0 \leq v_1 \leq \dots \leq v_m$ .  $v$  est appelé vecteur nodal, les  $v_i$  des nœuds.

La  $i^{\text{ème}}$  fonction B-splines de degré  $d$  est définie par récurrence par :

$$N_{i,0}(t) = \mathbb{1}_{[v_i, v_{i+1}[}(t)$$

$$N_{i,d}(t) = \frac{t - v_i}{v_{i+d} - v_i} N_{i,d-1}(t) + \frac{v_{i+d+1} - t}{v_{i+d+1} - v_{i+1}} N_{i+1,d-1}(t)$$

**Remarque.** On peut également, en gardant les notations précédentes, poser :

$$\omega_{i,d}(t) = \frac{t - v_i}{v_{i+d} - v_i}$$

Et, par convention, si  $v_i = v_{i+d}$ , on a  $\omega_{i,d} = 0$ .

On obtient alors la formule :

$$N_{i,0}(t) = \mathbb{1}_{[v_i, v_{i+1}[}(t) \tag{1}$$

$$N_{i,d}(t) = \omega_{i,d}(t) N_{i,d-1}(t) + (1 - \omega_{i+1,d}(t)) N_{i+1,d-1}(t) \tag{2}$$

Contrairement aux polynômes de Bernstein qui sont des fonctions polynomiales, les fonctions B-splines sont elles, des fonctions polynomiales par morceaux.

**Remarque.** Tout comme nous l'avions vu au sujet des polynômes de Bernstein en [I.1](#) page 4, les fonctions B-splines forment une base d'un espace vectoriel. Il s'agit ici de l'espace vectoriel des fonctions polynomiales par morceaux de degré inférieur ou égal à  $k$ , définies sur  $[v_0, v_{m-d}]$ .

**Proposition II.1.** Pour définir  $n + 1$  fonctions de degré  $d$  à partir d'un vecteur nodal de longueur  $m + 1$ , on doit avoir :

$$m = n + d + 1$$

*Démonstration.* En effet,  $N_{n,d}(t)$  a pour support  $[v_n, v_{n+d+1}]$ . Comme c'est la dernière fonction B-spline, on doit avoir  $v_{n+d+1} = v_m$ , c'est-à-dire  $n + d + 1 = m$ .  $\square$

## II. B-SPLINES

Nous allons maintenant faire le lien entre fonctions B-splines et polynômes de Bernstein. En effet, nous allons voir que les polynômes de Bernstein sont un cas très particulier de fonctions B-splines.

**Remarque.** Soit  $v = (v_0, \dots, v_{2k+1})$  un vecteur de nœud tel que l'on ait :  $v_0 = \dots = v_d = 0$  et  $v_{d+1} = \dots = v_{2d+1} = 1$ .

Alors, pour  $i$  variant de 0 à  $d$ , on a l'égalité suivante :

$$\begin{cases} N_{i,d}(t) = B_{i,d}(t) & \text{pour } t \in [0, 1] \\ N_{i,d}(t) = 0 & \text{sinon} \end{cases}$$

On peut en effet le remarquer sur la figure 8 page 18, en prenant le vecteur nodal  $v = (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1)$ .

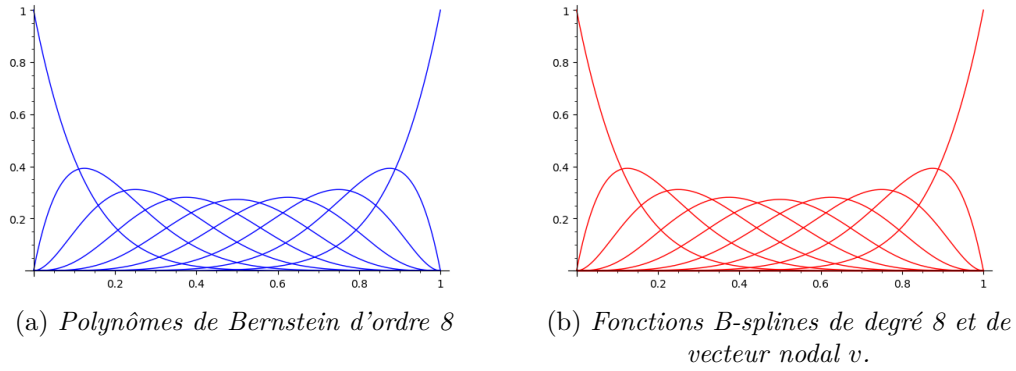


FIGURE 8 – Tracés des polynômes de Bernstein avec et sans les B-splines.

Les fonctions B-splines vérifient différentes propriétés. Nous allons les regrouper sans les démontrer dans la proposition suivante. En effet, certaines des démonstrations seront très semblables à celles vues précédemment pour les polynômes de Bernstein.

**Proposition II.2.** Soient  $n + 1$  fonctions  $N_{i,d}$  de degré  $d$  avec  $n$  et  $d \in \mathbb{N}$  et  $i \in \llbracket 0, n \rrbracket$ . Soit  $v = (v_0, \dots, v_m)$  le vecteur nodal correspondant. On a alors :

- Pour  $t \in ]v_i, v_{i+1}[$ ,  $0 \leq N_{i,d}(t) \leq 1$
- $\sum_{i=0}^{m-d-1} N_{i,d}(t) = 1$
- Pour tout  $i \leq d$ , la fonction  $B_{i,k}$  est une fonction de classe  $\mathcal{C}^\infty$  à droite de chaque point.

### II. 2) Courbes B-splines

**Définition II.2.** La courbe B-spline de degré  $d$  contrôlée par  $n + 1$  points, définie à partir d'une famille  $(N_{i,d}(t))_{i \in \llbracket 0, n \rrbracket}$  est la courbe paramétrée par :

$$P(t) = \sum_{i=0}^n N_{i,d}(t) P_i$$

Pour obtenir une courbe de degré  $d$  à partir de  $n + 1$  points, il faut se donner un vecteur nodal de longueur  $n + d + 2$ .

**Remarque.** Il existe deux conventions pour le choix du vecteur nodal.

- La première consiste à prendre un vecteur uniforme, c'est-à-dire avec le même écart entre chaque nœud, par exemple  $(0, 1, 2, 3, 4, 5, 6, 7)$ . L'inconvénient est qu'une courbe B-Spline définie à partir de ce vecteur ne passera pas par les points extrêmes, mais par l'origine.
- La seconde convention est de prendre un vecteur avec une multiplicité  $k + 1$  aux deux extrémités et uniforme au milieu, par exemple  $(0, 0, 0, 0, 1, 2, 3, 4, 4, 4, 4)$  pour un degré 3. On choisira cette méthode car elle permet de construire des courbes qui passent par les points de contrôle extrêmes.

**Remarque.** De par leur définition, un des principaux avantages des courbes B-splines par rapport aux courbes de Bézier, est leur contrôle local. Ces courbes étant définies par des fonctions polynomiales par morceaux, modifier le point  $P_i$  entraîne une modification de la courbe sur  $[v_i, v_{i+1}]$ .

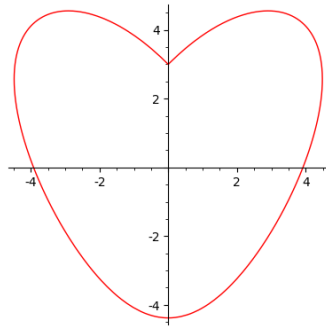
De plus, avec les B-splines, il est possible de faire des courbes non nécessairement arrondies.

La figure 9 page 20 est un exemple de comparaison entre une courbe de Bézier et une courbe B-spline. Ces tracés sont obtenus avec les mêmes points de contrôle. On utilisera le vecteur nodal  $v = (0, 0, 0, 0, 0, 1, 2, 3, 4, 5, 6, 7, 8, 8, 8, 8, 8)$  et on tracera une courbe B-spline de degré 4.

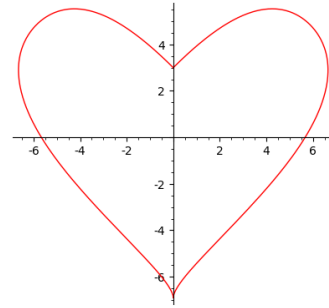
**Remarque.** Il existe aussi pour les B-splines des algorithmes permettant d'augmenter le degré d'une courbe. On trouve parmi ces derniers l'algorithme de De Casteljau ou l'algorithme de De Boor.

## II. B-SPLINES

---



(a) *Tracé avec les courbes de Bézier*



(b) *Tracé avec les spline et le vecteur nodal  $v$ .*

FIGURE 9 – *Tracés de courbes avec et sans les B-splines.*

## Conclusion

Nous avons donc vu deux méthodes différentes permettant de construire des courbes et de les modéliser. Dans un premier temps, nous avons étudié les courbes de Bézier, en particulier deux méthodes de construction. Au fil de cette étude, nous avons pu remarquer de nombreux avantages, mais également certains inconvénients. C'est pourquoi nous nous sommes penchés dans un second temps sur les courbes B-splines. Celles-ci nous ont en effet permis, tout comme les courbes de Bézier, de construire rapidement et simplement des courbes qui sont stables, faciles à manipuler et instinctives dans le positionnement des points de contrôle. Mais elles ont également d'autres avantages, notamment celui d'être contrôlées localement et ainsi d'obtenir des formes plus variées. De plus, elles sont bien moins gourmandes en temps de calcul, puisque leur degré ne dépend pas du nombre de points de contrôle.

---

## Remerciements

Nous tenons à remercier M. Chouly pour l'aide apportée tout au long du semestre. Nous le remercions pour ses conseils, son encadrement et son attention. De plus nous remercions M. Dupuis pour nous avoir initiés aux langages utilisés tout au long du projet.

Enfin, nous remercions M. Lefort pour son aide face à certaines difficultés.

# Bibliographie

- [1] Claude Basdevant. Courbes pour la CAO. <http://www.math.univ-paris13.fr/~basdevan/Licence/B-Spline.pdf>.
- [2] Thomas Guillod. Interpolations, courbes de Bézier et B-splines. <http://www.sens-neuchatel.ch/bulletin/no34/art3-34.pdf>.
- [3] Pascal Mignot. Cours de maîtrise d'informatique. <http://mathinfo.univ-reims.fr/image/siRendu/Documents/Archives/option2003-chap02-CourbesSurfaces.pdf>. Chapitre 2.
- [4] Daniel Perrin. Les courbes de Bézier. <https://www.math.u-psud.fr/~perrin/CAPES/geometrie/BezierDP.pdf>.
- [5] Nicolas Szafran. Fiches de cours. <http://www-ljk.imag.fr/membres/Nicolas.Szafran/ENSEIGNEMENT/MASTER2/CS/cours-fiches.pdf>.