

Modélisation Géométrique - TP

Splines Hermite

Exercice 1 Implémentation des splines Hermite

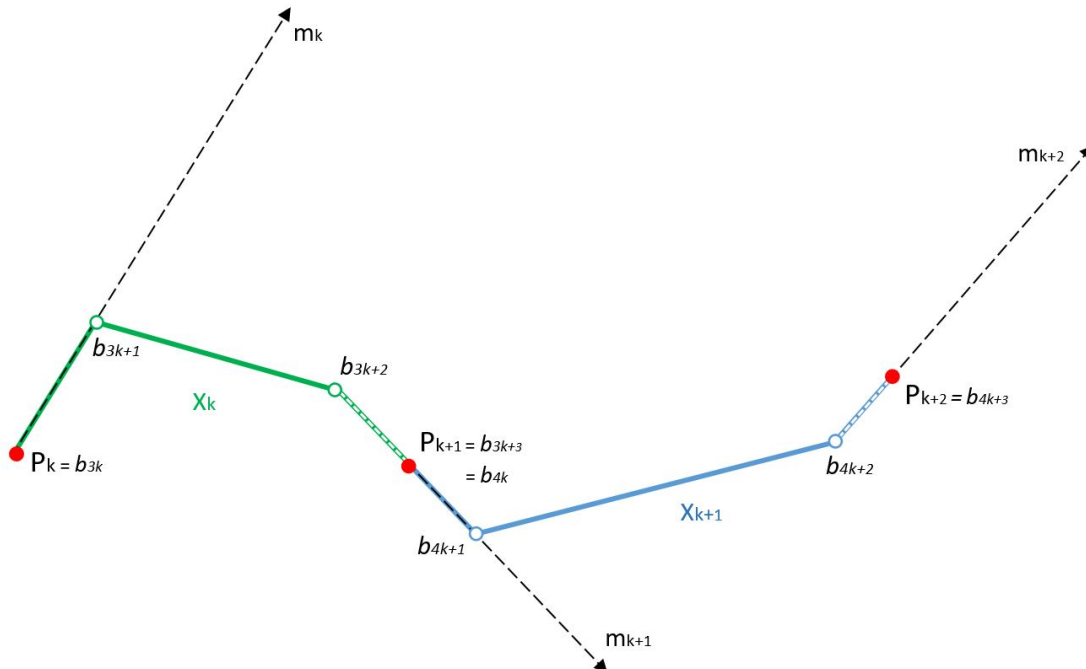
1. Pour $u \in [u_0, u_N]$ et $t \in [0, 1]$ la spline Hermite (2) est :

$$\begin{aligned}
 P(u) &= P_k H_0(t) + P_{k+1} H_1(t) + (u_{k+1} - u_k) m_k H_2(t) + (u_{k+1} - u_k) m_{k+1} H_3(t) \\
 &= P_k (B_0^3(t) + B_1^3(t)) + P_{k+1} (B_2^3(t) + B_3^3(t)) + (u_{k+1} - u_k) m_k \frac{1}{3} B_1^3(t) - (u_{k+1} - u_k) m_{k+1} \frac{1}{3} B_2^3(t) \\
 &= P_k B_0^3(t) + (P_k + (u_{k+1} - u_k) m_k \frac{1}{3}) B_1^3(t) + (P_{k+1} - (u_{k+1} - u_k) m_{k+1} \frac{1}{3}) B_2^3(t) + P_{k+1} B_3^3(t)
 \end{aligned}$$

Ainsi par identification dans la base des polynômes de Bernstein :

$$\begin{aligned}
 b_{3k} &= P_k \\
 b_{3k+1} &= P_k + (u_{k+1} - u_k) \frac{m_k}{3} \\
 &= P_k + \frac{m_k}{3} \\
 \text{car } (u_{k+1} - u_k) &= 1 \\
 b_{3k+2} &= P_{k+1} - (u_{k+1} - u_k) \frac{m_{k+1}}{3} \\
 &= P_{k+1} - \frac{m_{k+1}}{3} \\
 b_{3k+3} &= P_{k+1}
 \end{aligned}$$

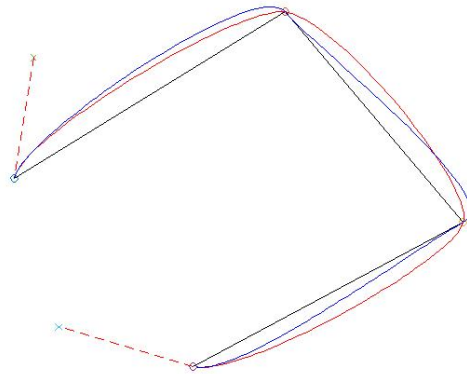
2. Pour les polygones de contrôle x_k et x_{k+1} on obtient la figure suivante :



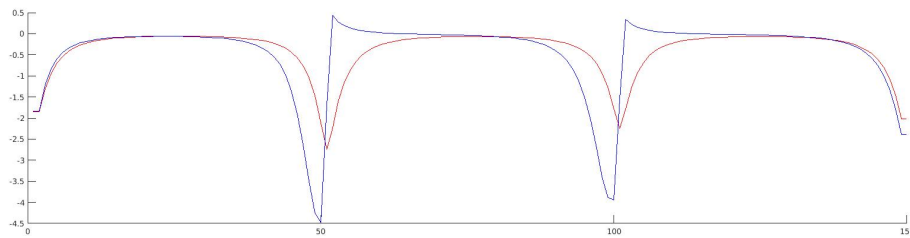
3. On peut penser à une estimation utilisant les points P_{k-1} et P_{k+1} :

$$m_k = \frac{P_{k+1} - P_{k-1}}{2(u_{k+1} - u_{k-1})}$$

Le coefficient $\frac{1}{2}$ sert seulement à réduire la norme du vecteur dérivée (sans changer sa direction). Ainsi, la courbe obtenue s'écarte moins du polygone de contrôle. En utilisant cette formule, on obtient les courbes suivantes :

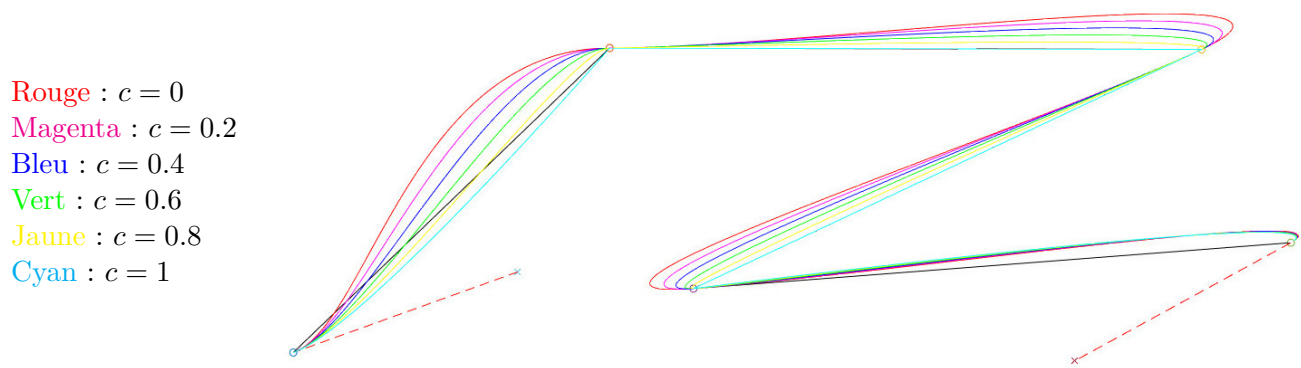


La courbe **bleue** correspond à une interpolation par la méthode des *cardinal splines* avec un coefficient c de 0.5, la courbe **rouge** est, elle, obtenue par la méthode proposée ci-dessus. On remarque que cette dernière présente une forme qui semble plus naturelle (les changements de directions sont moins brusques). Cette impression est confirmée par le graphe de courbure des deux courbes :



En effet, on peut remarquer que la courbe **bleue** présente des inversions de courbure (cette dernière passe brutalement d'une valeur très négative à une valeur positive) ce qui n'est pas le cas de la courbe **rouge**.

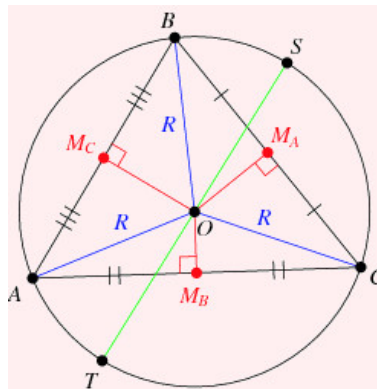
4. Notre implémentation permet le dessin de splines Hermite dont les tangentes sont estimées par la méthode des *cardinal splines*. Pour cela, lors de la création d'une spline, une boîte de dialogue permet de sélectionner le paramètre c et la couleur de la courbe. L'utilisateur entre ensuite, à la souris, le polygone de contrôle et les tangentes aux extrémités de ce dernier. L'ajout d'autres splines avec un coefficient c différent est possible à partir du menu : celles-ci seront ajoutées sur la figure une fois la boîte de dialogue (c et couleur) remplie. Voici les différentes courbes que nous avons obtenues :



D'après nos tracés, on constate que lorsque le paramètre c augmente (tout en restant dans l'intervalle $[0; 1]$), la courbe se rapproche du polygone de contrôle (on a une diminution des oscillations). Cependant, plus c augmente, plus la courbe devient irrégulière (les changements de direction deviennent plus anguleux). Pour le cas extrême $c = 1$, la courbe se confond avec le polygone de contrôle : elle n'est plus dérivable.

5. Afin de mesurer la qualité de notre courbe, nous avons choisi de représenter le graphe de courbure $\kappa(u)$. La courbure d'une courbe en un point donné est égale à l'inverse du rayon du cercle étant le plus tangent à la courbe au point considéré (On parle de cercle osculateur). Ainsi, lorsque la courbe est confondue avec une droite, la courbure est nulle (on a un cercle osculateur dégénéré de rayon infini). Au contraire, lorsque la courbe change de direction, le cercle osculateur voit son rayon diminuer ce qui induit une augmentation de la courbure. Lors de changements de direction très brusques (la courbe possède un point anguleux et donc une dérivée discontinue), la courbure devient infinie (le cercle osculateur est réduit au point de la courbe considérée, son rayon est donc nul).

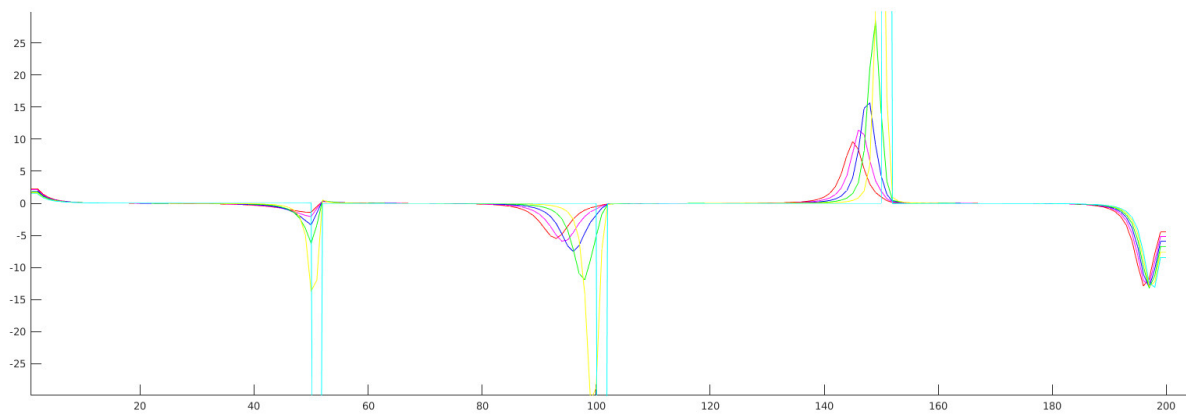
Voici comment nous calculons la courbure $\kappa(u)$:



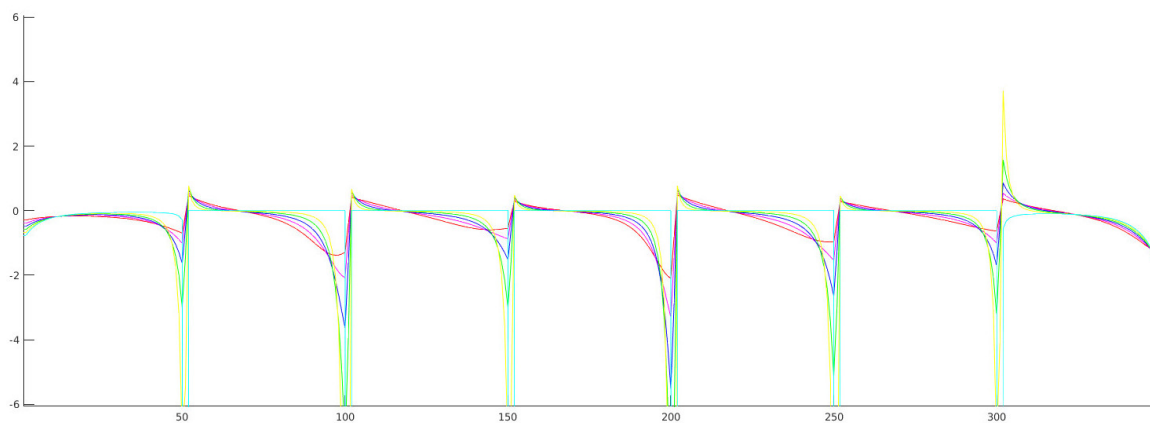
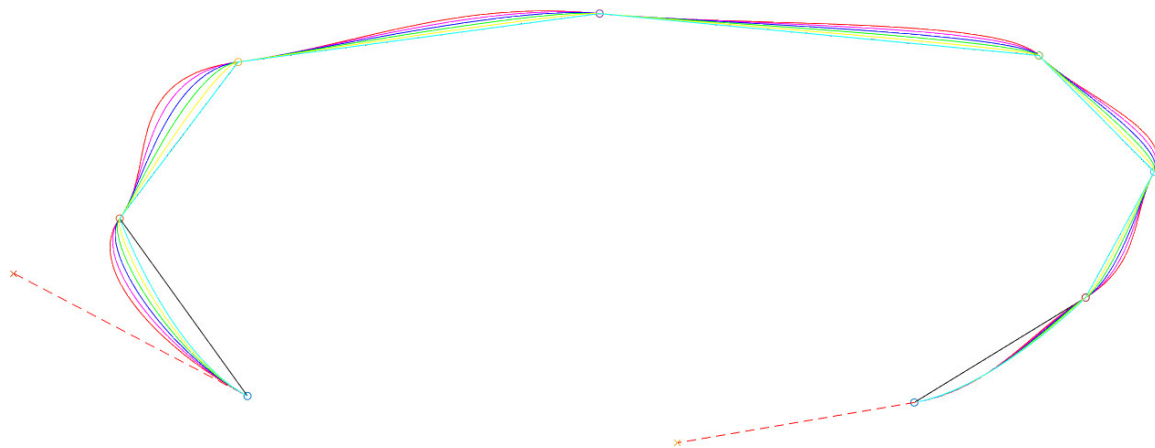
Pour calculer la courbure au point B, nous utilisons les points A et C qui sont les points voisins de B sur notre courbe. Nous obtenons ainsi un triangle dont le cercle circonscrit correspond au cercle osculateur de notre courbe au point B. Grâce aux coordonnées de ces trois points nous pouvons calculer le rayon de ce cercle et donc la courbure de notre courbe au point B (qui est égale à l'inverse de ce rayon).

Notons, qu'avec cette méthode, nous sommes obligés de considérer que la courbure au premier point de notre courbe (respectivement au dernier point) est égale à la courbure au deuxième point de notre courbe (respectivement à l'avant dernier point). En effet, il est impossible de construire un triangle pour ces deux points particuliers.

Avec les mêmes conventions de couleur pour le tracé des différentes valeurs de c , on obtient le graphe de courbure suivant (pour le polygone présenté en question 4) :

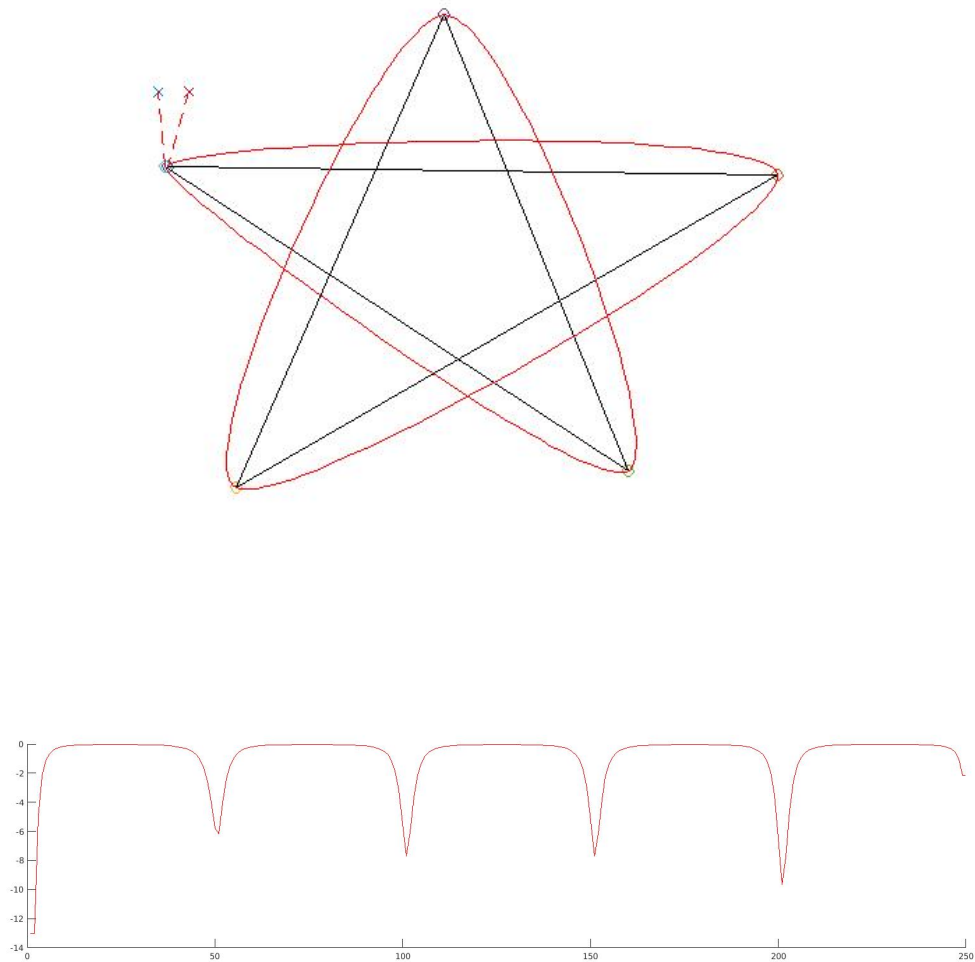


Pour un polygone présentant des angles plus ouverts on obtient les graphes suivants (avec le même code couleur) :



Le tracé du graphe de courbure peut être considéré comme un indicateur de qualité car les variations de ce dernier représentent les changements de direction des splines. Ainsi un graphe de courbure sans sauts de valeurs représentera une courbe régulière, c'est à dire une courbe continue et dérivable donc C^1 . Un saut de valeur infini sera au contraire associé un point anguleux et donc une dérivée discontinue.

6. Voici notre dessin ainsi que son graphe de courbure :



Bilan

Lors de ce projet, nous avons résolu le problème suivant : Interpoler $N + 1$ points de \mathbb{R}^2 donnés en utilisant la méthode des splines d'Hermite. Nous avons pu constater que cette méthode est efficace et permet dans le cas des cardinals splines une certaine flexibilité sur la forme de la courbe. Cependant, grâce aux tracés des graphes de courbure, nous avons noté que cette méthode avait tendance à donner des courbes présentant de brusques changements de direction pouvant aller jusqu'à l'apparition de points anguleux. Néanmoins, la méthode d'interpolation que nous avons proposé à la question 3 corrige en partie ce défaut.