

Courbes de Bézier - Approximation

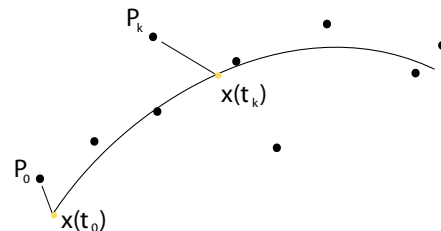
Projet noté

Approximation d'un nuage de points par une courbe de Bézier

Soient $N+1$ points $P_0, \dots, P_N \in \mathbb{R}^2$ et $N+1$ paramètres correspondants $t_0 \leq t_1 \leq \dots \leq t_N$, $t_i \in [a, b]$ (à vous de faire ce choix de la paramétrisation).

On cherche la courbe de Bézier $x(t) = \sum_{i=0}^n b_i B_i^n(t)$, $t \in [a, b]$ de degré n qui approxime au sens des moindres carrés cet ensemble de $N+1$ points.

$$\sum_{k=0}^N \|P_k - x(t_k)\|^2 \rightarrow \min.$$



Choix de la paramétrisation

voir TD-2 en salle machine.

Travail demandé

Implémentation de la méthode d'approximation (la correction de paramètres est optionnel mais un plus). Il est en plus attendu que vous ayez un regard critique par rapport aux données d'entrée et à la qualité des résultats obtenus.

1. Formulez mathématiquement la solution du problème.
2. Utilisez votre programme du TD-1 *Courbe_de_Bezier* (p.ex. le programme MATLAB) pour programmer la solution. On entre à la souris les points P_i , on choisit la paramétrisation et le degré n de la courbe x . Deux nouveaux boutons servent ensuite à exécuter l'approximation équidistante et chordale sur le *même* ensemble de points P_i donné initialement.
3. En plus: Faites de votre programme un outil qui facilite *l'interaction* et *l'analyse* des résultats (inventez des outils p.ex.). A vous d'imaginer des fonctionnalités supplémentaires de votre programme. P.ex. on doit pouvoir afficher deux courbes simultanément si elles approximent le même ensemble de points, mais possèdent une paramétrisation différente. Choix des couleurs.... Visualisez l'erreur commise, ...
4. Permettez à l'utilisateur de varier la valeur de n , le degré de la courbe.

Votre rapport

Rediger un rapport contenant

- la formulation mathématique du problème et la solution, les résultats obtenus, une analyse des deux algorithmes, et vos observations.
- un grand nombre d'illustrations et d'exemples !!

Vous pouvez vous poser un certain nombre de questions et tenter d'y répondre, comme p.ex.

- Comment la valeur de n influence t'elle le résultat? Peut-on fixer une valeur "optimale" par défaut? si oui, laquelle?
- Peut-on comparer les solutions? Avec quels critères?
- Y a t'il des ensembles de données pour lesquelles l'une ou l'autre méthode marche bien ou mieux?

Soutenance: s'inscrire sur TEIDE.

Dépot sur TEIDE avant la soutenance: rapport en PDF, toutes vos sources en .zip.

En option

Il existe un algorithme qui permet d'améliorer la qualité de l'approximation.

Algorithme de correction des paramètres:

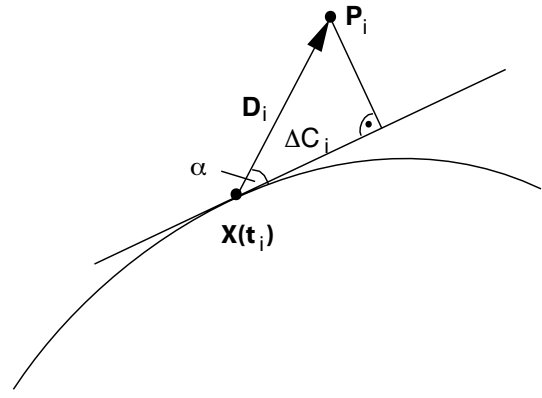
Soit $D_k := P_k - x(t_k)$.

Input: $P_k, k = 0, \dots, N$.

Output: $x(t), t \in [a, b]$.

Procédure:

0. Choix des valeurs de paramètres t_k ,
valeur limite L (du nombre d'itérations),
erreur ε ,
 $j=0$.
1. Calculer μ (longueur du polygone $\{P_k\}$)
2. Déterminer la courbe de Bézier $x(t)$ approximant par la méthode des moindres carrés.
3. FOR $i=0$ TO N DO
4. Calculer $\Delta C_i = \frac{\langle D_i, x'(t_i) \rangle}{\|x'(t_i)\|}$ /* approximation du changement de paramètre nécessaire */
5. Calculer $t_i^* = t_i + \Delta C_i \frac{b-a}{\mu}$ /* nouveau paramètre */
Calculer $\|D_i\|$ et $\|D_i^*\|$
IF $\|D_i\| > \|D_i^*\|$ ($t_i := t_i^*$ AND GOTO 6)
ELSE $\Delta C_i = \Delta C_i / 2$, et goto 5.
6. $\alpha_i = \arccos\left(\frac{\langle D_i^*, x'(t_i^*) \rangle}{\|D_i^*\| \|x'(t_i^*)\|}\right)$
END FOR
7. Calculer $\alpha = \max(\alpha_i)$
 $j = j+1$
IF $|\pi/2 - \alpha| < \varepsilon$
THEN calculer la courbe de Bézier approximant pour les derniers paramètres t_i et
stop
ELSE IF $j < L$ GOTO 2.



Attention ! vérifiez toujours que $t_i^* \in [a, b]$ et que $t_i \leq t_{i+1}$.