NMB - Oefenzitting 4
Splines en Geometrische Modellering

Simon Telen

Opgave 1. Deze opgave maak je met behulp van een java-pakket dat de theorie van de cursus illustreert. Het pakket is te vinden op Toledo (`cagd.jar`).

Start het programma op met

```
java -jar cagd.jar
```

Controleer de volgende eigenschappen uit de cursus:

a) Interpolerende veeltermcurven: $\vec{x}(u) = \sum_{i=0}^{n} \vec{p}_i L_i^n(u)$

- De curve zal doorgaans sterk oscilleren (zeker wanneer n stijgt). Het is zeer moeilijk om zachtverlopende curven te bekomen.
- Een kleine wijziging van één van de vectoren $\vec{p}_i$ zal vaak grote veranderingen veroorzaken in de ligging van de curve.

b) Bézier-curven: $\vec{x}(t) = \sum_{i=0}^{n} \vec{b}_i B_i^n(t)$

- De curve ligt binnen de convex omhullende van de Bézier-punten $\{\vec{b}_0, \vec{b}_1, \ldots, \vec{b}_n\}$.
- De Bézier-curve interpoleert in begin- en eindpunt: $\vec{x}(0) = \vec{b}_0$ en $\vec{x}(1) = \vec{b}_n$ en raakt er aan $\vec{b}_1 - \vec{b}_0$ respectievelijk aan $\vec{b}_n - \vec{b}_{n-1}$.
- De curve $\vec{x}(t)$ ligt achtereenvolgens dicht bij $\vec{b}_0, \vec{b}_1, \ldots, \vec{b}_n$, maar elke $\vec{x}(t), 0 < t < 1$ is afhankelijk van alle $\vec{b}_i, i = 0, \ldots, n$.
- De curve zal een zachtverlopend (niet sterk oscillerend) karakter hebben.
- Bij herhaalde toepassing van subdivisie convergeren de samengevoegde controleveelhoeken naar de curve $\vec{x}(t)$.

c) Splinecurven: $\vec{s}(u) = \sum_{i=-k}^{n-1} \vec{d}_i N_{i,k+1}(u)$

- Bij wijziging van één de Boor-punt $\vec{d}_j$ zal slechts een deel van de splinecurve wijzigen (*lokale* afhankelijkheid).
- Elk punt van een splinecurve van graad k ligt in de convex omhullende van $k + 1$ de Boor-punten.

- Splinecurven van lagere graad sluiten dichter aan bij de controleveelhoek.
- Gesloten curven kunnen op een eenvoudige manier worden voorgesteld.
- Samenvallende knooppunten kunnen gebruikt worden voor het verminderen van de continuïteitseigenschappen van de curve.
- Toevoegen van knooppunten kan gebruikt worden om het deel van de curve beïnvloed bij wijziging van een controlepunt te verkleinen (voor het aanbrengen van lokale wijzigingen).

Opgave 2. In deze opgave maken we gebruik van splines om een discrete verzameling van punten te interpoleren. De interpolerende kubische spline $s(x)$ wordt gedefineerd door middel van $n+1$ abscissen $\{x_i\}_{i=0}^n$, met bijhorende functiewaarden $\{f_i\}_{i=0}^n$. De interpolerende spline voldoet aan de voorwaarden

$$s(x_i) = f_i \text{ voor } i = 0, 1, \dots, n. \tag{1}$$

In geval van een kubische, *natuurlijke* spline wordt dit aangevuld met de volgende randvoorwaarden:

$$s''(x_0) = s''(x_n) = 0. \tag{2}$$

Per definitie geldt

$$s(x) = p_i(x), \quad x \in [x_{i-1}, x_i]$$

waarbij $p_i(x)$ een veelterm van graad 3 is. We gebruiken de notatie $\Delta x_i = x_i - x_{i-1}$ en $\Delta f_i = f_i - f_{i-1}$. Continuïteit van de tweede afgeleide impliceert dat

$$p_i''(x) = \frac{(x - x_{i-1})s''(x_i) - (x - x_i)s''(x_{i-1})}{\Delta x_i}.$$

Tweemaal integreren en invullen van de interpolatievoorwaarden (1) voor x_{i-1}, x_i geeft

$$p_i(x) = A_i(x) + s''(x_i)B_i(x) + s''(x_{i-1})C_i(x) \tag{3}$$

waarbij

$$\begin{aligned}
A_i(x) &= \frac{f_i(x - x_{i-1}) + f_{i-1}(x_i - x)}{\Delta x_i}, \\
B_i(x) &= \frac{1}{6}\left(\frac{(x - x_{i-1})^3}{\Delta x_i} - \Delta x_i(x - x_{i-1})\right), \\
C_i(x) &= -\frac{1}{6}\left(\frac{(x - x_i)^3}{\Delta x_i} - \Delta x_i(x - x_i)\right).
\end{aligned}$$

De overblijvende onbekenden in (3) zijn de waarden van de tweede afgeleide $s''(x)$ in de abscissen. Continuïteit van de eerste afgeleide, tesamen met de natuurlijke randvoorwaarden

(2) geeft het stelsel

$$M \begin{bmatrix} s''(x_1) \\ s''(x_2) \\ \vdots \\ s''(x_{n-1}) \end{bmatrix} = 6 \begin{bmatrix} \Delta f_2/\Delta x_2 - \Delta f_1/\Delta x_1 \\ \Delta f_3/\Delta x_3 - \Delta f_2/\Delta x_2 \\ \vdots \\ \Delta f_n/\Delta x_n - \Delta f_{n-1}/\Delta x_{n-1} \end{bmatrix}$$

met

$$M = \begin{bmatrix} 2(\Delta x_1 + \Delta x_2) & \Delta x_2 & & & \\ \Delta x_2 & 2(\Delta x_2 + \Delta x_3) & \Delta x_3 & & \\ & \Delta x_3 & 2(\Delta x_3 + \Delta x_4) & \Delta x_4 & \\ & & & \ddots & \\ & & & \Delta x_{n-1} & 2(\Delta x_{n-1} + \Delta x_n) \end{bmatrix}.$$

Met deze gegevens kunnen we $s(x)$ evalueren als een stuksgewijze veelterm in het interval $[x_0, x_n]$.

a) Schrijf een Matlab-functie voor het opstellen en evalueren van natuurlijke kubische splinefuncties.

```
function y = naturalspline(x,f,t)
```

De functie krijgt als invoer een vector `x` van lengte $n+1$ die de abscissen bevat, een vector `f` van lengte $n+1$ met functiewaarden en een vector `t` van lengte N die de punten bevat waarin we de splinefunctie willen evalueren. Als uitvoer geeft de functie de vector `y` van lengte N terug met de functiewaarden van de opgestelde spline, in de punten `t`.

b) Gebruik je code om de functies $\cos(x)$ en de Rungefunctie $\frac{1}{1+6x^2}$ te benaderen op het interval $[-2, 2]$. Gebruik een equidistante set van N absciswaarden. Vergelijk met een veelterminterpolant door dezelfde punten. Hint: je kan de Matlabfunctie `vander` gebruiken om de Vandermonde matrix op te stellen.

c) Gebruik daarna de code om een kromme te benaderen. Je kan hiervoor gebruik maken van de functie `click`: deze functie geeft de coördinaten van een willekeurig stel punten die je met de muis kan aanklikken. Teken een Griekse letter naar keuze (maar kies er wel een die niet in expliciete vorm $y = f(x)$ kan voorgesteld worden). Vergelijk het resultaat met de kleinste kwadraten en interpolerende veelterm met behulp van je code uit oefenzitting 2.

d) Gebruik interpolerende splines om je voornaam te schrijven.