Numerical Algorithms

Winter semester 2018/2019
Prof. Dr. Marc Alexander Schweitzer
Denis Duesseldorf

Exercise sheet 10. Submission on **Tuesday, 2019-01-15, after the lecture**.

Exercise 31. (B-spline derivatives)

We have already looked into B-splines but not yet at their derivatives.

a) Show that

$$\frac{\mathrm{d}}{\mathrm{d}\xi}N_{i,p}(\xi) = p\left(\frac{N_{i,p-1}(\xi)}{\xi_{i+p}-\xi_i} - \frac{N_{i+1,p-1}(\xi)}{\xi_{i+p+1}-\xi_{i+1}}\right) .$$

Hint: Recursion formula and induction starting with $p = 1$.

b) Using a), compute the derivative of the B-spline curve

$$\frac{\mathrm{d}}{\mathrm{d}\xi}C(\xi) = \frac{\mathrm{d}}{\mathrm{d}\xi}\sum_{i=1}^{n} N_{i,p}(\xi)B_i .$$

(2 Points)

Exercise 32. (Minimal degree condition)

A not yet posed or answered is how p_T, p_e should be related in a higher order FEM with V-E-C shape functions. One strategy for this is the minimal degree condition.
Let T be an element of some triangulation τ of the domain Ω for some PDE. For each edge e of the triangulation it should hold that

$$p_e = \min\{p_T | e \text{ is edge of } T\} .$$

For the depicted mesh, choose the degrees p_e of each edge e such that the minimal degree condition is fulfilled

$p = 1$	$p = 3$	$p = 1$
$p = 2$	$p = 4$	$p = 2$
$p = 1$	$p = 3$	$p = 1$

(2 Points)

Exercise 33. (sup − inf and n-width)

In the lecture the notion of classification of best approximation of a set $Y \subset X$ with some normed space $(X, \|\cdot\|_X)$ was introduced with the Kolmogorov n-width

$$d_n := \inf_{E_n \subset X, \dim E_n \leq n} \sup_{u \in Y} \inf_{v_n \in E_n} \|u - v_n\|_X \ .$$

If $X, (\cdot,\cdot)_X$ is a Hilbert space and $Y = TH$ with $H, (\cdot,\cdot)_H$ being another Hilbert space that can be compactly embedded in Y via the operator T

$$H \overset{C}{\hookrightarrow} X \ , \qquad T : H \to X \ , \qquad \|Tx\|_X \leq C\|x\|_H \ , \qquad T \text{ compact} \ .$$

these quantity allows for a more practical expression. With H being a linear space it is reasonable to exclude multiplicative constants leading to

$$\begin{aligned} Y &:= \{x \in H : \|x\|_H = 1\} = \{Tx : x \in H, \|x\|_H = 1\} \subset X \ , \\ \Psi(E_n) &:= \sup_{u \in Y} \inf_{v_n \in E_n} \|u - v_n\|_X \ , \\ d_n &= \inf_{E_n \subset X, \dim E_n \leq n} \Psi(E_n) \ . \end{aligned}$$

Ψ is called a sup − inf of E_n with respect to the $\|\cdot\|_X$ approximation of Y. For convenience the embedding T is dropped from notation.
In this special case of a compact embedding these quantities are related to the generalized eigenvalue problem

$$(u_k, v)_X = \lambda_k (u_k, v)_H \ , \qquad \forall v \in H$$

with eigenpairs $(\lambda_k, u_k), k = 0, \dots$ and eigenvalues

$$\lambda_0 \geq \lambda_1 \geq \lambda_2 \ldots > 0 \ , \qquad d_n = \sqrt{\lambda_n}$$

There exists a similar generalized eigenvalue problem for the computation of $\Psi(E_n)$.

a) Show that for $u \in H$, $E_n \subset X$ and $\dim E_n = n$ we have

$$\inf_{v_n \in E_n} \|u - v_n\|_X \leq \|u\|_H \Psi(E_n)$$

b) Assume that $H \subset Z$ with Z being another Hilbert space with a basis $\varphi_i, i \in I = \{1, \dots, N\}$. Using the coefficients of a H basis with respect to φ_i, give the matrix representation of the generalized eigenvalue problem for the computation of d_n.

(4 Points)

Exercise 34. (Moving Least Squares)

Given data $x_i, f_i, i = 1, \dots, N$ and an approximation space $P = \mathrm{span}\langle\varphi_i\rangle$ the natural extension of least squares is moving least squares (MLS) approximation.

For a locally supported non-negative function W, often referred to as *window function* or *weight function*, the pointwise *moving least squares energy* of an approximation π is given as

$$J_x(\pi) = \sum_{i=1}^{N} W(x - x_i)(f_i - \pi(\xi))^2 = \sum_{W(x-x_i)>0} W(x - x_i)(f_i - \pi(x_i))^2 \; .$$

The MLS approximation π of the data x_i, f_i is given pointwise via the minimizers $\pi_x \in P$ of J_x

$$\pi(x) = \pi_x(x) \; .$$

Using the basis φ_i and a representation $\pi_x = \sum_i u_{x,i}\varphi_i$, $u_x = (u_{x,i})_i$ we can compute the solutions π_x as

$$\begin{aligned} G_x u_x &= f_x \; , \\ (G_x)_{k,l} &:= \sum_{W(x-x_i)>0} \varphi_k(x_i) W(x - x_i) \varphi_l(x_i) \; , \\ f_x &:= \sum_{W(x-x_i)>0} f_i W(x - x_i) \varphi_l(x_i) \; . \end{aligned}$$

a) Expand and compute

$$\frac{\mathrm{d}}{\mathrm{d}x}\pi(x) \; .$$

b) The case $P = \mathrm{span}\langle 1\rangle$ is called *Shepard* approximation. For this case, compute π, $\frac{\mathrm{d}}{\mathrm{d}x}\pi$.

(4 Points)

Programming exercise 11. (Refined Spinoraptor)

Consider the B-Spline curve defined through the knot vector and control points given in programming exercise 10. Since the resolution of the model is rough, a further analysis requires the refinement of the model.

The analogue of *h-refinement* from finite elements for B-Splines (and B-Spline curves and surfaces and volumes) is *knot* insertion. Similarly to h-refinement, this can be done without changing the curve geometrically or parametrically.

Given a knot vector

$$\Xi = (\xi_1, \ldots, \xi_{n+p+1})$$

and let $\hat{\xi} \in [\xi_k, \xi_{k+1})$ be a desired new knot. The $n+1$ basis functions $\hat{N}_{i,p}$ are formed recursively using the new knot vector

$$\hat{\Xi} = (\xi_1, \ldots, \xi_k, \hat{\xi}, \xi_{k+1}, \ldots, \xi_{n+p+1})$$

of length $n+p+2$.

The new $n+1$ control points $\hat{B}_i, 1 \leq i \leq n+1$ are formed from the original control points $B_i, 1 \leq i \leq n$ by

$$\hat{B}_i = \alpha_i B_i + (1 - \alpha_i) B_{i-1} \ , \qquad \alpha_i = \begin{cases} 1 \ , & 1 \leq i \leq k-p \ , \\ \frac{\hat{\xi} - \xi_i}{\xi_{i+p} - \xi_i} \ , & k-p+1 \leq i \leq k \ , \\ 0 \ , & k+1 \leq i \leq n+p+2 \ . \end{cases}$$

a) Write a function using the described algorithm for insertion of valid knots. The function should compute the modified control points accordingly.

b) Using appropriate knots and control points, show an example of internal knots appearing more than p times leading to discontinuous functions.

c) Compute the midpoints of the unique knots

$$\eta_j = \frac{1}{2}(\xi_i + \xi_{i+1}) \ , \qquad \xi_i < \xi_{i+1} \ .$$

d) Use a) to insert the knots η_j into the knot vector from programming exercise 10.

e) Use the modified knots and control points from d) to plot $\hat{N}_{i,p}$.

f) Use the modified knots and control points from d) to plot the corresponding B-Spline curve.

(4 Points)

Send to `duesseldorf@ins.uni-bonn.de`