Algorithms of Scientific Computing (Algorithmen des Wissenschaftlichen Rechnens)

Generating Systems

A useful tool when dealing with hierarchical bases is the representation through *generating systems*. We will see how this works for the onedimensional case.

Consider the space V_n of piecewise linear, continuous functions $u : [0,1] \to \mathbb{R}, u(0) = u(1) = 0$ living on a regular grid with mesh width $h_n = 2^{-n}$.

$$\tilde{\Psi}_n := \bigcup_{l=1}^{n} \{\phi_{l,i} : 1 \leq i < 2^l\}$$

is not a basis for $n > 1$, yet it is a generating system: for every $u \in V_n$ one can find a (not necessarily unique) representation as a linear combination

$$u = \sum_{l=1}^{n} \tilde{w}_l = \sum_{l=1}^{n} \sum_{1 \leq i < 2^l} v_{l,i} \phi_{l,i}\,, \qquad \text{for } \tilde{w}_l \in V_l.$$

Write the coefficients $v_{l,i}$ in ascending order with respect to l first and i then. The resulting vector shall further be referenced by $\vec{v}^E$.

Outline algorithms that for given vector $\vec{v}^E$ compute representations of the same function in the generating system $\tilde{\Psi}_n$

- using the nodal basis: $\vec{v}^{E,N}$ with $v_{l,i} = 0$ for $l < n$
- using the hierarchical basis: $\vec{v}^{E,H}$ with $v_{l,i} = 0$ for even i.

Now back to Finite Elements! Consider the mass matrix (coefficient matrix B for L_2 scalar product) — but this also works for other bilinear forms.

For now we assemble B^E naively, i.e. we ignore the fact that we actually do not have a basis anymore and just compute the scalar products of the ansatz functions.

(i) Show that the result of the product $B^E \vec{v}^E$ is the same for all possible representations of function u in the generating system.

(ii) Knowing the coefficients of $B^E \vec{v}^E$ for level l, how can you compute the coefficients for level $l-1$ without recomputing the scalar product or B^E itself?

(iii) How can you get from B^E to a representation of the mass matrix $B^N \in \mathbb{R}^{(2^n-1)\times(2^n-1)}$ in the nodal basis?

(iv) Find matrices S_1, S_2 such that
$$B^E = S_1 B^N S_2.$$

(v) Is it possible to get a finite element approximation through a linear system of equations with B^E being the coefficient matrix?