Scientific Computing II

Summer Semester 2014
Lecturer: Prof. Dr. Beuchler
Assistent: Bastian Bohn

Excercise sheet 7.

Closing date **27.05.2014**.

Theoretical exercise 1. (Multilevel Schwarz projector [10 points])

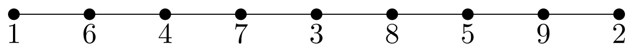
Build a preconditioner for the system of linear equations corresponding to the model Poisson problem on $[0,1]^d$ for Neumann boundary conditions with $d = 1, 2$ discretized by linear FE (piecewise linear nodal basis).

To this end, let

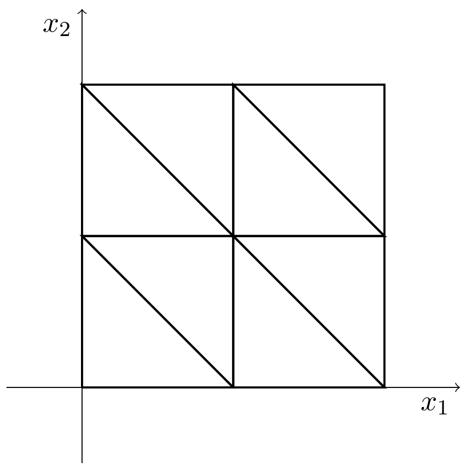
$$\mathbb{V}_i^l := \operatorname{span}\{\phi_i^l\} \text{ and } \mathbb{V}^l := \operatorname{span}\{\phi_i^l\}_{i=1}^{N_l}$$

where $N_l = (2^l + 1)^d$ for $l > 0$, ϕ_i^l is a piecewise linear nodal basis function and $\mathbb{V}^l$ is the linear FE space obtained from an l-times uniformly refined mesh corresponding to $\mathbb{V}^0$, $l = 0, \ldots, L$.

The numbering of the basis functions in each $\mathbb{V}^l$ is from left to right for the 1-dimensional case and lexicographically for the 2-dimensional case. When numbering nodes on different levels, the nodes on the coarse levels come first, here is an 1d example for the numbering of the basis functions (corresponding to their nodes) for the hierarchical basis of level 3:



In one dimension on level 1 you start with the three nodal basis functions settled in the points $0, \frac{1}{2}, 1$. In two dimensions on level 1 you start with the grid



with its corresponding nodal basis functions.

Let $\mathcal{B}_l$ be the set of nodes which are new on level l.

a) For $d = 1$ derive the additive Schwarz projector of the two-level decomposition

$$\mathbb{V} = \sum_{l=1}^{2} \sum_{i=1}^{N_l} \mathbb{V}_i^l$$

and explicitely calculate its matrix representation with respect to the basis on level 2.

b) For both cases $d = 1$ and $d = 2$ derive the additive Schwarz projector of the multi-level decomposition

$$\mathbb{V} = \sum_{l=0}^{L} \sum_{i=1}^{N_l} \mathbb{V}_i^l$$

and calculate its matrix representation with respect to the basis on level L as a sum of products of matrices.

Hint: Use prolongation operators from level l to $l-1$.

c) For $d = 1$ derive the additive Schwarz projector of the two-level decomposition

$$\mathbb{V} = \sum_{l=1}^{2} \sum_{i \in \mathcal{B}_l} \mathbb{V}_i^l$$

and explicetely calculate its matrix representation with respect to the basis on level 2.

d) For both cases $d = 1$ and $d = 2$ derive the additive Schwarz projector of the multi-level decomposition

$$\mathbb{V} = \sum_{l=0}^{L} \sum_{i \in \mathcal{B}_l} \mathbb{V}_i^l$$

and calculate its matrix representation with respect to the basis on level L as a sum of product of matrices.

Hint: Use prolongation operators from level l to $l-1$.

Programming exercise 1. (Multiplicative Schwarz preconditioner [15 points + Bonus: 10 points])

The closing date for submission of the programming exercise is **June, 21st**. Please mail your commented and compilable code to `bohn@ins.uni-bonn.de`. Points are given for correctness, readability and runtime complexity.

Implement the multilevel decomposition [**Bonus**: 5 points] and the hierarchical basis decomposition as a preconditioner for CG to solve a two-dimensional finite element system

$$K_h u = f_h \tag{1}$$

in the corresponding basis, where K_h is the stiffness matrix and f_h the right hand side for the Poisson problem with Neumann boundary conditions.

To apply a preconditioner to the residual, you will need to implement the action of the corresponding Schwarz projector on the error, i.e.

$$\begin{aligned}
&\text{For } j = 1, \ldots, m \text{ do:} && \\
&\qquad \text{Set } R_{j-1} &=&\; V_j^T r_{j-1} \\
&\qquad \text{Solve } (V_j^T K_h V_j) w_j &=&\; R_{j-1} \\
&\qquad \text{Set } u_i &=&\; u_{i-1} + V_i w_i \\
&\qquad \text{Set } r_i &=&\; K_h u_i - f \\
&\qquad \text{end do} &&
\end{aligned}$$

where u_{-1} is the latest approximative solution to (1) and r_{-1} the corresponding residual. Note that the numbering of the basis functions should be as in the theoretical exercise! The system that has to be solved has size 1×1. It does not have to be solved exactly, it suffices to do an approximation to the system.
The resulting r_m is then the result vector. Here, m is the number of subspaces in the Schwarz splitting

$$\mathbb{V} = \sum_{i=1}^{m} \mathbb{V}_i$$

and V_i is defined such that $\Phi_i = \Phi \cdot V_i$, where Φ_i is the basis of $\mathbb{V}_i$ and Φ is the basis of $\mathbb{V}$.
This setting has to be transferred to the multilevel and hierarchical decomposition: Given an $L+1$ times refined mesh you have a hierarchy of linear Lagrangian FE-bases

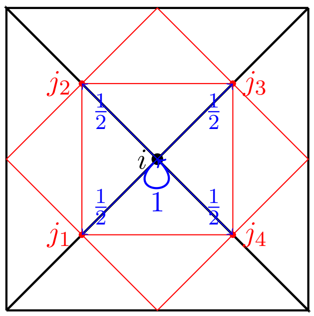
$$\Phi_0 = (\phi_1^0, \dots, \phi_{N_0}^0), \dots, \Phi_L = (\phi_1^L, \dots, \phi_{N_L}^L).$$

Let $\mathbb{V}^l := \operatorname{span}\{\Phi_l\}$ and $\mathbb{V}_i^l := \operatorname{span}\{\phi_i^l\}$, then the multilevel decomposition is given by

$$\mathbb{V}^L = \sum_{l=0}^{L} \sum_{i=1}^{N_l} \mathbb{V}_i^l,$$

see also Theoretical Exercise 1b). The hierarchical basis is given analogously, see 1d).
Hints:

- You will need to implement the application of the prolongation operator $T^{l \to l+1}$ to interpolate from level l to $l+1$. Do this in the following way: The coefficient c of the basis function ϕ settled in node i on level l will be added to the coefficient of the basis function in that same node on the next level. Furthermore for every new node on an edge that contains i the value of $\frac{1}{2}c$ will be added to these basis functions. Here is an example:

  Assume the large black triangles were (red) refined and the red triangles are the result. Then the prolongation operator $T^{l \to l+1}$ distributes the value of the basis function in the node i on level l to the basis functions in the node i on level $l+1$ (with the weight 1) and to the (new) basis functions in the nodes $j_1, \dots, j_4$ (with the weights $\frac{1}{2}$).

  Note, that the information about father/son-relations between different nodes is implicitely contained in the edge-hierarchy after refinement.

- You should give each node (basis function) the additional information on what level l it first appeared.

- In case you need a function to query the edges of a certain node or the elements of a certain node, feel free to implement it.

Test your algorithm:

Solve the Poisson problem with right hand side 0 on the unit cube (`SampleGrid_Neumann.txt` and `SampleGrid_Neumann2.txt`) with different initial mesh and different Neumann boundary and refine the domain uniformly 7 times. Solve with the already implemented standard (Jacobi) preconditioned CG, the multilevel preconditioned CG and the hierarchical basis preconditioned CG and write out the number of iterations of the algorithm on each level.

[**Bonus**: 5 points] Solve the problem from sheet 3 with 7 refinements of the Rivara algorithm and use the different preconditioners to compare the iteration numbers.