Algebraic Multigrid with Local Support*

Gundolf Haase†

28th December 1999

1 Introduction

Without loss of generality we want to solve a linear second order partial differential equation (PDE) with homogeneous Dirichlet boundary conditions in a domain $\Omega \subset \mathbb{R}^d$, $d = 2, 3$

$$\begin{aligned} L(x)u(x) &= F(x) \qquad \forall x \in \Omega \\ u(x) &= 0 \qquad \forall x \in \partial\Omega \end{aligned} \tag{1}$$

(with appropriate function spaces and interface conditions), such that the weak formulation is represented by:

$$\text{Find } u \in \mathbf{H}_1(\Omega) : \quad a(u,v) = \langle F, v \rangle \quad \forall v \in \mathbf{H}_1(\Omega) \tag{2}$$

with bilinear form $a(u,v) : \mathbf{H}_1 \times \mathbf{H}_1 \mapsto \mathbb{R}$ and $\langle F, v \rangle : \mathbf{H}_1^* \times \mathbf{H}_1 \mapsto \mathbb{R}$ as duality product.

A discretization of the domain Ω results in a finite element (f.e.) mesh τ_h consisting of elements δ^r and nodes $x^{[j]}$ with $j \in \omega$. We denote by $r \in \tau_h$ that an element δ^r is part of mesh τ_h and assume for simplicity that $\Omega = \bigcup_{r \in \tau_h} \delta^{(r)}$ holds which implies $\delta^{(r)} \subset \Omega, \quad \forall r \in \tau_h$.
The f.e. basis $\Phi = \left\{ \varphi^{[j]}(x) \,,\, j \in \omega \right\}$ has local support

$$\overline{\text{supp}}\, \varphi^{[i]} = \bigcup_{r:\, x^{[i]} \in \delta^{(r)}} \delta^{(r)} \tag{3}$$

and spans a f.e. function space $\mathbb{V} = \text{span}\Phi$ and a test space $\mathbb{W} = \mathbb{V}$. Eq. (2) changes to:

$$\text{Find } u_h \in \mathbb{V}(\Omega) \subseteq \mathbf{H}_1 : \quad a(u_h, v_h) = \langle F, v_h \rangle \quad \forall v_h \in \mathbb{W}(\Omega) \ . \tag{4}$$

*This work has been partially supported by the Austrian Science Fund - "Fonds zur Förderung der wissenschaftlichen Forschung" - within the SFB F013 "Numerical and Symbolic Scientific Computing"

†Institute for Analysis and Computational Mathematics, Johannes Kepler University Linz, A-4040 Linz, Austria, ghaase@numa.uni-linz.ac.at

Now, the f.e. isomorphism

$$u_h(x) \;=\; \sum_{j\in\omega} u^{[j]} \cdot \varphi^{[j]}(x) \;=\; \Phi \cdot \underline{u} \;, \tag{5}$$

guarantees $\forall u_h \in \mathbb{V}$, $v_h \in \mathbb{W}$ the equivalence of (4) to the system of equations

$$K_h \cdot \underline{u} \;=\; \underline{f} \tag{6}$$

with

$$K_h^{[i,j]} \;=\; a(\varphi^{[j]}, \varphi^{[i]}) \tag{7}$$

$$f^{[i]} \;=\; \left\langle F, \varphi^{[i]} \right\rangle \qquad\qquad \forall i,j \in \omega \;. \tag{8}$$

$K_h^{[i,j]}$ will be generated by means of element matrices

$$K_{ij}^{(r)} = a(\varphi^{[j]}\big|_{\delta^{(r)}}, \varphi^{[i]}\big|_{\delta^{(r)}}) \qquad \forall \delta^{(r)} \in \tau_h \tag{9}$$

such that

$$K_h^{[i,j]} = \sum_{r\in\tau_h} K_{ij}^{(r)} \qquad\qquad \forall i,j \in \omega \;. \tag{10}$$

Solving (6) can be done by means of algebraic multigrid (AMG) methods for various operators. Usually, they require the presetting of one or two parameters in the coarsening process which often change also with the discretization. Most AMG algorithms start with the accumulated matrix from (7) and neglect the additional information available with the element matrices whose are automatically available in FEM-codes. On the basis of element matrices, Reitzinger [4] developed a coarsening algorithms which uses only the information already contained in the element matrices surrounding a node. We will briefly introduce this technique in Sec. 2. The problem is, that the standard AMG interpolation does not preserve the sparsity structure of the original matrix and therefore the new coarsening algorithm will be too expensive after a few coarsening steps. Section 3 constructs a new coarse f.e. basis resulting in a new element wise interpolation such that the coarse matrix can be again expressed as patches, see Sec. 3.5. The algorithms for the new two-grid AMG are presented in Sec. 4 followed by the conclusions in the final section.

2 Some preliminaries

An AMG algorithm consists of the following components:

1. Determining the set of coarse nodes ω_C ("coarse grid") such that $\omega = \omega_C \cup \omega_F$, $N = |\omega|$, $N_C = |\omega_C|$ and $N_F = |\omega_F|$.

2. Defining an interpolation matrix $P = \{p_{ij}\}_{i\in\omega, j\in\omega_C} \ : \ \mathbb{R}^{N_C} \mapsto \mathbb{R}^N$ for intergrid transfer from coarse to fine which preserves the local null space of K, i.e., the row sum of K.

3. Calculation of coarse matrix $K_H \in \mathbb{R}^{N_C} \times \mathbb{R}^{N_C}$ by the Galerkin approach

$$K_H \ := \ P^T \cdot K_h \cdot P \ . \tag{11}$$

4. Recursive application of setup steps 1.-3.

5. Ordinary multigrid with components from above.

Coarsening and interpolation algorithms in AMG use information from matrix entries of the neighborhood $\mathcal{N}_s$ around a node $s \in \omega$. The more information of the PDE is contained therein the better these components are (and the resulting multigrid algorithm is more robust). We are interested in a good local approximation of the PDE and, therefore, we want to use an approximation of the PDE in patch $\Omega_s \subset \Omega$ to derive the interpolation operator.

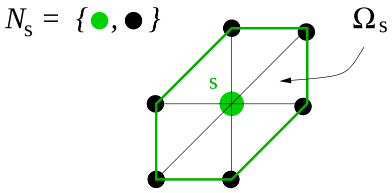

Figure 1: Neighborhood $\mathcal{N}_s$ and patch Ω_s of a node s.

Let us denote by $\widetilde{\Omega}_s$ the patch Ω_s extended by the next layer of elements. The choice $\widetilde{K}_s = \sum\limits_{i\in\mathcal{N}_s} \sum\limits_{j\in\mathcal{N}_s} K_h^{[i,j]}$ for patch Ω_s is the discrete representation of

$$\begin{aligned} L(x)u(x) &= F(x) \qquad \forall x \in \widetilde{\Omega}_s \\ u(x) &= 0 \qquad\quad \forall x \in \partial\widetilde{\Omega}_s \ , \end{aligned}$$

i.e., some artificial Dirichlet boundary conditions appear in the local PDE. On the other hand, the choice

$$\widehat{K}_s \ := \sum_{\delta^{(r)}\subset\Omega_s} K^{(r)} \tag{12}$$

together with $\widehat{\underline{f}}_s := \sum\limits_{\delta^{(r)}\subset\Omega_s} \underline{f}^{(r)}$ represents die local PDE

$$\begin{aligned} L(x)u(x) &= F(x) \qquad \forall x \in \Omega_s \\ u(x) &= 0 \qquad \forall x \in \partial\Omega_s \cap \partial\Omega \\ \frac{\partial u(x)}{\partial \mathbf{n}} &= 0 \qquad \forall x \in \partial\Omega_s \setminus \partial\Omega \ , \end{aligned}$$

which has no additional Dirichlet boundary conditions. The usage of (12) preserves as much as possible from the original problem (1) in the local one and therefore we require for our algorithms the knowledge of the element matrices $K^{(r)}$. These patch matrices $\widehat{K}_s$ will be used in setup steps 1 of AMG to determine the direction of the algebraically smooth error, i.e., to find out the coarse and fine nodes around a fine node s.

3 The new coarse basis

3.1 Notations

We start with the element matrices $K^{(r)}$ defined in (9) and K_h from (10). The following notations are introduced:

- $\mathcal{E}_q := \{r \in \tau_h \,:\, K_{qq}^{(r)} \neq 0\}$ - element neighborhood of node q,
- $\mathcal{E}_r^* := \{q \in \omega \,:\, r \in \mathcal{E}_q\} = \{q \in \omega \,:\, K_{qq}^{(r)} \neq 0\}$ - nodes in element $\delta^{(r)}$.
- $\mathcal{N}_q := \bigcup_{r \in \mathcal{E}_q} \mathcal{E}_r^*$ - node neighborhood of node q.

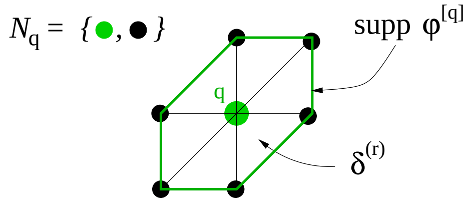

Figure 2: Neighborhood $\mathcal{N}_q$ and support of base function $\varphi^{[q]}$ of a node q.

We can group elements near a node to a fine patch and define

- $\mathcal{P}_q^h$ - set of elements for fine patch q, $q \in \omega$,
- $\mathcal{P}_r^{h^*} := \{q \in \omega \,:\, r \in \mathcal{P}_q^h\}$ - fine patches containing element r,
- $\Omega_q^h := \bigcup_{r \in \mathcal{P}_q^h} \delta^{(r)}$ - fine patch q.
- $\mathcal{N}_q^h := \bigcup_{r \in \mathcal{P}_q^h} \mathcal{P}_r^{h^*}$ - nodes in fine patch q,

Accumulating the stiffness matrix over this fine patch results in the patch matrix

$$K_{h,q}^{[i,j]} = \sum_{r \in \mathcal{P}_q^h} K_{ij}^{(r)} \quad . \tag{13}$$

The simplest (but not necessary) choice would be $\mathcal{P}_q^h := \mathcal{E}_q$ such that Ω_q^h is identical to the support of the base function of node q, implying $\mathcal{P}_r^{h^*} = \mathcal{E}_r^*$.

The local patch matrices $K_{h,q}$ can be used to find strong connections in the neighborhood of node q for the coarsening $\omega = \omega_C \cup \omega_F$. The prolongation operator P defines for each node $k \in \omega$ its interpolation pattern $\mathcal{M}_k := \{j \in \omega_C \,:\, p^{[k,j]} \neq 0\}$ (rows in interpolation operator) and the restriction pattern $\mathcal{M}^*_j := \{k \in \omega \,:\, p^{[k,j]} \neq 0\}$ for all $j \in \omega_C$. The relation $\mathcal{M}_k = \mathcal{N}_k \cap \omega_C$ is valid for standard cases, e.g., Laplacian, but it may become necessary for for anisotropic problems that $p^{[k,i]} \neq 0$ although $K^{[k,i]} = 0$!

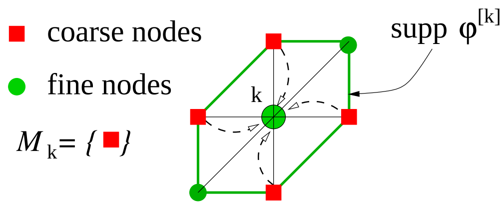

Figure 3: Interpolation pattern of node k.

Straightforward, we can combine a bigger set of elements to a coarse patch, i.e.,

- $\mathcal{P}^H_s$ - set of elements for coarse patch s, $s \in \omega_C$,
- $\mathcal{P}^{H^*}_r := \{s \in \omega_C \,:\, r \in \mathcal{P}^H_s\}$ - coarse patches containing element r,
- $\Omega^H_s := \bigcup_{r \in \mathcal{P}^H_s} \delta^{(r)}$ - coarse patch s.
- $\mathcal{N}^H_s := \bigcup_{r \in \mathcal{P}^H_s} \mathcal{P}^{H^*}_r$ - nodes in coarse patch s,

3.2 Standard approach

We mean by standard approach a classical interpolation P introduced in [5] together with the Galerkin approach (11) for coarse matrix ($\forall i, j \in \omega_C$)

$$K^{[i,j]}_H = \left(P^T \cdot K_h \cdot P\right)^{[i,j]} = \sum_{k \in \mathcal{M}^*_i} \sum_{\ell \in \mathcal{M}^*_j} p^{[k,i]} K^{[k,\ell]}_h p^{[\ell,j]} \; . \tag{14}$$

This is exactly eq. (7) using coarse basis functions $\varphi^{[i]}_H$ from the standard coarse basis

$$\Phi^H = \left\{ \varphi^{[i]}_H(x) = \sum_{k \in \mathcal{M}^*_i} p^{[k,i]} \cdot \varphi^{[k]}(x) \, , \, i \in \omega_C \right\} = \Phi \cdot P \tag{15}$$

instead of those from the the original f.e. basis $\Phi = \left\{\varphi^{[k]}(x) \, , \, k \in \omega\right\}$. The above definition for the coarse basis suggests the natural choice for fine and coarse patches

$$\mathcal{P}_q^h := \mathcal{E}_q \qquad \Longrightarrow \qquad \Omega_q^h = \bigcup_{r\in\mathcal{E}_q} \delta^{(r)} \qquad = \operatorname{supp}\varphi_h^{[q]} \tag{16}$$

$$\mathcal{P}_s^H := \bigcup_{q\in\mathcal{M}_s^*} \mathcal{P}_q^h \qquad \Longrightarrow \qquad \Omega_s^H = \bigcup_{r\in\mathcal{P}_s^H} \delta^{(r)} = \bigcup_{q\in\mathcal{M}_s^*} \Omega_q^h = \operatorname{supp}\varphi_H^{[s]} \tag{17}$$

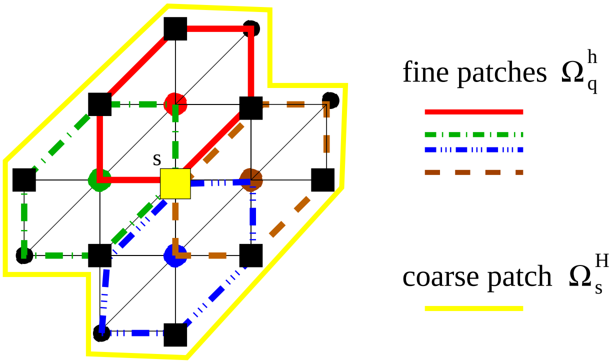

Figure 4: Standard coarse patch Ω_s^H, $s \in \omega_C$.

The coarse matrices from (14) are getting denser with each coarsening level by long connections, i.e., by matrix entries $K_H^{[i,j]}$, $j \notin \mathcal{N}_i^H$ due to an overlap $\Omega_i^H \cap \Omega_j^H \neq \emptyset$ caused by the overlap $\Omega_k^h \cap \Omega_\ell^h \neq \emptyset$ with $k \in \mathcal{M}_i^* \setminus \mathcal{M}_j^*$, $\ell \in \mathcal{M}_j^* \setminus \mathcal{M}_i^*$, see Fig. 5.

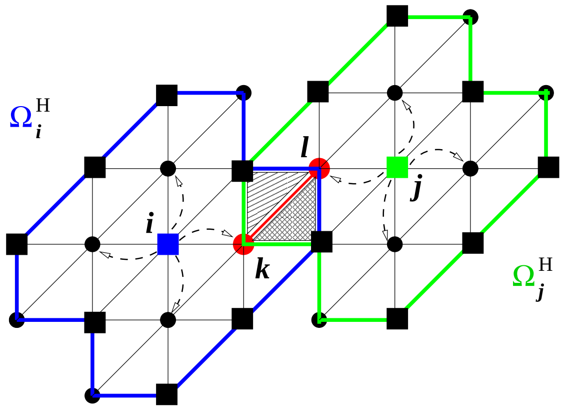

Figure 5: Long connection between coarse nodes i and j.

3.3 Skipping/lumping of matrix entries

The simplest solution, skipping all long connections in K_H would change the row sum of K_H, and therefore change the properties of the coarse operator in general. This can be easily verified for the discretization of $L(x) = -\Delta$ in (1) without b.c. in which the constant functions $c(x) \equiv c$ has to be represented on each level such that $K_H \cdot \underline{c}_H = 0$. holds.

A lumping of skipped entries to the main diagonal is one opportunity to preserve the row sum. The disadvantage is, that we cannot interpret the resulting coarse matrix as Galerkin matrix in an appropriate basis spanning a subspace of $\mathbf{H}_1$. The agglomeration method with smoothing by Vanek/Mandel/Brezina [6] solves this problem in a different way.

Does there exist any chance to represent the original operator with its original null space by means of Galerkin approach without having long connections in K_H ?

3.4 Element based interpolation

We observe from Fig. 5 that the overlapping of Ω_i^H and Ω_j^h is caused by the two elements from $\mathcal{P}_i^H \cap \mathcal{P}_j^H$ containing connection $k \leftrightarrow \ell$. Therefore, we simply exclude those and all other critical elements from the new coarse patches $\widehat{\mathcal{P}}_i^H$, $\widehat{\mathcal{P}}_j^H$. The definition for these new coarse patches is

$$\widehat{\mathcal{P}}_i^H := \bigcup_{r:\, \mathcal{E}_r^* \cap \omega_F \subset \mathcal{M}_i^*} \delta^{(r)} \quad , \tag{18}$$

i.e., only those elements are included whose fine nodes contain coarse node i in there standard interpolation pattern. The result can be seen in Fig. 6.

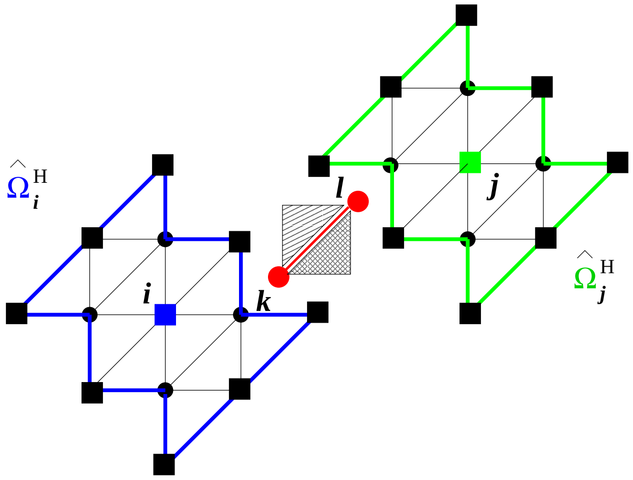

Figure 6: Restricted coarse patches $\widehat{\Omega}_i^H$ and $\widehat{\Omega}_j^H$.

The standard coarse basis Φ^H (15) preserves the the constant function, i.e., the null space of the operator, and we require the same for our new coarse basis $\Psi^H = \{\psi_H^{[i]}(x)\}_{i\in\omega_C}$ with supp $\psi_H^{[i]}(x) := \widehat{\Omega}_i^H$. A standard coarse basis can be expressed as

$$\begin{aligned}\varphi_H^{[i]}(x) &= \sum_{k\in\mathcal{M}_i^*} p^{[k,i]}\cdot\varphi^{[k]}(x) = \sum_{r\in\tau_h}\sum_{k\in\mathcal{E}_r^*} p^{[k,i]}\cdot\varphi^{[k]}(x)\big|_{\delta^{(r)}} \\ &= \sum_{r\in\widehat{\mathcal{P}}_i^H}\sum_{k\in\mathcal{E}_r^*} p^{[k,i]}\cdot\varphi^{[k]}(x)\big|_{\delta^{(r)}} + \sum_{r\notin\widehat{\mathcal{P}}_i^H}\sum_{k\in\mathcal{E}_r^*} p^{[k,i]}\cdot\varphi^{[k]}(x)\big|_{\delta^{(r)}} \qquad (19)\end{aligned}$$

The second sum contains all shape function outside $\widehat{\mathcal{P}}_i^H$ which we redistribute to preserve the constant. Fig. 7 illustrates this redistribution of shape functions $p^{[k,i]}\cdot\varphi^{[k]}(x)\big|_{\delta^{(r)}}$ to coarse nodes s, t whose patches include the two elements.

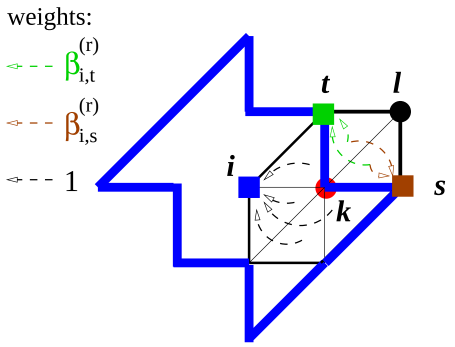

Figure 7: Redistribution of function $p^{[k,i]}\cdot\varphi_k^h(x)$ wrt. $\widehat{\Omega}_i^H$.

The part of $p^{[k,i]}\cdot\varphi^{[k]}(x)$, $k\in\omega_F$ over element $\delta^{(r)}\not\subseteq\widehat{\Omega}_i^H$ will be weighted with $\beta_{k,s}^{(r)}$ and added to the coarse base function of node s, $s : r\in\widehat{\mathcal{P}}_i^H$ such that

$$\sum_{s:\, r\in\widehat{\mathcal{P}}_s^H} \beta_{k,s}^{(r)} = 1 \qquad r\in\mathcal{E}_k\setminus\widehat{\mathcal{P}}_i^H,\ k\in\mathcal{M}_i^* \qquad (20)$$

holds. If we use some matrix weighted prolongation then the choice

$$\beta_{k,s}^{(r)} = p^{[k,s]}\Big/\sum_{j:\, r\in\widehat{\mathcal{P}}_j^H} p^{[k,j]} \qquad (21)$$

takes discontinuous coefficients into account.

Lemma 3.1
The new coarse basis $\Psi^H = \{\psi_H^{[i]}(x)\}_{i\in\omega_C}$ consisting of base functions

$$\psi_H^{[i]}(x) := \sum_{r\in\widehat{\mathcal{P}}_i^H}\sum_{k\in\mathcal{E}_r^*}\left\{p^{[k,i]} + \sum_{s:\, r\notin\widehat{\mathcal{P}}_s^H} p^{[k,s]}\beta_{k,i}^{(r)}\right\}\cdot\varphi^{[k]}(x)\big|_{\delta^{(r)}} \qquad (22)$$

with $\beta_{k,i}^{(r)}$ from (21) has local support $\operatorname{supp}\psi_H^{[i]}(x) = \widehat{\Omega}_i^H$ *and preserves the constant.*

Proof. The local support property follows directly from definition (22) and the standard coarse basis Φ^H (15) preserves already the constant. Neglecting bondary conditions, we can express this constant as

$$1 \equiv \sum_{i\in\omega_C} \varphi_H^{[i]}(x) \;=\; \sum_{i\in\omega_C} \sum_{k\in\mathcal{M}_i^*} p^{[k,i]} \cdot \varphi^{[k]}(x) \;\; .$$

Application of (19) and (20) changes into

$$1 = \sum_{i\in\omega_C} \sum_{r\in\widehat{\mathcal{P}}_i^H} \sum_{k\in\mathcal{E}_r^*} p^{[k,i]} \cdot \varphi^{[k]}(x)\big|_{\delta^{(r)}} + \sum_{i\in\omega_C} \sum_{r\notin\widehat{\mathcal{P}}_i^H} \sum_{k\in\mathcal{E}_r^*} \underbrace{\sum_{s:\, r\in\widehat{\mathcal{P}}_s^H} \beta_{k,s}^{(r)}}_{=1} p^{[k,i]} \cdot \varphi^{[k]}(x)\big|_{\delta^{(r)}}$$

and $\sum\limits_{i\in\omega_C} \sum\limits_{r\notin\widehat{\mathcal{P}}_i^H} \sum\limits_{k\in\mathcal{E}_r^*} \sum\limits_{s:\, r\in\widehat{\mathcal{P}}_s^H} = \sum\limits_{r\in\tau_h} \sum\limits_{i:\, r\notin\widehat{\mathcal{P}}_i^H} \sum\limits_{s:\, r\in\widehat{\mathcal{P}}_s^H} \sum\limits_{k\in\mathcal{E}_r^*} = \sum\limits_{r\in\tau_h} \sum\limits_{s:\, r\in\widehat{\mathcal{P}}_s^H} \sum\limits_{k\in\mathcal{E}_r^*} \sum\limits_{i:\, r\notin\widehat{\mathcal{P}}_i^H}$ rearranges the sums into

$$1 = \sum_{i\in\omega_C} \sum_{r\in\widehat{\mathcal{P}}_i^H} \sum_{k\in\mathcal{E}_r^*} p^{[k,i]} \cdot \varphi^{[k]}(x)\big|_{\delta^{(r)}} + \sum_{s\in\omega_C} \sum_{r\in\widehat{\mathcal{P}}_s^H} \sum_{k\in\mathcal{E}_r^*} \sum_{i:\, r\notin\widehat{\mathcal{P}}_i^H} \beta_{k,s}^{(r)} p^{[k,i]} \cdot \varphi^{[k]}(x)\big|_{\delta^{(r)}} \;\; .$$

Swapping subscripts i and s in the second term results in

$$1 = \sum_{i\in\omega_C} \sum_{r\in\widehat{\mathcal{P}}_i^H} \sum_{k\in\mathcal{E}_r^*} \Big\{ p^{[k,i]} + \sum_{s:\, r\notin\widehat{\mathcal{P}}_s^H} p^{[k,s]} \beta_{k,i}^{(r)} \Big\} \cdot \varphi^{[k]}(x)\big|_{\delta^{(r)}} \;=\; \sum_{i\in\omega_C} \psi_H^{[i]}(x)$$

■

Lemma 3.1 can be applied directly as construction principle for a new interpolation $\widehat{P}$.

Lemma 3.2
The element weighted prolongation $\widehat{P} := \Big\{ p_{k,i}^{(r)} \Big\}_{k\in\omega, i\in\omega_C, r\in\tau_h}$ *is defined by*

$$p_{k,i}^{(r)} \;:=\; \alpha_k^{(r)} \cdot p^{[k,i]} \;, \qquad \textit{with} \qquad \alpha_k^{(r)} \;:=\; \frac{\sum_{s\in\mathcal{M}_k} p^{[k,s]}}{\sum_{s:\, r\in\widehat{\mathcal{P}}_s^H} p^{[k,s]}} \tag{23}$$

and $p^{[k,i]}$ from standard interpolation, and transforms the original fine basis Φ^h into $\Psi^H = \Phi^h \cdot \widehat{P}$.

Proof. We set $p_{k,i}^{(r)}$ equal the term in braces from (22) and replace $\beta_{s,i}^{(r)}$ with its definition (21):

$$
\begin{aligned}
p_{k,i}^{(r)} \;&:=\; p^{[k,i]} + \sum_{s\colon r\notin\widehat{\mathcal{P}}_s^H} p^{[k,s]}\cdot\beta_{k,i}^{(r)} \;=\; p^{[k,i]} + \sum_{s\colon r\notin\widehat{\mathcal{P}}_s^H} p^{[k,s]}\cdot\frac{p^{[k,i]}}{\sum\limits_{j\colon r\in\widehat{\mathcal{P}}_j^H} p^{[k,j]}} \\
&=\; p^{[k,i]}\cdot\left(1+\frac{\sum_{s\colon r\notin\widehat{\mathcal{P}}_s^H} p^{[k,s]}}{\sum_{j\colon r\in\widehat{\mathcal{P}}_j^H} p^{[k,j]}}\right) \;=\; p^{[k,i]}\cdot\underbrace{\frac{\sum_{s\in\mathcal{M}_k} p^{[k,s]}}{\sum_{s\colon r\in\widehat{\mathcal{P}}_s^H} p^{[k,s]}}}_{=:\alpha_k^{(r)}}\;.
\end{aligned}
$$

Now, $\Psi^H = \Phi^h\cdot\widehat{P}$ follows directly from Lemma 3.1. ∎

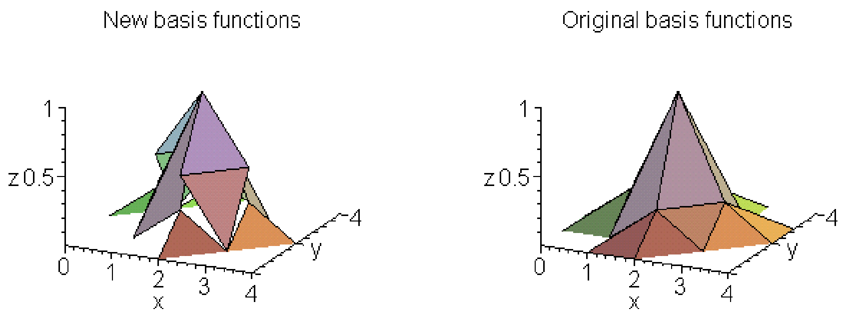

Figure 8: New and original basis functions.

Remark 3.1
If the local null space of K is only the constant, e.g., for the Laplace equation, then the numerator in (23) is equal 1.

Remark 3.2
Relations (23) simplify the presentation of coarse basis functions (22) to

$$
\psi_H^{[i]}(x) \;:=\; \sum_{r\in\widehat{\mathcal{P}}_i^H}\sum_{k\in\mathcal{E}_r^*} p^{[k,i]}\cdot\alpha_k^{(r)}\cdot\varphi^{[k]}(x)\big|_{\delta^{(r)}}\;, \tag{24}
$$

i.e., the shape functions are scaled seperately in each element.

Lemma 3.3
The new coarse matrix K_H representing (1) in the new basis Ψ^H from (22) has the entries

$$
K_H^{[i,j]} = \sum_{r\in\widehat{\mathcal{P}}_i^H\cap\widehat{\mathcal{P}}_j^H}\sum_{k,\ell\in\mathcal{E}_r^*} p^{[\ell,i]}\cdot\alpha_\ell^{(r)}\cdot K_{\ell,k}^{(r)}\cdot\alpha_k^{(r)}\cdot p^{[k,j]} \qquad,\; i,j\in\omega_C\;. \tag{25}
$$

Proof. Follows directly by applying basis Ψ^H and (23) in (7)-(10). ■

The element wise interpolation $\widehat{P}$ cannot be applied directly to interpolation and restriction, we have to modify it.

Lemma 3.4
The element weighted node interpolation

$$\widetilde{P} := \left\{ \widetilde{p}_{k,i}^{(r)} := \gamma_k^{(r)} \cdot p_{k,i}^{(r)} \right\}_{k\in\omega, i\in\omega_C, r\in\tau_h} , \quad \textit{with} \quad \gamma_k^{(r)} := \frac{K_{k,k}^{(r)}}{K^{[k,k]}} \tag{26}$$

and $p_{k,i}^{(r)}$ *from (23), interpolates* $\underline{u}_H \in \mathbb{R}^{\omega_C}$ *into* $\underline{u}_h = \sum_{r\in\tau_h} \widetilde{P} \cdot \underline{u}_H \in \mathbb{R}^{\omega}$.

Proof. We cannot express $\underline{u}_h$ in terms of $\underline{u}_H$ because of the element weights. Therefore, we introduce weights $\gamma_k^{(r)}$ such that $\sum\limits_{r\in\mathcal{E}_k} \gamma_k^{(r)} \equiv 1$ holds and use

$$\sum_{r\in\mathcal{E}_k} \varphi^{[k]}\Big|_{\delta^{(r)}} = \varphi^{[k]}(x) = \sum_{r\in\mathcal{E}_k} \gamma_k^{(r)} \cdot \varphi^{[k]}(x) \tag{27}$$

as averaging. The f.e. isomorphism (5) and weights (26) allow us to write the interpolated function of $u_H(x)$ as

$$\begin{aligned} u_h(x) = \Phi^h \cdot \underline{u}_h &= \Phi^h \cdot \sum_{r\in\tau_h} \widetilde{P} \cdot \underline{u}_H = \sum_{k\in\omega} \sum_{i\in\mathcal{M}_k} \sum_{r\in\mathcal{E}_k} \gamma_k^{(r)} \cdot p_{k,i}^{(r)} \cdot u_H^{[i]} \cdot \varphi^{[k]}(x) \\ &= \sum_{k\in\omega} \underbrace{\sum_{r\in\mathcal{E}_k} \left(\sum_{i\in\mathcal{M}_k} p_{k,i}^{(r)} \cdot u_H^{[i]} \right) \cdot \gamma_k^{(r)}}_{=:u_h^{[k]}} \cdot \varphi^{[k]}(x) = \sum_{k\in\omega} u_h^{[k]} \cdot \varphi^{[k]}(x) \ . \end{aligned}$$

The special assumption $p_{k,i}^{(r)} = p_{k,i}^{(q)}$, $\forall r, q \in \widehat{\mathcal{P}}_i^H$ together with (27) leads to

$$= \sum_{k\in\omega} \sum_{r\in\mathcal{E}_k} \left(\sum_{i\in\mathcal{M}_k} p_{k,i}^{(r)} \cdot u_H^{[i]} \right) \cdot \varphi^{[k]}\Big|_{\delta^{(r)}} = \sum_{i\in\omega_C} \sum_{k\in\mathcal{M}_i^*} \sum_{r\in\mathcal{E}_k} p_{k,i}^{(r)} \cdot u_H^{[i]} \cdot \varphi^{[k]}\Big|_{\delta^{(r)}}$$

and, because of $p_{k,i}^{(r)} = 0 \quad \forall r \notin \widehat{\mathcal{P}}_i^H$, finally to

$$\begin{aligned} &= \sum_{i\in\omega_C} u_H^{[i]} \sum_{r\in\widehat{\mathcal{P}}_i^H} \sum_{k\in\mathcal{E}_r^*} p_{k,i}^{(r)} \cdot \varphi^{[k]}\Big|_{\delta^{(r)}} = \sum_{i\in\omega_C} u_H^{[i]} \psi_H^{[i]}(x) \\ &= \Psi^H \cdot \underline{u}_H = u_H(x) = \Phi^h \widehat{P} \cdot \underline{u}_H \ . \end{aligned}$$

■

Remark 3.3
Only the special assumption $p_{k,i}^{(r)} = p_{k,i}^{(q)}$, $\forall r, q \in \widehat{\mathcal{P}}_i^H$ *leads to the conclusion* $u_h(x) \equiv u_H(x)$ *in the above proof. This is not valid in general! This implies also that* $\widehat{P}^T K \widehat{P} \neq \widetilde{P}^T K \widetilde{P}$. *The interpolation* $\widetilde{P}$ *defines a new coarse basis in the way of Lemma 3.2. The choice of* $\gamma_k^{(r)}$ *in (26) guarantees* $p_{k,i}^{(q)} = \sum_{r\in\widehat{\mathcal{P}}_i^H} p_{k,i}^{(r)}$, $\forall q \in \widehat{\mathcal{P}}_i^H$.

Remark 3.4
We could also define our coarse grid matrix as $\widetilde{K}_H = \widetilde{P}^T K \widetilde{P}$ *with entries*

$$\widetilde{K}_H^{[i,j]} = \sum_{r\in\widehat{\mathcal{P}}_i^H\cap\widehat{\mathcal{P}}_j^H} \sum_{k,\ell\in\mathcal{E}_r^*} p^{[\ell,i]} \cdot \alpha_\ell^{(r)} \cdot \gamma_\ell^{(r)} \cdot K^{[\ell,k]} \cdot \gamma_k^{(r)} \cdot \alpha_k^{(r)} \cdot p^{[k,j]} \quad , \tag{28}$$

$i, j \in \omega_C$. *The term* $\gamma_\ell^{(r)} \cdot K^{[\ell,k]} \cdot \gamma_k^{(r)}$ *in (28) can be interpreted as an averaged distribution of the accumulated matrix entry* $K^{[\ell,k]}$ *on the elements, which contains less physical information than the element entry* $K_{\ell,k}^{(r)}$. *Therfore, we prefer to use (25) for the definition of the coarse matrix.*

3.5 The coarse patches

We cannot apply the approach from the previous section for calculating a coarse matrix with local support recursively. The reason consists in the fact that $\widehat{\Omega}_s^H \subset \Omega_s^H = \bigcup_{q\in\mathcal{M}_s^*} \Omega_q^h$, we cannot use the matrices of the fine patches as elements for the coarse matrix. The author has not find a way to define such "coarse" elements. Therefore, we try to get at least the patch matrices $K_{H,s}$, $s \in \omega_C$ to continue the coarsening process based on patches.

$$K_{H,s}^{[i,j]} = \sum_{r\in\widehat{\mathcal{P}}_s^H} \sum_{k,\ell\in\mathcal{E}_r^*} p^{[\ell,i]} \cdot \alpha_\ell^{(r)} \cdot K_{\ell,k}^{(r)} \cdot \alpha_k^{(r)} \cdot p^{[k,j]} \quad , \; i, j \in \mathcal{N}_s^H \cap \omega_C \; . \tag{29}$$

Note, that we have to store K_H and all $K_{H,s}$ because $K_H \neq \sum_{s\in\omega_C} K_{H,s}$.

4 Two grid AMG with local basis

4.1 Some auxiliary routines

We assume that a coarsening algorithm $\textsc{Coarse}(K, \omega, \omega_C, \omega_F)$ and a standard algorithm for calculating the interpolation weights $\textsc{Weights}(K, \omega_C, \omega_F, P)$ are already performed. Both may use element matrices $K_{ij}^{(r)}$ instead of the accumulated one. We need a routine to calculate the weights α in (23).

Algorithm 1 Calculation of α-weights $\textsc{CalcAlpha}(P, \omega_C, \omega_F, \alpha)$

```
t̲ ← 0                                   {Numerator: temporary array.}
for all k ∈ ω do
  for all i ∈ M_k do
    t_k ← t_k + p^[k,j]
  end for
end for
α̲ ← 0                                   {Denominator: α̲ as temporary array.}
for all s ∈ ω_C do
  for all r ∈ P̂_s^H do
    for all k ∈ E_r^* do
      if s ∈ M_k then
        α_k^(r) ← α_k^(r) + p^[k,j]
      end if
    end for
  end for
end for
for all r ∈ τ_h do {                     Calculate fraction.}
  for all k ∈ E_r^* do
    α_k^(r) ← t_k / α_k^(r)
  end for
end for
```

4.2 Coarse matrix

A direct implementation of formulae (25) would require too much search processes for the usual storing of data. Therefore, the loops over the coarse nodes will be the most inner ones.

We realize (29) by adding the following loop after the IF-statement in Alg. 2.

for all $s \in \mathcal{N}_i^H \cap \mathcal{N}_j^H \cap \omega_C$ **do**

$K_{H,s}^{[i,j]} \leftarrow K_{H,s}^{[i,j]} + p^{[\ell,i]} \cdot \alpha_\ell^{(r)} \cdot K_{\ell,k}^{(r)} \cdot \alpha_k^{(r)} \cdot p^{[k,j]}$

end for

Algorithm 2 New Galerkin approach $\textsc{Galerkin}(K, \omega_C, \omega_F, K_H)$

$\textsc{Weights}(K, \omega_C, \omega_F, P)$ {Calculate standard weights.}
$\textsc{CalcAlpha}(P, \omega_C, \omega_F, \alpha)$ {Calculate α-weights.}
$K_H \leftarrow 0$
for all $r \in \tau^h$ **do** { All elements.}
  for all $k \in \mathcal{E}_r^*$, $\ell \in \mathcal{E}_r^*$ **do** { Element nodes.}
    for all $j \in \mathcal{M}_k$, $i \in \mathcal{M}_\ell$ **do** { Coarse nodes.}
      if $r \in \widehat{\mathcal{P}}_i^H \cap \widehat{\mathcal{P}}_j^H$ **then** { Element $\delta^{(r)}$ in both coarse patches?}
        $K_H^{[i,j]} \leftarrow K_H^{[i,j]} + p^{[\ell,i]} \cdot \alpha_\ell^{(r)} \cdot K_{\ell,k}^{(r)} \cdot \alpha_k^{(r)} \cdot p^{[k,j]}$
      end if
    end for
  end for
end for

4.3 Element weighted interpolation

We provided already α in (23) by Alg. 1 such that the interpolation can be implemented as follows.

Algorithm 3 Element weighted interpolation $\textsc{EWeights}(K, \omega_C, \omega_F)$

$\textsc{Weights}(K, \omega_C, \omega_F, P)$ {Calculate standard weights.}
$\textsc{CalcAlpha}(P, \omega_C, \omega_F, \alpha)$
$\underline{u}_h \leftarrow 0$
for all $r \in \tau^h$ **do**
  for all $k \in \mathcal{E}_r^*$ **do**
    for all $i \in \mathcal{M}_k$ **do**
      $u_{h,k} \leftarrow u_{h,k} + \gamma_k^{(r)} \cdot \alpha_k^{(r)} \cdot p^{[k,i]} \cdot u_{H,i}$
    end for
  end for
end for

5 Conclusions

We derived an element wise interpolation implying a non-continuous coarse basis with local support. A two level scheme results in a sparse matrix which represents exactly bilinear form (9) in this new basis Ψ^H so that we have again coarse patches with a limited number of nodes in it.

The algorithms are rather easy to implement due to the fact that they require only a small amount of additional data structures. Until now, only some experiments using MAPLE have been done. The weights for Galerkin projection (23) and for prolongation (26) are different but the author cannot find an opportunity to apply (23) in the node prolongation. On the other hand the use of (28) as coarse matrix would be feasible.

Unfortunately, we have no "coarse" elements so that we have to skip long connections on the next coarse level to preserve sparsity. Potentially, we could use some modified (element weighted) patch matrices $K_{H,s}$ as element input for this level. A recursive algorithm of the presented strategy makes sense and can be realized if we have an extended interpolation stencil such that $\mathcal{M}_i^* = \mathcal{N}_i^H \cap \omega_F$ holds, i.e., the restriction to a coarse node i gets input from all fine nodes of the coarse patch Ω_i^H (see also Fig. 3 and Fig. 4).

References

[1] M. Brezina, A. Cleary, R. Falgout, V. H. J. Jones, S. M. T.A. Manteuffel, and J. Ruge, *Algebraic multigrid based on element interpolation (AMGe)*, SIAM J.Sci.Comp., (1998). submitted for publication.

[2] S. Reitzinger, *Algebraic multigrid and element preconditioning I*, Preprint SFB Report 98-15[1], 1998.

[3] ——, *Algebraic multigrid and element preconditioning II*, Preprint SFB Report 99-18[2], 1999.

[4] ——, *Algebraic multigrid coarsening on patch basis*, Preprint 00-00, SFB F013 Linz, 1999. to appear.

[5] J. W. Ruge and K. Stüben, *Algebraic multigrid (AMG)*, in Multigrid Methods, S. F. McCormick, ed., vol. 3 of Frontiers in Applied Mathematics, SIAM, Philadelphia, PA, 1987, pp. 73–130.

[6] P. Vaněk, J. Mandel, and M. Brezina, *Algebraic multigrid based on smoothed aggregation for second and fourth order problems*, Computing, 56 (1996), pp. 179–196.

[1] http://www.sfb013.uni-linz.ac.at/ reitz/sfb01.ps

[2] http://www.sfb013.uni-linz.ac.at/ reitz/sfb02.ps