



LAWRENCE  
LIVERMORE  
NATIONAL  
LABORATORY

# AMG by element agglomeration and constrained energy minimization interpolation

Tz. V. Kolev, P. S. Vassilevski

March 1, 2006

Numerical Linear Algebra with Applications

## **Disclaimer**

---

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

# AMG BY ELEMENT AGGLOMERATION AND CONSTRAINED ENERGY MINIMIZATION INTERPOLATION

TZANIO V. KOLEV AND PANAYOT S. VASSILEVSKI

ABSTRACT. This paper studies AMG (algebraic multigrid) methods that utilize energy minimization construction of the interpolation matrices locally, in the setting of element agglomeration AMG. The coarsening in element agglomeration AMG is done by agglomerating fine-grid elements, with coarse element matrices defined by a local Galerkin procedure applied to the matrix assembled from the individual fine-grid element matrices. This local Galerkin procedure involves only the coarse basis restricted to the agglomerated element. To construct the coarse basis, one exploits previously proposed constraint energy minimization procedures now applied to the local matrix. The constraints are that a given set of vectors should be interpolated exactly, not only globally, but also locally on every agglomerated element. The paper provides algorithmic details, as well as a convergence result based on a “local-to-global” energy bound of the resulting multiple-vector fitting AMG interpolation mappings. A particular implementation of the method is illustrated with a set of numerical experiments.

## 1. AMGE BACKGROUND

Element based AMG methods (or AMGe for short), were proposed in [2], [10], [5], [6], see also [15]. The AMGe methods have the advantage that they can assess the quality of their interpolation mappings by proving global estimates on the basis of easily ensured in practice local estimates. A possible disadvantage of this class of methods is their somewhat costly setup, both in terms of time and storage, since one has to assemble local matrices and create coarse element matrices at every coarsening step. For difficult finite element problems, such as anisotropic problems with anisotropy that is not grid aligned, or thin-body elasticity, AMGe seem to be the only currently available method that has the potential to be scalable (cf., [6]). The version of the AMGe method that we study in the present paper seems to give satisfactory convergence and complexity results for higher order elements (quadratics and cubics) and the high setup cost can be justified if the constructed hierarchy of matrices is used for a sequence of problems.

The paper proposes a combination of the “vector preserving” construction of interpolation matrices introduced in [18] (see also [17]) and its additive Schwarz version in [20], with the element agglomeration used previously. More specifically, we apply the method of [20] locally, on every agglomerated element  $E$  to construct a local interpolation matrix  $P_E$  that fits the restriction of a given vector. If several vectors need to be fitted, one uses the multiple vector interpolation procedure proposed in [16] (again

---

*Date:* April 19, 2005–beginning; Today is February 17, 2006.

*1991 Mathematics Subject Classification.* 65F10, 65N20, 65N30.

*Key words and phrases.* vector preserving interpolation, minimal energy interpolation, element agglomeration, algebraic multigrid.

This work was performed under the auspices of the U. S. Department of Energy by University of California Lawrence Livermore National Laboratory under contract W-7405-Eng-48.

locally). Then, a global interpolation matrix  $P$  is constructed from the local ones, by proper averaging based on partition of unity. The averaging procedure guarantees that the global  $P$  will also fit the given vector(s). We analyze the energy boundedness of  $P$  and prove that local bounds on  $P_E$  imply a global estimate. This yields the two-grid convergence of the resulting method. Details are found in Section 3. Even though it is not discussed in the present paper, another important feature of the method is that it can be used adaptively, in the framework proposed in [4], [3]. This is possible due to the algorithm from [16] which allows one to update a given  $P$  by hierarchically extending the previous coarse space, such that it interpolates a new adaptively computed “algebraically smooth” vector. The remainder of the paper is structured as follows. In Section 2, we describe the energy minimization interpolation procedure and provide an algorithm to generate a global interpolation matrix from local ones. In the following Section 3, we provide an estimate characterizing the global energy boundedness of the interpolation operator assuming natural local energy bounds. We also comment on the case of multiple vector interpolation. The final section contains a number of numerical tests, including ones assessing the convergence properties of our implementation of the method.

## 2. CONSTRAINED ENERGY MINIMIZATION INTERPOLATION

Consider a given (fine-grid) linear system with a symmetric and positive definite (s.p.d.) matrix  $A$ . Let  $\mathcal{N}$  denote the set of row indices of  $A$  (also called fine-grid degrees of freedom, or dofs). Let  $\{E\}$ , the set of agglomerated elements, form an overlapping partition of  $\mathcal{N}$ . For any vector  $\mathbf{v}$  we denote its restriction to  $E$  by  $\mathbf{v}_E$ . We assume that  $A$  is assembled from local matrices  $A_E$  in the sense that for any fine-grid vectors  $\mathbf{v}$  and  $\mathbf{w}$ ,

$$\mathbf{v}^T A \mathbf{w} = \sum_E \mathbf{v}_E^T A_E \mathbf{w}_E.$$

Let  $\mathcal{N}_c \subset \mathcal{N}$  be a given set of coarse degrees of freedom (or coarse-grid dofs). The main task in AMG is the construction of the interpolation operator  $P$ , which maps a coarse-grid vector to a fine-grid one.

In this section, we review the constrained energy minimization interpolation algorithm originating in [18] (see also [17]) in the form studied in [20]. The difference, in our case, is that we use it locally, namely, for every agglomerated element  $E$  and respective local stiffness matrix  $A_E$  we construct a local interpolation matrix  $P_E$ . The global  $P$  will then be obtained from these local matrices by proper averaging at the shared dofs, as explained in Section 3. To construct  $P$  (and  $P_E$ ) we first have to select the coarse dofs  $\mathcal{N}_c$ , and also specify the sparsity pattern, or support  $\Omega_i \subset \mathcal{N}$ , of each column  $\boldsymbol{\psi}_i$ ,  $i \in \mathcal{N}_c$  of the global  $P$ . In AMGe it is natural to define the support  $\Omega_i$  of  $\boldsymbol{\psi}_i$  to be covered by the union  $\mathcal{N}_i$  of all agglomerated elements that share the coarse dof  $i$ . One excludes from  $\Omega_i$  all other coarse dofs as well as fine dofs that belong to agglomerated elements outside of  $\mathcal{N}_i$ . Once the support  $\Omega_i$  is defined, its restriction,  $\Omega_{E,i}$ , to each  $E$  is also determined. For a simple illustration, see Figure 3 in Section 4. The procedure to determine  $P_E$  and hence  $P$ , utilizes an a priori given vector  $\mathbf{e}$ . More specifically, the local interpolation matrix is the solution of the following problem:

Find  $P_E = (\boldsymbol{\psi}_{E,i})_{i \in \mathcal{N}_c \cap E}$ , where each  $\boldsymbol{\psi}_{E,i}$  is supported in  $\Omega_{E,i}$ , with minimal energy, i.e.

$$(2.1) \quad \sum_{i \in \mathcal{N}_c \cap E} \boldsymbol{\psi}_{E,i}^T A_E \boldsymbol{\psi}_{E,i} \mapsto \min,$$

subject to the constraint that for some  $\{g_i\}$

$$(2.2) \quad \sum_{i \in \mathcal{N}_c \cap E} g_i \boldsymbol{\psi}_{E,i} = \mathbf{e}_E.$$

Each  $\boldsymbol{\psi}_i$  is normalized, so that its  $i$ th entry equals one. Therefore, the coefficient  $g_i$  above equals the  $i$ th entry of  $\mathbf{e}$  (due to the proper choice on the support of  $\boldsymbol{\psi}_i$ ).

We next describe the additive Schwarz based algorithm from [20] applied to each  $E$  and  $A_E$ . Let  $I_{E,i}$  be the extension by zero of a vector defined on  $\Omega_{E,i}$  to a vector defined on  $E$ . Then the  $i$ th Schwarz block is defined as  $T_{E,i} = I_{E,i} (I_{E,i}^T A_E I_{E,i})^{-1} I_{E,i}^T$ . To build a  $P_E$  such that  $P_E \mathbf{e}_c = \mathbf{e}_E$ , where  $\mathbf{e}_c = \mathbf{e}_E|_{\mathcal{N}_c \cap E}$  one proceeds as follows. Let  $T_E = \sum_{i \in \mathcal{N}_c \cap E} g_i^2 T_{E,i}$ . Then,  $\boldsymbol{\psi}_{E,i} = g_i T_{E,i} T_E^{-1} \mathbf{e}_E$  defines the  $i$ th column of  $P_E$ .

The vector  $\mathbf{x}_E = T_E^{-1} \mathbf{e}_E$  is computed by the preconditioned CG method applied to  $T_E \mathbf{x}_E = \mathbf{e}_E$  with a simple diagonal preconditioner  $D_E = \sum_{i \in \mathcal{N}_c \cap E} g_i^2 I_{E,i} D_{E,i}^{-1} I_{E,i}^T$ , where

$D_{E,i} = \text{diag} (I_{E,i}^T A_E I_{E,i})$ . It is readily seen (by a local analysis) that  $D_E$  is a spectrally equivalent preconditioner to  $T_E$ . Thus, only few PCG iterations are needed to compute  $\mathbf{x}_E = T_E^{-1} \mathbf{e}_E$ . At any rate, since we are dealing with matrices of small size the latter spectral equivalence is not a major issue here. By construction then,  $P_E \mathbf{e}_c = \sum_{i \in \mathcal{N}_c \cap E} g_i \boldsymbol{\psi}_{E,i} = \mathbf{e}_E$ . The fact that the resulting  $P_E$  leads to a  $P_E^T A_E P_E$  with

minimal trace is proved in [20]. For practical purposes, Gauss-Seidel iterations can be used to evaluate (approximately) the actions of  $(I_{E,i}^T A_E I_{E,i})^{-1}$ . We have to comment that even though  $A_E$  can be only semi-definite, the principal submatrices  $I_{E,i}^T A_E I_{E,i}$  can be assumed invertible. The latter can be achieved if the set  $\mathcal{N}_c \cap E$  is sufficiently rich noting that  $I_{E,i}$  has zero rows corresponding to  $E \cap (\mathcal{N}_c \setminus \{i\})$ .

At this point, we stress upon the fact that the local constrained minimization problem (2.1)–(2.2), gives a mapping  $P_E R_E$  with minimal  $A_E$ -trace norm. Here  $R_E$  is the restriction of fine-grid vectors  $\mathbf{v}_E$  to  $\mathcal{N}_c \cap E$ . Assuming that the size of each  $E$  is bounded, and since all norms in finite dimensional vector space are equivalent, one may expect that the thus computed  $P_E$  will be small in other norms as well. In particular, we assume throughout the rest of the paper the following local estimate, for any vector  $\mathbf{v}_E$

$$(2.3) \quad \mathbf{v}_E^T (I - P_E R_E)^T A_E (I - P_E R_E) \mathbf{v}_E \leq \eta_E \mathbf{v}_E^T A_E \mathbf{v}_E,$$

with a reasonable constant  $\eta_E \geq 1$ . Note that, if  $\mathbf{v}_E$  is in the nullspace of  $A_E$ , the latter estimate implies that  $P_E$  must interpolate that vector exactly, i.e.,  $\mathbf{v}_E = P_E R_E \mathbf{v}_E$ . We address this property further in Section 3.2.

### 3. FROM LOCAL TO GLOBAL ESTIMATES

In this section, we derive energy norm bounds for projections  $Q = PR$ , where  $P$  is an interpolation and  $R$  is a restriction mapping with  $RP = I$ .

**3.1. A general local to global norm bound.** Recall that we are given a set of coarse degrees of freedom (or coarse-grid dofs)  $\mathcal{N}_c$ . We view  $\mathcal{N}_c$ , as a subset of  $\mathcal{N}$ . We have vectors (or grid functions) defined on  $\mathcal{N}$  (fine-grid vectors) and vectors defined on  $\mathcal{N}_c$  (coarse-grid vectors). Let  $n = |\mathcal{N}|$  and  $n_c = |\mathcal{N}_c|$ , with  $n_c < n$ , be the respective size (cardinality) of  $\mathcal{N}$  and  $\mathcal{N}_c$ . The space of coarse-grid vectors is identified with  $\mathbb{R}^{n_c}$  and, similarly, the fine-grid vectors are identified with  $\mathbb{R}^n$ . We have global mappings  $R : \mathbb{R}^n \mapsto \mathbb{R}^{n_c}$  and  $P : \mathbb{R}^{n_c} \mapsto \mathbb{R}^n$ . Respectively, there are local mappings  $R_E$  and  $P_E$ , such that  $R_E$  restricts a local fine-grid vector defined on  $E$  to a local (coarse-grid) vector defined on  $E \cap \mathcal{N}_c$ , whereas the local interpolation mapping  $P_E$  interpolates a local coarse vector defined on  $E \cap \mathcal{N}_c$  to a vector defined on  $E$ . The specific form of all mappings involved will be given later on.

We assume that for another s.p.d. matrix  $M$ , which can be assembled from  $\{M_E\}$  in the same way as  $A$ , there is a stable local procedure that defines  $P_E$  such that for a mapping  $R_E : R_E P_E = I$ , one has the bound,

$$(3.1) \quad (\mathbf{v}_E - P_E R_E \mathbf{v}_E)^T M_E (\mathbf{v}_E - P_E R_E \mathbf{v}_E) \leq \eta_E \mathbf{v}_E^T A_E \mathbf{v}_E,$$

for any  $\mathbf{v}_E \in \mathbb{R}^n$ . We comment that  $M_E$  may equal  $A_E$ .

Note that, in general,  $P_E$  may not agree on dofs that are shared by more than one subdomain  $E$ . That is why we need a partition of unity non-negative diagonal weight matrices  $D_E = (d_{E,i})_{i \in E}$ , which are defined for vectors restricted to  $E$ . Let also  $I_E$  be the matrix representing extension by zero outside of  $E$ . It is clear that  $\mathbf{v}_E = I_E^T \mathbf{v}$  and  $M = \sum_E I_E M_E I_E^T$ . Note that we have also assumed that the row indices of  $P_E$  are in  $E$ .

Partition of unity means that

$$\sum_E I_E D_E I_E^T = I.$$

In other words,  $\sum_{E: i \in E} d_{E,i} = 1$  for every dof  $i$ . To be specific, we will choose

$$(3.2) \quad d_{E,i} = \frac{\|M_E\|}{\sum_{E': i \in E'} \|M_{E'}\|}.$$

Let  $I_E^c$  be the extension by zero of coarse vectors defined on  $E \cap \mathcal{N}_c$  to a vector defined on  $\mathcal{N}_c$ . Therefore,  $(I_E^c)^T$  restricts a coarse vector defined on  $\mathcal{N}_c$  to a coarse vector defined on  $E \cap \mathcal{N}_c$ . The following matrix definitions are then in place:

$$R^T = \begin{bmatrix} 0 \\ I \end{bmatrix} \begin{matrix} \} \mathcal{N} \setminus \mathcal{N}_c \\ \} \mathcal{N}_c \end{matrix}, \quad R_E^T = \begin{bmatrix} 0 \\ I \end{bmatrix} \begin{matrix} \} (\mathcal{N} \setminus \mathcal{N}_c) \cap E \\ \} \mathcal{N}_c \cap E \end{matrix},$$

and

$$I_E = \begin{bmatrix} 0 \\ I \end{bmatrix} \begin{matrix} \} \mathcal{N} \setminus E \\ \} E \end{matrix}, \quad I_E^c = \begin{bmatrix} 0 \\ I \end{bmatrix} \begin{matrix} \} \mathcal{N}_c \setminus E \\ \} E \cap \mathcal{N}_c \end{matrix}.$$

Finally, we assume the following form of  $P_E$ ,

$$(3.3) \quad P_E = \begin{bmatrix} * \\ I \end{bmatrix} \begin{matrix} \} E \setminus \mathcal{N}_c \\ \} E \cap \mathcal{N}_c \end{matrix}.$$

With the help of the partition of unity diagonal matrices we are in a position to define a global  $P$  as follows,

$$(3.4) \quad P = \sum_E I_E D_E P_E (I_E^c)^T.$$

The global  $Q = PR$  takes the form

$$Q = \sum_E I_E D_E P_E R_E I_E^T.$$

The latter holds since  $R_E I_E^T = (I_E^c)^T R$ . One can also see that  $RP = I$ , that is,  $Q$  is a projection.

Assuming that  $P_E$ , or rather  $Q_E = P_E R_E$ , are bounded in the  $M_E$ -energy norm in terms of  $A_E$  as in (3.1), we would like to prove a similar  $M$ -energy bound for the global  $P$ .

One has,  $\mathbf{v} - Q\mathbf{v} = \sum_E I_E D_E (I - P_E R_E) I_E^T \mathbf{v}$ . Hence, letting  $\mathbf{w}_E = (I - P_E R_E) I_E^T \mathbf{v}$ , one gets

$$\begin{aligned} \mathbf{v}^T (I - Q) M (I - Q) \mathbf{v} &= ((I - Q) \mathbf{v})^T M \sum_E I_E D_E (I - P_E R_E) I_E^T \mathbf{v} \\ &\leq \left( ((I - Q) \mathbf{v})^T M (I - Q) \mathbf{v} \right)^{\frac{1}{2}} \\ &\quad \times \left( \left( \sum_E I_E D_E \mathbf{w}_E \right)^T \sum_{E'} I_{E'} M_{E'} I_{E'}^T \sum_E I_E D_E \mathbf{w}_E \right)^{\frac{1}{2}}. \end{aligned}$$

Assume now the estimate (cf. [11] for a different application) to be verified in what follows

$$(3.5) \quad \sup_{(\mathbf{w}_E)} \frac{\sum_{E'} \|I_{E'}^T \sum_E I_E D_E \mathbf{w}_E\|_{M_{E'}}^2}{\sum_E \mathbf{w}_E^T M_E \mathbf{w}_E} \leq K.$$

Then the following bound on  $P$  (or  $Q = PR$ ) holds,

$$(3.6) \quad \begin{aligned} \mathbf{v}^T (I - Q) M (I - Q) \mathbf{v} &\leq \sum_{E'} \left( \sum_E I_E D_E \mathbf{w}_E \right)^T I_{E'} M_{E'} I_{E'}^T \sum_E I_E D_E \mathbf{w}_E \\ &\leq K \sum_E \left( (I - P_E R_E) I_E^T \mathbf{v} \right)^T M_E (I - P_E R_E) I_E^T \mathbf{v} \\ &\leq K \sum_E \eta_E \mathbf{v}^T I_E A_E I_E^T \mathbf{v} \\ &\leq K \left( \max_E \eta_E \right) \mathbf{v}^T A \mathbf{v}. \end{aligned}$$

Assuming the specific form (3.2) of the weight matrices, next we show how to estimate  $K$  in (3.5). Let  $w_i$  stand for the  $i$ th entry of  $\mathbf{w}$ . One has

$$\begin{aligned} \|I_{E'}^T \sum_E I_E D_E \mathbf{w}_E\|_{M_{E'}}^2 &\leq \|M_{E'}\| \|I_{E'}^T \sum_E I_E D_E \mathbf{w}_E\|^2 \\ &= \|M_{E'}\| \sum_{i \in E'} \left( \sum_{E: i \in E} d_{E, i} w_i \right)^2 \\ &= \sum_{i \in E'} \left( \sum_{E: i \in E} \|M_{E'}\|^{\frac{1}{2}} d_{E, i} w_i \right)^2. \end{aligned}$$

Therefore,

$$\sum_{E'} \|I_{E'}^T \sum_E I_E D_E \mathbf{w}_E\|_{M_{E'}}^2 \leq \sum_{E'} \sum_{i \in E'} \left( \sum_{E: i \in E} \|M_{E'}\|^{\frac{1}{2}} d_{E,i} w_i \right)^2.$$

Since  $\|M_{E'}\| d_{E,i} \leq \|M_E\|$  for  $i \in E \cap E'$  and  $\sum_{E: i \in E} d_{E,i} = 1$ , using Cauchy-Schwarz inequality, we get

$$\begin{aligned} \sum_{E'} \|I_{E'}^T \sum_E I_E D_E \mathbf{w}_E\|_{M_{E'}}^2 &\leq \sum_{E'} \sum_{i \in E'} \left( \sum_{E: i \in E} \|M_E\|^{\frac{1}{2}} (d_{E,i})^{\frac{1}{2}} |w_i| \right)^2 \\ &\leq \sum_{E'} \sum_{i \in E'} \sum_{E: i \in E} \|M_E\| w_i^2 \sum_{E: i \in E} d_{E,i} \\ &= \sum_{E'} \sum_{i \in E'} \sum_{E: i \in E} \|M_E\| w_i^2 \\ &= \sum_{E'} \sum_{i \in E'} \sum_{E: i \in E} \text{Cond}(M_E) \lambda_{\min}(M_E) w_i^2 \\ &\leq \max_E \text{Cond}(M_E) \kappa \sum_{i \in E} \lambda_{\min}(M_E) w_i^2 \\ &\leq \max_E \text{Cond}(M_E) \kappa \sum_E \mathbf{w}_E^T M_E \mathbf{w}_E. \end{aligned}$$

Thus,  $K \leq \kappa \max_E \text{Cond}(M_E)$  where  $\kappa \geq 1$  is the maximum number of subdomains that share a given dof, and  $\text{Cond}(M_E) = \lambda_{\max}(M_E)/\lambda_{\min}(M_E)$  denotes the condition number of  $M_E$ . In summary, one has the following main result.

**Theorem 3.1.** *Define the global  $P$  on the basis of the local ones,  $P_E$ , and the set of weighted matrices  $\{D_E\}$  given in (3.2). Assume the local estimates (3.1). Finally, let  $\kappa \geq 1$  be the maximal number of subdomains  $E$  that contain a given dof. Then, the following global norm bound holds:*

$$\mathbf{v}^T (I - Q)^T M (I - Q) \mathbf{v} \leq \kappa \max_E \text{Cond}(M_E) \left( \max_E \eta_E \right) \mathbf{v}^T A \mathbf{v}.$$

The above theorem makes sense only if  $M_E$  are non-singular. In particular, the following corollary holds (cf., [7]).

**Corollary 3.1.** *Let  $M$  be well-conditioned, for example,  $M$  be the diagonal of  $A$ , hence  $M_E$  is the diagonal of  $A_E$ . Then the local estimates (3.1) imply the global estimate in Theorem 3.1, which in turn shows that  $A$  is spectrally equivalent to  $M$  in the subspace  $\text{Range}(I - Q)$ . The latter space is complementary to the coarse space. Hence, the two-grid method based on  $P$  and a smoother  $M$  (after proper scaling, such that  $\mathbf{v}^T A \mathbf{v} \leq \mathbf{v}^T M \mathbf{v}$ ) is optimally convergent with a convergence factor bounded by  $1 - \frac{1}{\eta}$ , where  $\eta = \kappa \max_E \text{Cond}(M_E) \left( \max_E \eta_E \right)$  is the constant in the global norm estimate of  $Q$  in Theorem 3.1.*

**3.2. Energy norm bounds.** The results of the previous subsection were formulated for arbitrary  $M$ , but in practice, we are mostly interested in the case  $M = A$ , when the estimate (3.1) coincides with our assumption (2.3). Since we allow the matrices  $A_E$  to be singular, the assumption (3.5) is modified to read:

$$(3.7) \quad \sum_{E'} \|I_{E'}^T \sum_E I_E D_E \mathbf{w}_E\|_{A_{E'}}^2 \leq K \sum_E \mathbf{w}_E^T A_E \mathbf{w}_E,$$



for any  $\mathbf{w}_E$  not in the nullspace of  $A_E$ . With this modification, we can extend the “local-to-global” energy norm estimate as follows.

**Theorem 3.2.** *Define the global  $P$  on the basis of the local ones,  $P_E$ , and the set of weighted matrices  $\{D_E\}$  given in (3.2) (with  $M_E = A_E$ ). Assume the local estimates (2.3), and in particular assume that  $\mathbf{v}_E = P_E R_E \mathbf{v}_E$  for any  $\mathbf{v}_E$  in the nullspace of  $A_E$ . Finally, let  $\kappa \geq 1$  be the maximal number of subdomains  $E$  that contain a given dof. Then, the following global norm bound holds,*

$$\mathbf{v}^T (I - Q)^T A (I - Q) \mathbf{v} \leq \kappa \max_E \frac{\lambda_{\max}(A_E)}{\lambda_{\min}^+(A_E)} \left( \max_E \eta_E \right) \mathbf{v}^T A \mathbf{v},$$

where  $\lambda_{\max}(A_E)$  and  $\lambda_{\min}^+(A_E)$  denote the largest and the smallest nonzero eigenvalues of  $A_E$ .

*Proof.* First, note that, for  $\{D_E\}$  given by (3.2), (3.7) can be obtained in the same manner as (3.5), with a constant  $K \leq \kappa \max_E (\lambda_{\max}(A_E)/\lambda_{\min}^+(A_E))$ . The rest of the proof proceeds as in Theorem 3.1, taking into account that  $\mathbf{w}_E = (I - P_E R_E) I_E^T \mathbf{v}$  is not in the nullspace of  $A_E$ .  $\square$

The energy norm boundedness of  $PR$  guarantees a two-grid convergence of AMG based on  $P$  and an appropriate smoother. The following result follows from Theorem 5.1 in [8].

**Corollary 3.2.** *Under the assumptions of Theorem 3.2, the two-grid AMG method based on  $P$  and Richardson smoother, for example, has a convergence factor bounded by  $1 - \frac{\lambda_{\min}(A_{ff})}{\|A\|} \frac{1}{\eta}$ , where  $A_{ff}$  is the submatrix of  $A$  corresponding to fine degrees of freedom not present on the coarse grid, and  $\eta = \kappa \max_E \frac{\lambda_{\max}(A_E)}{\lambda_{\min}^+(A_E)} \max_E \eta_E$ .*

The rest of this subsection deals with the practical construction of interpolation which fits the null-vectors of  $A_E$  exactly. One way to do this is to first coarsen the nullspace and then proceed with the rest of the interpolation.

Specifically, let  $P_{0,E}$  be local interpolation matrices (having the form (3.3)) with columns forming a basis for the nullspace of  $A_E$ . This gives an initial coarse grid  $\mathcal{N}_c^0$  together with the restriction mapping  $R_{0,E}$  and the projection  $Q_{0,E} = P_{0,E} R_{0,E}$ . Let  $P_E$  be standard interpolation matrices, that are defined on a complementary coarse grid  $\mathcal{N}'_c$ . The resulting composite grid  $\mathcal{N}_c$  equals  $\mathcal{N}_c^0 \cup \mathcal{N}'_c$  and we assume that  $\mathcal{N}'_c \cap \mathcal{N}_c^0 = \emptyset$ . It is natural to assume that for every  $E$ ,  $R_{0,E} P_E = 0$ . In particular, we assume that  $P_E$  have zero rows corresponding to the set  $\mathcal{N}_c^0$  (viewed as a subset of the fine-grid dofs).

Define the composite interpolation

$$\bar{P}_E = [P_{0,E}, P_E],$$

and the composite restriction mapping

$$\bar{R}_E = \begin{bmatrix} R_{0,E} \\ R_E (I - P_{0,E} R_{0,E}) \end{bmatrix}.$$

Then, the following simple identity holds:

$$(3.8) \quad I - \bar{P}_E \bar{R}_E = (I - P_E R_E) (I - P_{0,E} R_{0,E}).$$

Therefore, the composite interpolation will satisfy  $\mathbf{v}_E = \overline{P}_E \overline{R}_E \mathbf{v}_E$  for any  $\mathbf{v}_E$  in the nullspace of  $A_E$ , as needed in the local estimate (2.3). Furthermore,  $\overline{Q}_E = \overline{P}_E \overline{R}_E$  is a projection, as seen below,

$$\begin{aligned} \overline{R}_E \overline{P}_E &= \begin{bmatrix} R_{0,E} \\ R_E(I - P_{0,E} R_{0,E}) \end{bmatrix} [P_{0,E}, P_E] \\ &= \begin{bmatrix} R_{0,E} P_{0,E} & R_{0,E} P_E \\ R_E(I - P_{0,E} R_{0,E}) P_{0,E} & R_E(I - P_{0,E} R_{0,E}) P_E \end{bmatrix} \\ &= \begin{bmatrix} R_{0,E} P_{0,E} & 0 \\ 0 & R_E P_E \end{bmatrix} \\ &= \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}. \end{aligned}$$

We used the fact that  $R_{0,E} P_{0,E} = I$ ,  $R_E P_E = I$  and  $R_{0,E} P_E = 0$ .

Now, let  $\overline{Q}$  be the global mapping defined using the partition of unity diagonal matrices as in (3.4). The conclusion of Theorem 3.2 is that

$$\|(I - \overline{Q}_E) \mathbf{v}_E\|_{A_E}^2 \leq \eta_E \|\mathbf{v}_E\|_{A_E}^2$$

implies

$$\|(I - \overline{Q}) \mathbf{v}\|_A^2 \leq \eta \|\mathbf{v}\|_A^2,$$

where  $\eta$  depends only on local quantities ( $\eta_E$  and the spectrum of  $A_E$ ).

The subdomain-by-subdomain construction of  $P_0$  is not always as straightforward. In the setting of element agglomeration AMGe (subdomains being agglomerated elements) algorithms to build such an initial  $P_0$  that handles the nullspace components of  $A_E$  are found in [10] and [6]. If the nullspace components are given (the constant vector for Laplace-like problems or the rigid body modes for elasticity equations), one can use the multiple vector interpolation procedure of [16] in the local setting of Section 2.

#### 4. NUMERICAL EXPERIMENTS

In this section, we consider a specific implementation of the proposed AMGe interpolation algorithm and discuss its numerical performance.

The solver is designed for unstructured finite element problems in 3D. It requires the finite element topology, i.e., the table `element_dof` (describing the set  $\{E\}$ ) and the matrices  $A_E$ , as well as one, or few, vectors that are to be interpolated exactly on the coarser grids. Those vectors can be nullspace components of the local stiffness matrices or can be the representations, in the finite element basis, of some ‘‘smooth’’ functions. For example, if a second order elliptic equation is to be solved with linear finite elements, one can provide the constant vector as an input. Similarly, the six rigid body modes can be prescribed for elasticity problems.

Given an initial finite element topology and some vectors to be interpolated, the setup phase proceeds as follows:

- (1) Based on the `element_element` connectivity (which, for example, can be computed from the `element_dof` table, see [15]), perform one step of element agglomeration, i.e., form the table `AE_element`.
- (2) Choose the new coarse grid dofs and compute the local interpolation matrices  $P_E$ .

- (3) Generate the coarse-grid finite element topology using a variational procedure (i.e. the coarse-grid matrices are computed as  $P_E^T A_E P_E$ ). Compute the coarse vectors that are to be interpolated exactly on the next step.
- (4) If the so constructed coarse grid is small enough, stop. Otherwise, use it as an input to generate the next coarse level.

The solution algorithm is then AMG, based on the global  $P$  (defined by (3.4)) and the variationally computed global coarse-grid matrices. We used two smoothers in the test runs: Gauss-Seidel and a multiplicative overlapping Schwarz with domains defined by the coarse elements.

The agglomeration step above is a graph partitioning problem which can be solved with tools such as METIS and its parallel version ParMETIS, see [12]. There are other techniques, some of which are specifically designed for AMGe settings such as Algorithm 4.1 in [10]. For a survey of different AMGe coarsening strategies, see [19]. An example of a “nice” sequence of coarse elements in three dimensions is shown in Figure 1.

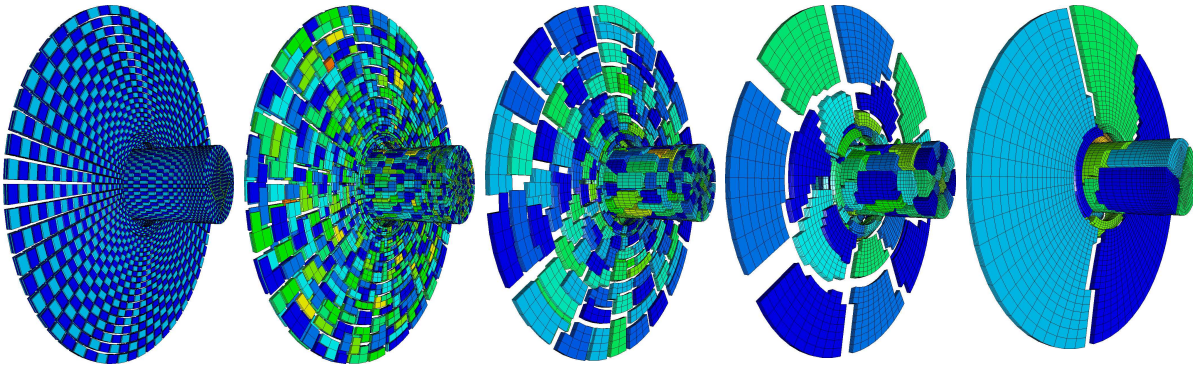


FIGURE 1. Four levels of element agglomeration with coarsening factor 8.

Note that the element agglomeration step can be completely decoupled from the rest of the setup, since the coarse grid element connectivity can be simply computed as the following Boolean matrix product:

$$\text{AE\_AE} = \text{AE\_element} \times \text{element\_element} \times (\text{AE\_element})^t.$$

The main part of the setup process is step (2), which itself consists of three steps: choosing the coarse grid (the columns of  $P_E$ ), defining the sparsity pattern of  $P_E$  and computing its actual entries. Here we use partitioning of the fine-grid dofs into groups (referred to as minimal intersection sets in [16]). These groups form a non-overlapping partition of the fine-grid dofs. They are uniquely determined after the set of the agglomerated elements in terms of the table  $\text{AE\_element}$  has been constructed. Basically, all dofs within a group belong to the same set of agglomerated elements. For an illustration of this concept see Figure 2 (center). If a vector  $\mathbf{e}$  is to be interpolated exactly, the coarse grid is chosen (or updated) by bringing one dof per group to the coarse level; namely, the dof in the group where  $|\mathbf{e}|$  is maximized (locally over the dofs in the group). This is illustrated in Figure 2. The sparsity patterns of  $P_E$  are then given by the sets  $\Omega_{E,i}$ , representing the element topology as discussed in Section 2. See Figure 3 for an example. Finally, the constrained energy minimization algorithm is used to determine the actual entries of  $P_E$ .

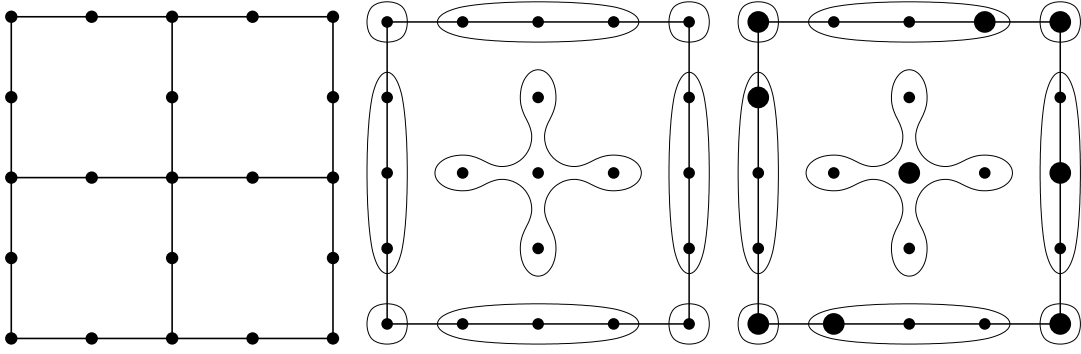


FIGURE 2. Fine elements with fine degrees of freedom to be agglomerated into one coarse element (left). Groups of degrees of freedom in the agglomerated element (center). Coarse degrees of freedom in each group (right).

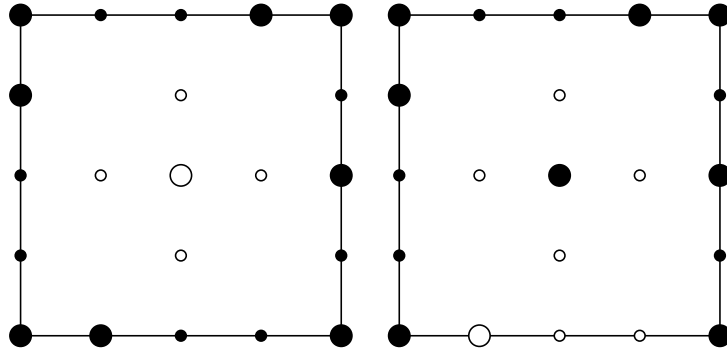


FIGURE 3. The restriction (to the given coarse element) of the support,  $\Omega_{E,i}$ , corresponding to the highlighted coarse degrees of freedom.

When more than one vector needs to be interpolated, we use the multiple vector preserving interpolation algorithm from [16]. For example, all linear functions can be brought to the coarse grid, as shown in Figure 4.

In the rest of this section, we consider numerical examples involving various PDE models, discretizations and “smooth” functions. We use an initial parallel version of the algorithm, mainly to illustrate the convergence behavior on larger problems. Making this parallel code scalable (especially in the setup phase) is a separate and ongoing effort. In each test case we list the number of processors used  $np$ , the global size of the linear system  $N$ , the commonly reported grid and operator complexities  $grid$  and  $op$ , the number of PCG iterations  $n_{it}$  needed to reduce the residual norm by six orders of magnitude, and the average convergence factor  $\rho$ .

The grid and operator complexities are standard measures for the quality of the coarsening in AMG. The former is defined as the ratio of the sum of the number of nonzero entries in the operator on all levels to the number of nonzero entries on the finest level. The latter is the sum of the grid sizes on all levels divided by the size of the finest grid.

The goal of the current paper is to demonstrate that the proposed AMGe algorithm can be a spectrally equivalent preconditioner to complicated problems, such as higher order discretizations. That is why we concentrate on the above quantities whereas the

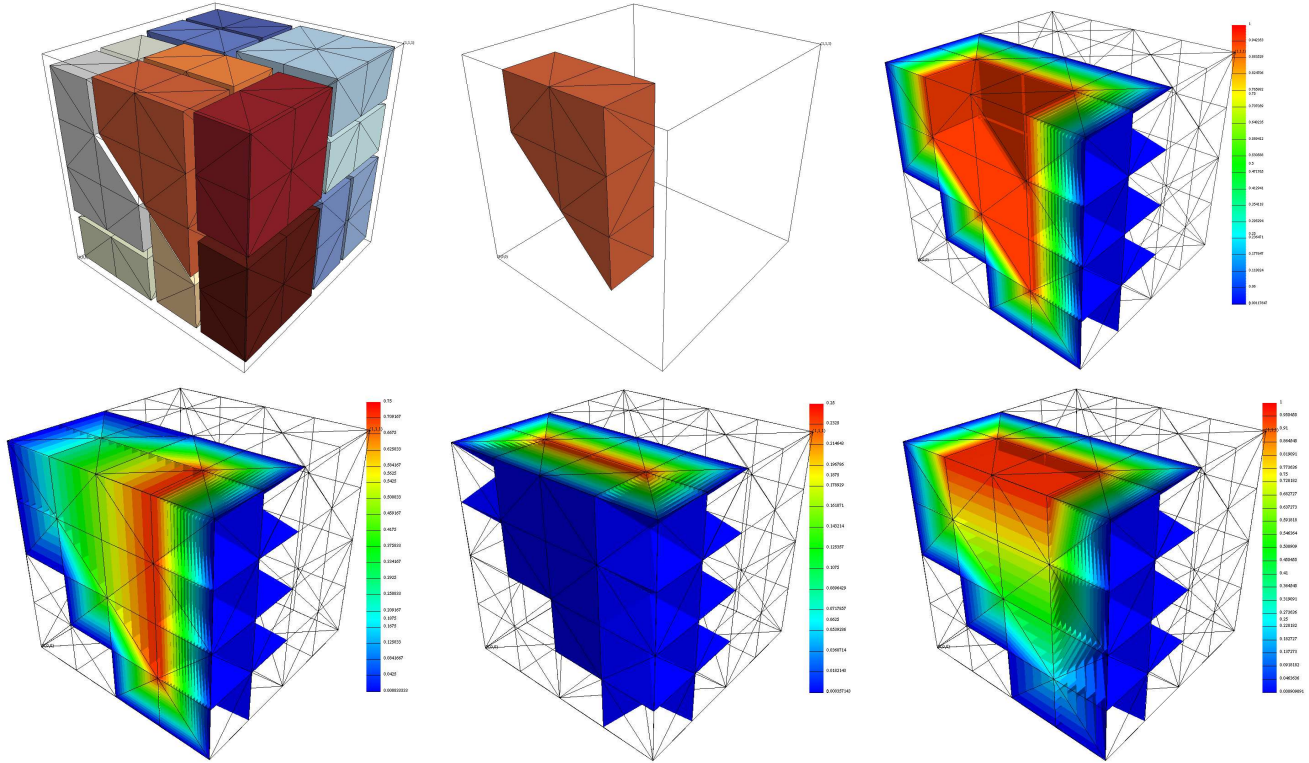


FIGURE 4. Coarse elements after two agglomeration steps (top left), together with the level surfaces corresponding to the representations of the functions 1,  $x$ ,  $y$  and  $z$  in one of them.

time for setup and solution is given as an information. In the current implementation the setup is not optimized and there is room for improvement.

Our first example problem is the Laplace equation

$$(4.1) \quad -\nabla \cdot (a(x)\nabla u) = f$$

posed on the unit cube with zero Dirichlet boundary conditions and  $a \equiv 1$ . We consider the linear system obtained from discretization with linear finite elements on a tetrahedral grid and solve it with PCG using the proposed constrained minimization AMGe as a preconditioner. The smoother is two sweeps of a parallel version of Gauss-Seidel with minres-type scaling, and the initial smooth function is the constant one. We used a coarsening factor of 8.

Since the coarse grid selection algorithm for dofs located in the vertices of the mesh leads to very large coarse grids, in this particular example, we ignored all groups associated with faces and element interiors. This leads, see Table 1, to operator complexities that are comparable with many coarsening schemes from classical AMG, cf. [9]. The number of iterations appears to be bounded, as predicted by the theory from the previous sections.

Table 1 also lists the times (in seconds) to setup and solve the problem on our test machine, having 2.4GHz Xeon processors. For this problem, both times show a slight and reasonable growth when the problem size is increased, while the size per processor is kept approximately the same.

$np$	$N$	$grid$	$op$	$n_{it}$	$\rho$	$setup$	$solve$
2	35937	2.30	5.24	3	0.01	14.4	1.62
4	68705	2.11	4.85	4	0.03	16.1	1.95
8	145505	2.06	5.15	5	0.04	20.5	2.61
16	320705	1.96	4.63	5	0.05	22.9	2.76
32	536769	2.32	4.94	6	0.08	29.8	3.67
64	1089729	2.15	5.02	7	0.12	35.3	4.15
128	2625921	1.87	4.47	7	0.12	39.2	4.41

TABLE 1. Numerical results for the Laplace problem on the unit cube discretized with linear finite elements. Reduced coarse grid by ignoring face and element groups.

We next consider a discretization of the same problem using cubic finite elements. Owing to the large size of the maximal intersection sets, we get very small grid and operator complexities, as seen in Table 2. This is in contrast to classical AMG methods, where such good complexities can be achieved only with a strength threshold very close to 1.

$np$	$N$	$grid$	$op$	$setup$
1	29449	1.22	1.35	22.3
2	60361	1.27	1.48	32.4
4	117649	1.24	1.37	28.1
8	228241	1.26	1.45	38.4
16	468625	1.29	1.51	48.1
32	912673	1.26	1.41	44.9
64	1797409	1.28	1.49	57.2
128	3692833	1.30	1.50	68.5
256	7189057	1.26	1.41	78.0
512	14266945	1.29	1.52	111.

TABLE 2. Grid, operator complexities and setup times for the Laplace problem on the unit cube, discretized with cubic finite elements.

Next, we compare two different smoother options. The first is the smoother from the previous example, while the second is a parallel Gauss-Seidel with coloring. The results in Table 3 show that Gauss-Seidel with coloring leads to better and stable number of iterations, but the increase in the number of colors needed results in worse total run times.

Finally, in Table 4, we illustrate the influence of varying the agglomeration algorithm on the same problem with an added mass term and no boundary conditions. We compare our default option—coarsening in each processor using METIS followed by processor agglomeration with ParMETIS, with two approaches similar to the bottom-up and the “rgb” methods from [19]. All methods result in solvers with comparable convergence properties, which suggests that the quality of the agglomeration may not be crucial. Note

$np$	$N$	$n_{it}$	$\varrho$	$solve$	$n_{it}$	$\varrho$	$solve$
1	29449	6	0.07	2.5	7	0.11	2.9
2	60361	7	0.12	4.8	6	0.09	7.2
4	117649	7	0.13	4.8	6	0.09	12.0
8	228241	8	0.17	5.9	7	0.12	18.1
16	468625	9	0.19	7.0	7	0.13	25.8
32	912673	8	0.17	6.8	7	0.13	27.8
64	1797409	10	0.22	8.6	8	0.15	43.7
128	3692833	10	0.24	10.3	8	0.17	63.3
256	7189057	11	0.28	11.6	9	0.18	74.6
512	14266945	13	0.33	15.4	9	0.20	104.

TABLE 3. Solve times for the Laplace problem on the unit cube discretized with cubic finite elements, using Gauss-Seidel with minres scaling (left columns) and Gauss-Seidel with coloring (right columns) smoothers.

that, while we can specify the coarsening factor (8) in METIS, the other agglomerations have smaller initial coarsening factors (around 6), which explain why their complexities are worse for this particular example.

$np$	$N$	$grid$	$op$	$setup$	$grid$	$op$	$setup$	$grid$	$op$	$setup$
1	29449	1.22	1.35	22.2	1.31	1.51	25.6	1.27	1.57	43.5
2	60361	1.27	1.47	32.5	1.31	1.51	28.1	1.26	1.45	52.8
4	117649	1.24	1.37	28.2	1.34	1.60	35.2	1.33	1.59	56.9
8	228241	1.26	1.45	38.6	1.34	1.60	41.1	1.32	1.63	86.1
16	468625	1.29	1.51	48.2	1.34	1.58	48.7	1.28	1.49	94.8
32	912673	1.26	1.41	45.3	1.37	1.66	64.1	1.35	1.62	135.
64	1797409	1.28	1.49	58.4	1.37	1.65	88.5	1.34	1.67	208.
128	3692833	1.30	1.50	67.5	1.29	1.52	103.	1.29	1.52	200.
256	7189057	1.26	1.41	78.6	1.39	1.68	124.	1.36	1.63	218.

TABLE 4. The default METIS/ParMETIS agglomeration (left) compared with two other agglomeration algorithms similar to the bottom-up approach (center) and the “rgb” method (right) from [19].

Our third example problem is, again, a discretization of (4.1), but this time posed on the hexahedral approximation of the unit ball shown in Figure 5. The coefficient  $a(x)$  is piecewise constant and has a lot of jumps. We employ trilinear finite elements for discretization, and as before, provide only the constant smooth function on input. The coarsening factor was 8, except for the finest grid, where a factor of 64 was used in order to try to reduce the size of the first coarse space. This aggressive coarsening requires a more powerful smoother, and we used a parallel version of multiplicative Schwarz with coloring.

Numerical results on up to 1024 processors are reported in Table 5, where we see that both the grid and the operator complexities are kept under control to quite acceptable

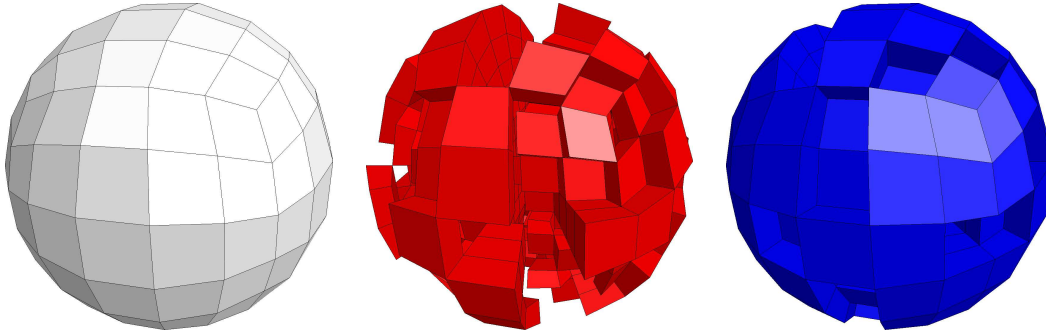


FIGURE 5. The initial hexahedral mesh approximating the unit ball (left), and its splitting into two subdomains where the coefficient  $a(x)$  has values of  $10^{-2}$  (center) and  $10^2$  (right).

levels. We also note that using the Schwarz smoother leads to low and constant number of iterations (and corresponding convergence factors).

$np$	$N$	$grid$	$op$	$n_{it}$	$\varrho$
2	7351	1.23	2.03	8	0.15
16	60483	1.30	2.71	11	0.28
128	489846	1.27	2.43	13	0.31
1024	3889285	1.26	2.30	12	0.31

TABLE 5. Numerical results for the Laplace problem on the unit ball discretized with trilinear finite elements on hexahedral grid.

Next, we consider the Lamé equations of linear elasticity, posed on the unit cube with two materials having Poisson ratios  $\nu = 0.2$  and  $\nu = 0.49$ , and Young's moduli  $E = 100$  and  $E = 1$ , see Figure 6. The matrix  $A$  corresponds to the bilinear form

$$\frac{E}{1+\nu} \int_{\Omega} \frac{\nu}{1-2\nu} \nabla \cdot \mathbf{u} \nabla \cdot \mathbf{v} + \epsilon(\mathbf{u}) : \epsilon(\mathbf{v}) \, dx,$$

where  $\mathbf{u}$  is the displacement vector and  $\epsilon(\mathbf{u}) : \epsilon(\mathbf{v}) = \frac{1}{4} \sum_{i,j=1}^3 (\partial_i \mathbf{u}_j + \partial_j \mathbf{u}_i)(\partial_i \mathbf{v}_j + \partial_j \mathbf{v}_i)$ . Both Dirichlet and Neumann boundary conditions were imposed on different parts of the boundary.

The problem was discretized with linear non-conforming finite elements on a tetrahedral grid. We choose this example since elasticity problems are usually problematic for algebraic multigrid methods, with the possible exception of smoothed aggregation, see [14, 1].

We performed two sets of tests—one using the translation vectors as initial smooth functions, and a second one where all six rigid body modes were used. In both cases, the smoother was multiplicative Schwarz. The results, presented in Table 6, suggest a definite improvement in the number of iterations when more smooth vectors are interpolated. However, this leads to significantly larger operator complexities and ultimately, slower running times. The obtained number of iterations is reasonable given that this is a fairly challenging problem.



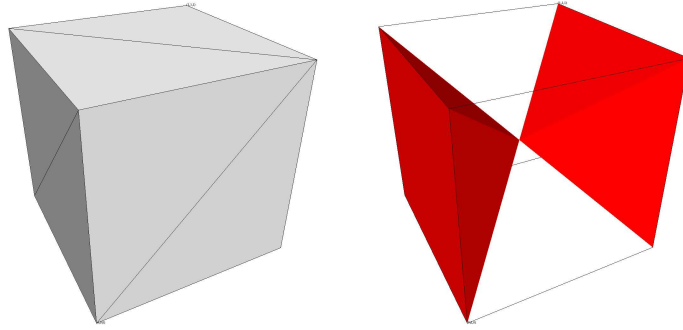


FIGURE 6. The initial tetrahedral mesh on the unit cube (left), and the subdomain where the Poisson ratio  $\nu$  has a value of 0.49 (right). The value in the rest of the domain is 0.2.

$np$	$N$	$grid$	$op$	$n_{it}$	$\rho$	$grid$	$op$	$n_{it}$	$\rho$
1	19584	1.27	1.94	26	0.58	1.49	4.11	22	0.53
2	38016	1.28	1.73	34	0.66	1.47	2.98	25	0.57
4	76032	1.33	2.17	28	0.60	1.47	3.26	21	0.50
8	152064	1.29	2.07	29	0.61	1.54	4.64	21	0.51
16	299520	1.31	1.89	37	0.69	1.52	3.45	28	0.60
32	599040	1.35	2.29	32	0.64	1.49	3.54	22	0.52
64	1198080	1.30	2.14	38	0.69	1.57	4.95	19	0.48
128	2377728	1.33	1.97	46	0.74	1.55	3.67	28	0.61
256	4755456	1.36	2.32	49	0.75	1.51	3.71	30	0.63
512	9510912	1.31	2.20	41	0.71	1.58	5.14	22	0.53
1024	18948096	1.33	1.98	47	0.74	1.55	3.69	25	0.58

TABLE 6. Numerical results for the linear elasticity problem with non-conforming finite elements. Three (left columns) and six (right columns) smooth vectors corresponding to translations and rigid body modes.

To further investigate the practical solution of this problem, we replace the Schwarz smoother with the previously used sweeps of Gauss-Seidel with minres scaling. We present the results in Table 7 and compare them with those from Table 6. As expected, we get worse number of iterations and convergence factors. However, the setup and solution times are faster and practically acceptable.

Finally, we consider the same elasticity problem discretized with (vector) cubic finite elements. The results from these experiments are reported in Table 8. We observe very good complexities and relatively stable number of iterations and convergence factors.

## REFERENCES

- [1] M. ADAMS, M. BREZINA, J. HU AND R. TUMINARO, “*Parallel multigrid smoothing: polynomial versus Gauss-Seidel*,” *Journal of Computational Physics*, **188**(2003), pp. 593–610.
- [2] M. BREZINA, A. J. CLEARY, R. D. FALGOUT, V. E. HENSON, J. E. JONES, T. A. MANTEUFEL, S. F. MCCORMICK, AND J. W. RUGE, “*Algebraic multigrid based on element interpolation (AMGe)*,” *SIAM Journal on Scientific Computing*, **22**(2000), pp. 1570-1592.

$np$	$N$	$grid$	$op$	$n_{it}$	$\varrho$	$setup$	$solve$
1	19584	1.27	1.94	40 (26)	0.71 (0.58)	17.3	3.6
2	38016	1.28	1.73	67 (34)	0.81 (0.66)	19.1	12.0
4	76032	1.33	2.17	42 (28)	0.71 (0.60)	28.1	8.2
8	152064	1.29	2.07	51 (29)	0.76 (0.61)	25.5	9.3
16	299520	1.31	1.89	79 (37)	0.84 (0.69)	32.5	17.4
32	599040	1.35	2.29	42 (32)	0.72 (0.64)	44.0	10.6
64	1198080	1.30	2.14	51 (38)	0.76 (0.69)	43.7	15.1
128	2377728	1.33	1.97	77 (46)	0.84 (0.74)	52.2	27.8
256	4755456	1.36	2.32	64 (49)	0.81 (0.75)	88.5	33.6

TABLE 7. Numerical results for the linear elasticity problem with non-conforming finite elements and Gauss-Seidel smoothing. The values in parenthesis are for multiplicative Schwarz. Smooth vectors corresponding to translations.

$np$	$N$	$grid$	$op$	$n_{it}$	$\varrho$
1	11775	1.16	1.15	16	0.41
2	24015	1.22	1.28	20	0.49
4	46875	1.22	1.29	19	0.47
8	88347	1.21	1.28	18	0.46
16	181083	1.24	1.33	24	0.54
32	352947	1.24	1.34	21	0.51
64	684723	1.24	1.36	23	0.54
128	1405875	1.27	1.45	27	0.60
256	2738019	1.25	1.35	25	0.57
512	5392227	1.28	1.46	27	0.60

TABLE 8. Numerical results for the linear elasticity problem with cubic finite elements. Three smooth vectors corresponding to translations.

- [3] M. BREZINA, R. FALGOUT, S. MACLACHLAN, T. MANTEUFFEL, S. MCCORMICK, AND J. RUGE, “Adaptive algebraic multigrid,” *SIAM Journal on Scientific Computing*, **27**(2006), pp. 1261–1286.
- [4] M. BREZINA, R. FALGOUT, S. MACLACHLAN, T. MANTEUFFEL, S. MCCORMICK, AND J. RUGE, “Adaptive smoothed aggregation (aSA),” *SIAM Journal on Scientific Computing*, **25**(2004), pp. 1896–1924.
- [5] T. CHARTIER, R. FALGOUT, V. E. HENSON, J. JONES, T. MANTEUFFEL, S. MCCORMICK, J. RUGE, AND P. S. VASSILEVSKI, “Spectral AMGe ( $\rho$ AMGe),” *SIAM Journal on Scientific Computing*, **25**(2003), pp. 1–26.
- [6] T. CHARTIER, R. FALGOUT, V. E. HENSON, J. JONES, T. MANTEUFFEL, S. MCCORMICK, J. RUGE, AND P. S. VASSILEVSKI, “Spectral element agglomerate AMGe,” *Proceedings of 16th International Conference of Domain Decomposition Methods*, New York University, 2005, Lecture Notes in Computational Science and Engineering, Springer Verlag (to appear).
- [7] R. D. FALGOUT AND P. S. VASSILEVSKI, “On generalizing the algebraic multigrid framework,” *SIAM Journal on Numerical Analysis* **42**(2004), pp. 1669–1693.
- [8] R. D. FALGOUT, P. S. VASSILEVSKI, AND L. T. ZIKATANOV, “On two-grid convergence estimates,” *Numerical Linear Algebra with Applications*, **12**(2005), pp. 471–494.

- [9] V. E. HENSON AND U. M. YANG, “*BoomerAMG: A Parallel Algebraic Multigrid Solver and Preconditioner*,” Applied Numerical Mathematics, **41**(2002), pp. 155-177. Also available as LLNL Technical Report UCRL-JC-141495.
- [10] J. E. JONES AND P. S. VASSILEVSKI, “*AMGe based on element agglomerations*,” SIAM Journal on Scientific Computing **23**(2001), pp. 109–133.
- [11] J. MANDEL, “*Balancing domain decomposition*,” Communications in Applied Numerical Methods **9**(1993), pp. 233–241.
- [12] “*METIS: Family of Multilevel Partitioning Algorithms*,” and “*ParMETIS: Parallel Graph Partitioning*,” available online at <http://www-users.cs.umn.edu/~karypis/metis/>.
- [13] A. TOSELLI AND O. WIDLUND, “*Domain Decomposition Methods - Algorithms and Theory*,” Springer-Verlag Berlin, Heidelberg, 2005.
- [14] P. VANĚK, J. MANDEL AND M. BREZINA, “*Algebraic Multigrid by Smoothed Aggregation for Second and Fourth Order Elliptic Problems*,” Computing, **3**(1996), pp. 179–196.
- [15] P. S. VASSILEVSKI, “*Sparse matrix element topology with application to AMG and preconditioning*,” Numerical Linear Algebra with Applications, **9**(2002), pp. 429–444.
- [16] P. S. VASSILEVSKI AND L. T. ZIKATANOV, “*Multiple Vector Preserving Interpolation Mappings in Algebraic Multigrid*,” SIAM Journal on Matrix Analysis and Applications, (to appear). Also available as Lawrence Livermore National Laboratory Technical Report UCRL-JRNL-208036, November 2004.
- [17] C. WAGNER, “*On the algebraic construction of multilevel transfer operators*,” Computing, **65**(2000), pp. 73–95.
- [18] W. L. WAN, T. F. CHAN AND B. SMITH, “*An energy-minimizing interpolation for robust multigrid methods*,” SIAM Journal on Scientific Computing, **21**(2000), pp. 1632–1649.
- [19] M. WABRO, “*AMGe—Coarsening Strategies and Application to the Oseen Equations*,” SIAM Journal on Scientific Computing, **27**(2006), pp. 2077–2097.
- [20] J. XU AND L. T. ZIKATANOV, “*On an energy minimizing basis in algebraic multigrid methods*,” Computing and Visualization in Science, **7**(2004), pp. 121-127.

CENTER FOR APPLIED SCIENTIFIC COMPUTING, LAWRENCE LIVERMORE NATIONAL LABORATORY,  
P.O. BOX 808, L-560, LIVERMORE, CA 94551, U.S.A.

*E-mail address:* tzanio@llnl.gov, panayot@llnl.gov