Preprint
UCRL-JC-141519

"Spectral-Element Agglomerate Coarsening in AMGe"

P. S. Vassilevski

This article was submitted to
Seminar Talk, Mathematics Department, Texas A & M University
College Station, TX
December 6, 2000

December 6, 2000

U.S. Department of Energy
Lawrence Livermore National Laboratory

"Spectral–element agglomerate coarsening in AMGe"

by

Panayot S. Vassilevski

Center for Applied Scientific Computing

UC Lawrence Livermore National Laboratory

College Station, TX, December 6, 2000

---

* Work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.

SPECTRAL–ELEMENT AGGLOMERATE COARSENING IN AMGe

PANAYOT S. VASSILEVSKI†

ABSTRACT. In this talk we will present a highly accurate coarsening algorithm for constructing coarse finite element spaces to be used in algebraic multigrid methods designed for finite element problems on generally unstructured meshes. The new algorithm relies on removing certain percentage of the high oscillating components from the spectrum of local stiffness matrices corresponding to element agglomerations. By doing so, one is guaranteed that the hierarchical complement finite element subspace gives rise to a well conditioned matrix. The coarsening consists of an agglomeration step and of computing a few minimal eigenvectors of the corresponding assembled agglomerate stiffness matrix. The method requires access to the individual element matrices. Based on the topological agglomeration algorithms we employed one is able to define coarse elements and coarse element matrices thus allowing for recursive use of the same algorithm.

Some numerical illustration for elliptic problems will also be given.

CENTER FOR APPLIED SCIENTIFIC COMPUTING, LAWRENCE LIVERMORE NATIONAL LABORATORY, 7000 EAST AVENUE, MAIL STOP 560, LIVERMORE, CA 94550

E-mail address: panayot@llnl.gov

Content

1. AMG for finite element problems; definition(s);

2. Building interpolation based on energy minimization principle;

3. Spectral–element agglomerate coarsening in AMGe;

4. Sparse element matrix topology and element agglomeration;

5. Examples of coarse elements;

6. Numerical experiments;

1. AMG for finite element problems; definition(s)

What is AMG for finite element problems ?

One needs a coarse space V_c, subspace of the original finite element space V. It is desirable that:

- the coarse space V_c admits a "weak" **approximation property**, i.e., for each $v \in V$ one should be able to find a $v_c \in V_c$ such that, for a "small" constant θ,

$$\|v - v_c\|_0 \leq \theta \, a(v, \, v)^{\frac{1}{2}} \quad \text{uniformly in } v \in V,$$

Here, $\|.\|_0$ is a weaker (typically L_2–norm) norm than the energy one.

In geometric MG, one has $\theta \simeq h_c$– the coarse mesh–size.

- A target property in AMG for finite element problems, is that the interpolation mapping P be approximate harmonic, in the sense that, for a "small" constant η, uniformly in $v_c \in V_c$, one has

$$a(v_c, \, v_c) \leq \eta \inf_{v: \, v|_c = v_c|_c} a(v, \, v).$$

In matrix–vector notation this reads:

$$\mathbf{v}_c^T A_c \mathbf{v}_c \equiv \mathbf{v}_c^T P^T A P \mathbf{v}_c \leq \eta \inf_{\mathbf{v}_f} \begin{bmatrix} \mathbf{v}_f \\ \mathbf{v}_c \end{bmatrix}^T A \begin{bmatrix} \mathbf{v}_f \\ \mathbf{v}_c \end{bmatrix}.$$

Theorem 1 (Two–grid convergence) *Assume inexact coarse solve, i.e., $B_c \simeq A_c$, such that*

$$0 \leq \mathbf{v}_c^T (B_c - A_c)\mathbf{v}_c \leq b\, \mathbf{v}_c^T A_c \mathbf{v}_c, \qquad \text{for all } v_c \in V_c.$$

Also, assume that,

$$\begin{aligned} \mathbf{v}^T \Big((D+L)D^{-1}(D+U) - A \Big) \mathbf{v} &\leq \delta \varrho(A) \|\mathbf{v}\|_0^2, \\ \mathbf{v}^T D^{-1} \mathbf{v} &\leq \gamma \mathbf{v}^T A^{-1} \mathbf{v}, \\ \mathbf{v}^T L D^{-1} U \mathbf{v} &\leq \kappa \varrho(A) \|\mathbf{v}\|_0^2, \end{aligned}$$

Here,

$$\varrho(A) = \sup_{\mathbf{v}} \frac{\mathbf{v}^T A \mathbf{v}}{\|\mathbf{v}\|_0^2}.$$

The following spectral equivalence estimate holds:

$$0 \leq \mathbf{v}^T (B_{tg} - A)\mathbf{v} \leq \max\left[2\gamma + b,\ \delta + \kappa\right] \sigma\ \mathbf{v}^T A \mathbf{v}.$$

Here, σ is a constant in the following stability estimate:

$$\varrho(A)\|\mathbf{v}_1\|_0^2 + (P\mathbf{v}_c)^T A (P\mathbf{v}_c) \leq \sigma\ \mathbf{v}^T A \mathbf{v}.$$

for a particular decomposition $\mathbf{v} = \mathbf{v}_1 + P\mathbf{v}_c$.

In the present application $\|\mathbf{v}\|_0 = \sqrt{\mathbf{v}^T A \mathbf{v}}$ (hence $\varrho(A) = 1$).

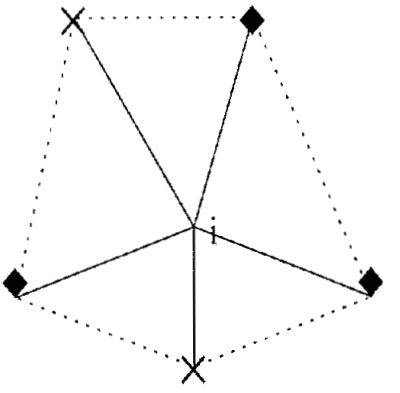
2. Building interpolation based on energy minimization principle;

2.1. Localizing estimates

In AMG one attempts to localize the estimates that provide "**approximation property**" and builds interpolation mapping which does "a best" job locally.

In AMGe, (given coarse grid) one computes P row-wise, such that

$$\sup_{\mathbf{v}} \frac{\|\mathbf{v} - (P\mathbf{v}_c)\|_0^2}{\mathbf{v}^T A_{\Omega(i)} \mathbf{v}} \mapsto \min, \quad \mathbf{v} = \begin{bmatrix} \mathbf{v}_f \\ \mathbf{v}_c \end{bmatrix}.$$



Neighborhood $\Omega(i)$ consisting of triangular elements about dof "i".

Here, $\Omega(i)$ is a mesh–neighborhood of i which contains the coarse dofs "$\diamond$" (forming the component $\mathbf{v}_c$ of $\mathbf{v}$) that are used to interpolate to i. The complementary set of dofs ("$\mathbf{i}$" and "$\mathbf{x}$") to the interpolatory coarse dofs forms the $\mathbf{v}_f$ component of $\mathbf{v}$. The matrix $A_{\Omega(i)}$ is a local matrix corresponding to the neighborhood $\Omega(i)$ and in the AMGe method it is assembled from the local element matrices of the elements that form $\Omega(i)$.

One notices, that local estimates imply global estimate.

Let, $P = \begin{bmatrix} P_{fc} \\ I \end{bmatrix}$, and $A_{\Omega(i)} = \begin{bmatrix} A_{ff} & A_{fc} \\ A_{cf} & A_{cc} \end{bmatrix}$. Then, since $\mathbf{v} - P\mathbf{v}_c = \begin{bmatrix} \mathbf{v}_f - P_{fc}\mathbf{v}_c \\ 0 \end{bmatrix}$, one gets, that P solves the following minimization problem,

$$\sup_{\mathbf{v}} \frac{\|\mathbf{v}^0\|_0^2}{\mathbf{v}^T Y^T A_{\Omega(i)} Y \mathbf{v}} \mapsto \min, \quad \mathbf{v}^0 = \begin{bmatrix} \mathbf{v}_f \\ 0 \end{bmatrix}, Y = [I,\ P].$$

Since the numerator is independent of $\mathbf{v}_c$ we can replace $\mathbf{v}_c$ with $t\,\mathbf{v}_c$, any $t \in \mathcal{R}$ and optimize with respect to t. This leads to

$$\sup_{\mathbf{v}=\begin{bmatrix} \mathbf{v}_f \\ \mathbf{v}_c \end{bmatrix}} \sup_{t} \frac{\|\mathbf{v}^0\|_0^2}{\mathbf{v}_f^T A_{ff} \mathbf{v}_f + 2t\mathbf{v}_f^T (A_{ff}P_{fc} + A_{fc})\mathbf{v}_c + t^2 \mathbf{v}_c^T P^T A_{\Omega(i)} P \mathbf{v}_c}$$

The denominator is minimized for

$$t = -\frac{\mathbf{v}_f^T (A_{ff}P_{fc} + A_{fc})\mathbf{v}_c}{\mathbf{v}_c^T P^T A_{\Omega(i)} P \mathbf{v}_c},$$

which leads to, the following identity,

$$\sup_{\mathbf{v}} \frac{\|\mathbf{v}-(P\mathbf{v}_c)\|_0^2}{\mathbf{v}^T A_{\Omega(i)}\mathbf{v}} =$$

$$\sup_{\mathbf{v}} \frac{\|\mathbf{v}^0\|_0^2}{\mathbf{v}_f^T A_{ff}\mathbf{v}_f - \frac{\left(\mathbf{v}_f^T(A_{ff}P_{fc}+A_{fc})\mathbf{v}_c\right)^2}{\mathbf{v}_c^T P^T A P \mathbf{v}_c}}, \quad \mathbf{v}^0 = \begin{bmatrix} \mathbf{v}_f \\ 0 \end{bmatrix}.$$

Then, it is clear that the best $P = \begin{bmatrix} P_{fc} \\ I \end{bmatrix}$ which solves the minimization problem, will satisfy,

$$A_{ff}P_{fc} + A_{fc} = 0.$$

I.e., $P_{fc} = -\left(A_{ff}\right)^{-1} A_{fc}$ and the interpolation weights, the row of P corresponding to dof i, are given by the ith row of $-\left(A_{ff}\right)^{-1} A_{fc}$.

We note, that in order to apply the AMGe method, the following **critical** assumption was made; namely, that one allows

"**access to the individual element matrices**" (on the fine grid).

3. Spectral–element agglomerate coarsening in AMGe

The selection of coarse dofs requires the following steps:

- an element agglomeration step;
- computing topology of the agglomerated elements, especially their vertices and faces;
- computing eigenvectors and eigenvalues of local Schur complements associated with the interior of the agglomerated elements and the interior of the faces of the agglomerated elements. The vertices of the agglomerated elements and the first few of the computed eigenvectors are the coarse dofs.
- creating the same information on the coarse level; namely, coarse elements and coarse element matrices.

Let A_E be a local matrix corresponding to an agglomerated element E. Let S_E be the Schur complement of A_E corresponding to the interior dofs of E, i.e.,

$$A_E = \begin{bmatrix} A_{II} & A_{IB} \\ A_{BI} & A_{BB} \end{bmatrix} \begin{matrix} \} \text{ interior dofs} \\ \} \text{ boundary dofs} \end{matrix}$$

Then $S_E = A_{II} - A_{IB}(A_{BB})^{-1}A_{BI}$. Let Q_k^I, $k = 1, 2, ...,$ be the eigenvectors of S_E. The vectors

$$Q_k = \begin{bmatrix} Q_k^I \\ 0 \end{bmatrix}$$

for $k = 1, 2, \ldots, q$, form part of the coarse basis.

Similarly, given a face (AEface F) one assembles a local matrix A_F of fine grid elements about the interior dofs of F. Then one computes the (interior) AEface Schur complement S_F and its first p eigenvectors $\mathbf{q}_k^I$ of S_F. The actual coarse basis vectors equal

$$\mathbf{q}_k = \begin{bmatrix} \mathbf{q}_k^I \\ 0 \end{bmatrix}.$$

Finally, one considers the vertex dofs i and forms the unit vectors $\mathbf{e}_i = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$. The coarse basis vectors are $\mathbf{e}_i$, $\mathbf{q}_k$ and Q_k Note that these are vectors of the fine–gird dimension. The coarse dofs are the coefficients in the expansion of the coarse vectors

$$\mathbf{v}_c = \sum c_i \mathbf{e}_i + \sum c_k \mathbf{q}_k + \sum c_k Q_k.$$

Note that the interpolation matrix

$$P = [(\mathbf{e}_i),\ (\mathbf{q}_k),\ (Q_k)]\,,$$

is orthogonal.

It is clear that the components $\{\mathbf{e}_i\}$, $\{\mathbf{q}_k\}$ and $\{Q_k\}$ are orthogonal; first, the three groups have non–intersecting support, and secondly within each group the vectors are orthogonal by construction (as eigenvectors of symmetric matrices).

3.2. Analysis of the two–grid method

Consider the complementary (hierarchical) set of degrees of freedom, $\{\mathbf{q}_k\}_{k>p}$ and $\{Q_k\}_{k>q}$. It spans the hierarchical complement space $\mathbf{V}_f$. The coarse space, spanned by $\{\mathbf{e}_i\}$, $\{\mathbf{q}_k\}_{k\leq p}$ and $\{Q_k\}_{k\leq q}$ is denoted by $\mathbf{V}_c$. Note that

$$\mathbf{V} = \mathbf{V}_f \oplus \mathbf{V}_c,$$

is a direct, and even orthogonal decomposition.

The restriction of A to the subspace $\mathbf{V}_f$ will be denoted by A_{ff}. We now state the first main result.

Lemma 1 *The block A_{ff} of A is well–conditioned.*

Proof: This is seen by construction. Let $\mathbf{v}_f \in \mathbf{V}_f$, that is, $\mathbf{v} = \mathbf{v}_f$ is a vector with vanishing coarse–grid component.

For some constants C_F (depending on the number of AEfaces of AEs), using well known minimization properties of Schur complements, one gets

$$
\begin{aligned}
&2\mathbf{v}_f^T A_{ff}\mathbf{v}_f = 2\mathbf{v}^T A\mathbf{v} \\
&= 2\sum_E \mathbf{v}_E^T A_E \mathbf{v}_E \\
&= \sum_E C_E \sum_{F:\ F\in\partial E} \mathbf{v}_E^T A_E \mathbf{v}_E + \sum_E \mathbf{v}_E^T A_E \mathbf{v}_E \\
&= \sum_{F\in\mathcal{F}} C_F \sum_{E:\ F\in\partial E} \mathbf{v}_E^T A_E \mathbf{v}_E + \sum_E \mathbf{v}_E^T A_E \mathbf{v}_E \\
&\geq \sum_{F\in\mathcal{F}} C_F \mathbf{v}_F^T S_F \mathbf{v}_F + \sum_E \mathbf{v}_I^T S_E \mathbf{v}_I \\
&\geq \sum_{F\in\mathcal{F}} C_F \lambda_{p+1}[S_F]\mathbf{v}_F^T\mathbf{v}_F + \sum_E \lambda_{q+1}[S_E]\mathbf{v}_I^T\mathbf{v}_I \\
&\geq \min\left\{\min_F C_F\lambda_{p+1}[S_F],\ \min_E \lambda_{q+1}[S_E]\right\}\|\mathbf{v}_f\|^2.
\end{aligned}
$$

$I = E\backslash\partial E$ is the interior of E and the components $\mathbf{v}_F$ and $\mathbf{v}_I$ of $\mathbf{v}_f$ are orthogonal; that is,

$$
\|\,\mathbf{v}_f\big|_E\,\|^2_{I\cup\partial E} = \sum_{F\in\partial E} \|\mathbf{v}_F\|^2_F + \|\mathbf{v}_f\|^2_I.
$$

This implies,

$$
\begin{aligned}
&\min\left\{\min_F C_F\lambda_{p+1}[S_F],\ \min_E \lambda_{q+1}[S_E]\right\}\|\mathbf{v}_f\|^2 \\
&\leq 2\mathbf{v}_f^T A_{ff}\mathbf{v}_f \\
&\leq 4\max_E \|A_E\|\|\mathbf{v}_f\|^2.
\end{aligned}
$$

The constants C_E and $C_F = \min\left\{C_{E_-}, C_{E_+}\right\}$, where $F = \partial E_- \cap \partial E_+$ (F is formed from the two adjacent AEs, E_- and E_+), depend only on the topology of the AEs. More specifically $\frac{1}{C_E}$ equals the number of AEfaces of E.

$\diamond$

Remark 1 *The above proof actually shows that if one diagonally scales A_{ff} (corresponding to the AEface–interior dofs and the AE–interior dofs) the resulting matrix will have condition number bounded by*

$$\sup_E \|A_E\| \left(\min\left\{\min_{F\in\partial E} \frac{1}{2} C_F \lambda_{p+1}[S_F],\ \lambda_{q+1}[S_E]\right\}\right)^{-1}.$$

Lemma 2 *The following approximately harmonic properties hold:*

$$\begin{aligned}
\mathbf{v}_f^T A \mathbf{v}_f &\leq C_A \inf_{\mathbf{v}_c \in \mathbf{V}_c} (\mathbf{v}_f + \mathbf{v}_c)^T A (\mathbf{v}_f + \mathbf{v}_c), \quad \mathbf{v}_f \in \mathbf{V}_f \\
\mathbf{v}_c^T A \mathbf{v}_c &\leq C_A \inf_{\mathbf{v}_f \in \mathbf{V}_f} (\mathbf{v}_f + \mathbf{v}_c)^T A (\mathbf{v}_f + \mathbf{v}_c), \quad \mathbf{v}_c \in \mathbf{V}_c.
\end{aligned}$$

Proof: The above two inequalities are equivalent to each other, since they are both equivalent to the following strengthened Cauchy–Schwarz inequality,

$$\mathbf{v}_f^T A \mathbf{v}_c \leq \left(1 - \frac{1}{C_A}\right)^{\frac{1}{2}} \left[\mathbf{v}_f^T A \mathbf{v}_f\right]^{\frac{1}{2}} \left[\mathbf{v}_c^T A \mathbf{v}_c\right]^{\frac{1}{2}}.$$

That is why, we prove the first one only. One has, for each AE E and its element matrix A_E, by rearranging the terms and using basic minimization properties of Schur complements of symmetric

positive semi–definite matrices,

$$
\begin{aligned}
&(\mathbf{v}_c+\mathbf{v}_f)^T A(\mathbf{v}_c+\mathbf{v}_f)\\
&=\sum_E (\mathbf{v}_c+\mathbf{v}_f)^T A_E(\mathbf{v}_c+\mathbf{v}_f)\\
&\geq \sum_F C_F(\mathbf{v}_c+\mathbf{v}_f)^T S_F(\mathbf{v}_c+\mathbf{v}_f)\\
&\qquad +\frac{1}{2}\sum_E (\mathbf{v}_c+\mathbf{v}_f)^T S_E(\mathbf{v}_c+\mathbf{v}_f)\\
&=\sum_F C_F(\mathbf{v}_f)^T S_F\mathbf{v}_f+\frac{1}{2}\sum_E (\mathbf{v}_f)^T S_E\mathbf{v}_f\\
&\qquad\qquad +\sum_F C_F(\mathbf{v}_c)^T S_F\mathbf{v}_c+\frac{1}{2}\sum_E (\mathbf{v}_c)^T S_E\mathbf{v}_c.
\end{aligned}
$$

We used here the fact that the components $\mathbf{v}_f$ and $\mathbf{v}_c$ restricted to AEface–interior and AE–interior are S_F or S_E-orthogonal, respectively.

By omitting the terms involving $\mathbf{v}_c$ (which are non–negative) one ends up with the estimate,

$$
\begin{aligned}
&(\mathbf{v}_c + \mathbf{v}_f)^T A (\mathbf{v}_c + \mathbf{v}_f) \\
&\geq \sum_F C_F (\mathbf{v}_f)^T S_F \mathbf{v}_f + \frac{1}{2} \sum_E (\mathbf{v}_f)^T S_E \mathbf{v}_f \\
&\geq \sum_F C_F \lambda_{p+1}[S_F] \|\mathbf{v}_f\|_F^2 + \frac{1}{2} \sum_E \lambda_{q+1}[S_E] \|\mathbf{v}_f\|_E^2 \\
&\geq \mu \sum_E \left[\lambda_{q+1}[S_E] \|\mathbf{v}_f\|_E^2 + \sum_{F \in \partial E} C_F \lambda_{p+1}[S_F] \|\mathbf{v}_f\|_F^2 \right] \\
&\geq \mu \min_E \left[\min \left\{ \min_{F \in \partial E} C_F \lambda_{p+1}[S_F],\ \lambda_{q+1}[S_E] \right\} \|A_E\|^{-1} \right. \\
&\qquad \left. \times \sum_E \|A_E\| \|\mathbf{v}_f\|_{E \cup \partial E}^2 \right. \\
&\geq \mu \min_E \left[\min \left\{ \min_{F \in \partial E} C_F \lambda_{p+1}[S_F],\ \lambda_{q+1}[S_E] \right\} \|A_E\|^{-1} \right. \\
&\qquad \left. \times \sum_E \mathbf{v}_f^T A_E \mathbf{v}_f \right. \\
&= \mu \min_E \left[\min \left\{ \min_{F \in \partial E} C_F \lambda_{p+1}[S_F],\ \lambda_{q+1}[S_E] \right\} \|A_E\|^{-1} \right. \\
&\qquad \left. \times (\mathbf{v}_f)^T A \mathbf{v}_f. \right.
\end{aligned}
$$

Again, as in Lemma 1, the constants here, μ and C_F, depend only on the topology of the AEs.

$\diamond$

The following results concerns the choice of smoother

Lemma 3 *Consider B_{ff} the symmetric Gauss–Seidel preconditioner to A. Then the following inequality holds for any $\mathbf{v}_f \in \mathbf{V}_f$*

$$0 \leq \mathbf{v}_f^T (B_{ff} - A)\mathbf{v}_f \leq C\, \mathbf{v}^T A \mathbf{v}, \quad \mathbf{v} = \mathbf{v}_f + \mathbf{v}_c, \text{ for all } \mathbf{v}_c$$

where the constant $C = \sigma C_A$; C_A is from Lemma 2 and σ depends on the condition number of A_{ff}.

Then, on the general two–grid convergence result shown earlier holds based on the three Lemmas 2, 1 and 3.

The spectral AMGe can be viewed also as a simple change of basis to a more stable two–level hierarchical one (which also provides orthogonal decomposition of $\mathbf{V} = \mathbf{V}_f \oplus \mathbf{V}_c$). Thus a further better interpolation may be needed, after the change of basis has been performed.

4. Sparse element matrix topology and element agglomeration

4.1. Main definitions and constructions

By definition, an element is a "list of degrees of freedom", $e = \{d_1, \ldots, d_{n_e}\}$,

and we are given an overlapping partition $\{e\}$ of $\mathcal{D}$ (the set of degrees of freedom).

In AMGe, each element is associated with an element matrix A_e, an $n_e \times n_e$ matrix, which here we assume to be symmetric and positive semi–definite.

The global matrix A is assembled from the individual element matrices A_e in the usual way, i.e.,

$$\mathbf{w}^T A \mathbf{v} = \sum_e \mathbf{w}_e^T A_e \mathbf{v}_e.$$

Here, $\mathbf{v}_e = \mathbf{v}|_e$, i.e., restriction to subset ($e \subset \mathcal{D}$).

In general AMG, we will only need the "**element topology**", and not the element matrices.

4.2. Graph based algorithms for coarsening

The coarsening algorithm based on the "sparse matrix element topology" is a straightforward extension of the standard finite element method, now in a graph setting.

Namely, given the following graph,

"**element_node**"

which is the incidence sparse matrix "element" i (rows) contains "node" j (columns),

i.e., it is the rectangular sparse matrix

Element_Node of ones and zeros, of size

(number of elements) $\times$ (number of nodes).

("node" means a number of degrees of freedom in order to handle systems of PDEs).

The incidence

"node" i belongs to "element" j,

is given by the transpose of the above rectangular sparse matrix, i.e.,

Node_Element = (Element_Node)^T.

One can consider a number of useful graphs (easily computable)

"element_element" =

$$= \text{"}\textbf{element\_node}\text{"} \times \text{"}\textbf{node\_element}\text{"},$$

"node_node" = **"node_element"** × **"element_node"**,

The first one shows the incidence

"element" i intersects "element" j,

whereas the second one shows the sparsity pattern of the assembled matrix, namely,

"node" i is connected to "node" j (hence the entry $a_{i,j}$ is potentially non–zero).

4.3. Element faces

It is typical that a finite element mesh generator can provide the element topology, namely,

"**element_face**", "**face_element**", "**face_node**", "**face_face**", etc.

If these are not available, one can actually compute them as follows.

The notion of "**face**" (similar to standard elements) is defined as a maximal intersection set, i.e., consider all intersections

$e_1 \cap e_2$, $e_1 \neq e_2$.

A face is a maximal intersection set of the above type, or a maximal intersection set of the type

$e \cap$ "boundary surface".

(if special lists of nodes is given, that provide additional information about the domain boundary)

If some of the graphs are not available one can construct all of the following

"**element_face**", "**face_element**", "**face_node**", "**face_face**", etc.

based on the symbolic part of the sparse–matrix operations. For example,

"**face_face**" = "**face_node**" × "**node_face**".

4.4. Element agglomeration

The topological information is used to devise an algorithm to agglomerate elements – a new overlapping partition $\{E\}$ of $\mathcal{D}$ where each $E = e_1 \cup e_2 \ldots \cup e_p$, i.e., to build the new graph

"**AE_element**" where **AE** stands for "agglomerated element".

The following algorithm has the motivation to generate "quasiuniform" "AE"s. In particular, it will restore coarse rectangular or triangular elements (up to boundary effects).

Algorithm 1 (Agglomeration of elements)

Given the graphs

*"***face_face***", "***element_face***" and "***face_element***"*

and a weight function $w(f) = 0$, f-face, one performs:

1. *find a face f with maximal $w(f) \geq 0$, then set $w(f) = -1$ and add on the list of the current "AE" the elements e_1 and e_2 such that $f = e_1 \cap e_2$.*

2. *update $w(f_1)$ for all f_1 connected to f (based on the graph "***face_face***"), according to the following topological rule, $w(f_1) := w(f_1) + 1$ if f_1 is connected to f and once more $w(f_1) := w(f_1) + 1$ if f_1 and f belong to a same element (here one uses the graph "***face_element***");*

3. *if for all faces f_1 of the already agglomerated elements e (here one uses the graph "***element_face***") in the current "AE" $w(f_1)$ is less than $w(f)$ where f was the last eliminated face, the agglomeration procedure for the current "AE" is terminated. Then, go to step 1 or stop.*

4.5. Generating coarse graphs needed for recursion

Assume that the graph "**AE_element**" has been constructed (somehow), then one can build

"**AE_face**" = "**AE_element**" × "**element_face**". I.e., the AE in terms of faces of the elements.

Now, one can define faces of agglomerated elements, "AEface"s, based on "**AE_face**"; namely, each "AE" has a number of faces of the original elements. That is, we have the lists "AE" – "faces".

Intersecting two different lists, one gets the faces of the "AE"s in terms of the faces of the original elements.

That is, one defines the new graphs

"**AEface_face**", and "**AE_AEface**".

Then to exploit recursion in the agglomeration algorithm one sets:

"**element_face**" := "**AE_AEface**",

"**face_element**" := "**AEface_AE**",

"**face_face**" : = "**AEface_face**" × "**face_face**" × ("**AEface_face**")^T.

As one can see no nodal information is needed.

If nodal information is needed, one can for example compute "**AEface_node**" = "**AEface_face**" × "**face_node**".

4.6. Vertices

Geometrically, vertices (in 2D) are nodes that belong to two (or more) "AEface"s. For this we need the graph

$$\text{"}\mathbf{node\_AEface}\text{"} = (\text{"}\mathbf{AEface\_node}\text{"})^T,$$

which can be computed as the transpose of

"**AEface_node**".

A formal definition of a vertex is a minimal intersection set of the type

$$V(i) = \cap\{\text{"}\mathbf{AEface}\text{"} : i \in \text{"}\mathbf{AEface}\text{"}\}$$

Definition 1 (Coarse nodes) *A minimal set of coarse nodes $\mathcal{N}_c \subset \mathcal{N}$ is provided by the vertices of the "AE"s, i.e., one forms the graph* "**node_coarsenode**";

Then one can construct,

"**coarseelement_coarsenode**" =

"**AE_node**" $\times$ "**node_coarsenode**";

"**coarseelement_coarseface**" = "**AE_AEface**";

"**coarseface_coarsenode**" =

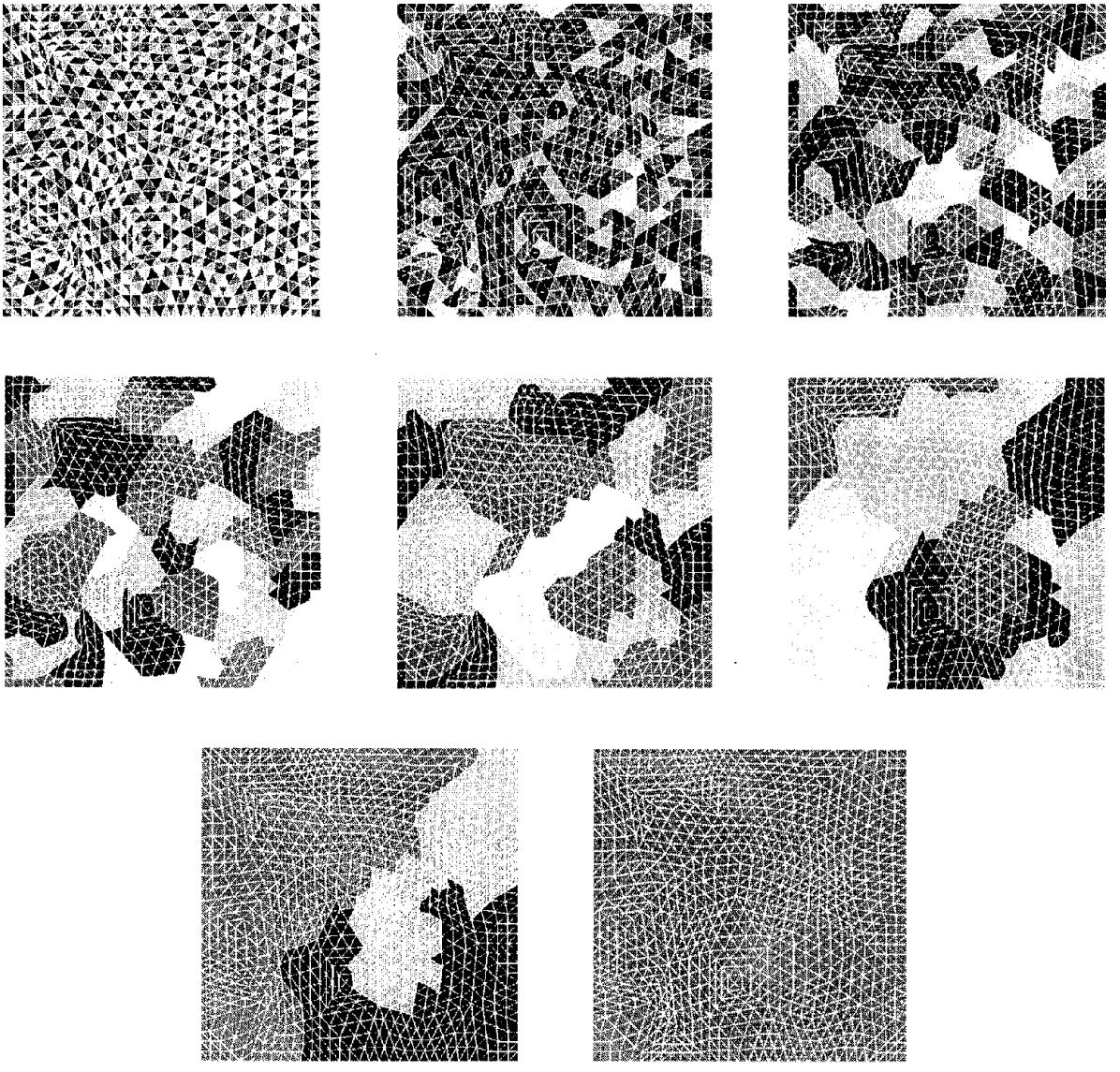
"**AEface_node**" $\times$ "**node_coarsenode**";

and hence be able to continue coarsening by recursion.

Note that the "topological" coarsening can be independently performed (since no nodal information was needed) and then the coarse node selection can be carried out.

The minimal coarse sets (based on the vertices) can be augmented by adding more nodes on the faces at given coarsening level.

5. Examples of coarse elements



Sequence of increasingly coarse elements, formed by element agglomeration.

6. Numerical Results

- anisotropic problems;

$$-\ \text{div}\ (\epsilon I + \underline{b}\underline{b}^T)\nabla u = f, \quad \text{posed in } \Omega \subset \mathcal{R}^2, \tag{1}$$

where $\epsilon \leq 1$ and $\underline{b} = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}$ for varying θ.

- 3d elasticity problems:

One notices that the construction of P is local based on local interpolation matrices P_E computed AE–wise.

Thus one can define coarse elements by performing

$$P_E^T A_E P_E;$$

and

$$P_E = [(\mathbf{e}_i),\ (\mathbf{q}_k),\ (Q_k)]_E\,.$$

Here, $p = p_F$ and $q = q_E$ are chosen such that for a prescribed tolerance $\theta \in (0, 1]$,

$\lambda_{\max}[S_F]\theta \leq \lambda_{p+1}[S_F]$ and

$\lambda_{\max}[S_E]\theta \leq \lambda_{q+1}[S_E]$.

Iteration counts and convergence factors for spectral AMGe; unstructured triangular fine grid, anisotropic Poisson equation, $\epsilon = 0.001$. Tolerance $\theta = \frac{1}{16} = 0.0625$.

Table 1

# elements	296	876	2 435	6 452
# dofs	169	466	1 256	3281
# levels	5	6	7	9
# iterations	7	8	9	11
ϱ	0.11	0.16	0.18	0.26
grid complexity	2.40	2.67	2.80	2.85
operator complexity	3.65	4.51	4.98	5.23

Iteration counts, coarsening history and convergence factors for spectral AMGe; unstructured triangular fine grid, anisotropic Poisson equation; $\epsilon = 0.001$. Tolerance $\theta = 0.25 = \frac{1}{4}$.

Table 2

level #	*# elm*	*# nodes*	*nnz entrie*
0	*6452*	*3281*	*22745*
1	*1398*	*3237*	*42043*
2	*369*	*1916*	*37924*
3	*141*	*955*	*21843*
4	*61*	*464*	*11484*
5	*27*	*212*	*5000*
6	*11*	*126*	*3214*
7	*4*	*55*	*1169*
8	*2*	*29*	*503*
grid compl.:	*3.131*	*operator compl.:*	*6.415*
# iter:	*11*	*conv. factor:*	*0.253*

Iteration counts, coarsening history and convergence factors for spectral AMGe; unstructured triangular fine grid, anisotropic Poisson equation; $\epsilon = 0.001$. Tolerance $\theta = 0.5 = \frac{1}{2}$.

Table 3

level #	*# elms*	*# nodes*	*nnz entries*
0	*6452*	*3281*	*22745*
1	*1398*	*3237*	*42043*
2	*369*	*2236*	*48296*
3	*145*	*1512*	*43880*
4	*60*	*1024*	*41748*
5	*26*	*757*	*40639*
6	*11*	*598*	*52546*
7	*5*	*507*	*71425*
8	*2*	*481*	*154405*
grid compl:	*4.155*	*operator compl:*	*22.76*
# iter:	*10*	*conv. factor:*	*0.233*

PCG convergence results: ALE3d elasticity problem; $V(1,1)$-cycle MG (Gauß–Seidel smoother). Tolerance: $\theta = 0.03125 = \frac{1}{32}$.

Table 4

level #	*# dofs*	*# elements*	*nnz*
0	*4191*	*1080*	*286533*
1	*3210*	*250*	*494352*
2	*1798*	*54*	*562838*
3	*1096*	*12*	*517230*
4	*781*	*4*	*549777*
5	*671*	*2*	*431017*
grid compl:	*2.802*	*operator compl:*	*9.917*
# iter:	*20*	*conv. factor:*	*0.499*

PCG convergence results: ALE3d elasticity problem; $V(1,1)$-cycle MG (Gauß–Seidel smoother). Tolerance: $\theta = 0.03125 = \frac{1}{32}$.

Table 5

level #	*# dofs*	*# elements*	*nnz*
0	*1638*	*375*	*103824*
1	*1067*	*90*	*133489*
2	*619*	*21*	*133273*
3	*376*	*6*	*98372*
4	*252*	*2*	*52114*
grid compl:	*2.412*	*operator compl:*	*5.018*
# iter:	*19*	*conv. factor:*	*0.479*

7. References

1. A. Brandt, S. McCormick and J. W. Ruge, "Algebraic multigrid (AMG) for sparse equations", in: (D. J. Evans, ed.) *Sparsity and its applications (Loughborough 1983)*, Cambridge Univ. Press, Cambridge, 1984, pp. 257-284.

2. A. Brandt, "General highly accurate algebraic coarsening", Electronic Transactions on Numerical Analysis **10**(2000), 1–20.

3. M. Brezina, A. J. Cleary, R. D. Falgout, V. E. Henson, J. E. Jones, T. A. Manteuffel, S. F. McCormick, and J. W. Ruge, "Algebraic Multigrid Based on Element Interpolation (AMGe)", SIAM J. Sci. Comput., to appear.
http://www.llnl.gov/CASC/linear_solvers/pubs.html

4. V. E. Henson and P. S. Vassilevski, "Element-free AMGe: General algorithms for computing interpolation weights in AMG", preprint,
http://www.llnl.gov/CASC/linear_solvers/pubs.html

5. J. E. Jones and P. S. Vassilevski, "AMGe based on element agglomeration", SIAM J. Sci. Comput. (to appear),
http://www.llnl.gov/CASC/linear_solvers/pubs.html

6. J. W. Ruge and K. Stüben, "Algebraic multigrid", in: (S. McCormick, ed.) *Multigrid Methods*, Philadelphia, PA, 1987, pp. 73–130

7. K. Stüben, "Algebraic Multigrid (AMG): An Introduction with Applications", GMD Report **53**(1999), GMD - Forschungszentrum Informationstechnik GmbH, Schloss Birlinghoven, Sankt Augustin, Germany.