

# Smoothed Aggregation Solvers for Anisotropic Diffusion

Jacob B. Schroder<sup>†</sup>

Student Paper Competition of the 2011 Copper Mountain Conference

## Abstract

A smoothed aggregation-based algebraic multigrid solver for anisotropic diffusion problems is presented. Algebraic multigrid (AMG) is a popular and effective method for solving sparse linear systems that arise from discretizing partial differential equations. However, while AMG was designed for elliptic problems, the case of non-grid-aligned anisotropic diffusion is not adequately addressed by existing methods. To achieve scalable performance, it is shown that neither new coarsening nor new relaxation strategies are necessary. Instead, a novel smoothed aggregation approach is developed which combines long-distance interpolation, coarse-grid injection, and an energy-minimization strategy that finds the interpolation weights. A measure is proposed that tracks “closeness” to the ideal interpolant and guides the sparsity pattern choice. The level of “closeness” needed is determined by experiments with grid-aligned anisotropic diffusion, which is known to produce scalable AMG solvers. This strategy is a first step towards a general approach for optimizing interpolation sparsity patterns. The paper concludes with encouraging numerical results, that also potentially show broad applicability, e.g., elasticity.

## 1 Introduction

Algebraic multigrid (AMG), e.g., classical AMG [5, 16] and smoothed aggregation (SA) [19, 18], are efficient solution methods for the large, sparse linear systems that arise from discretizing partial differential equations (PDEs). Classical multigrid methods were developed for symmetric positive-definite (SPD) systems that arise from elliptic PDEs; however, not all elliptic PDEs are adequately addressed by existing multigrid methods. For example, non-grid-aligned anisotropic diffusion is known to be problematic for even simple model problems, such as 2D rotated anisotropic diffusion defined by

$$-(c^2 + \epsilon s^2)u_{xx} - 2(1 - \epsilon)cs u_{xy} - (\epsilon c^2 + s^2)u_{yy} = f, \quad (1)$$

where  $\epsilon = 0.001$ ,  $c = \cos(\theta)$ ,  $s = \sin(\theta)$ , and  $\theta$  is the angle of rotation. When (1) is discretized with bilinear (Q1) finite elements on a regular grid of the unit box, the results using SA are disappointing, as shown in Table 1, where the conjugate gradient method (PCG) is accelerated by V(1,1)-cycles using symmetric Gauss-Seidel. The strength-of-connection drop-tolerance has been tuned for the selected  $\theta$ . To indicate scalability, the table entries  $a, b$  refer to quantities from experiments with a  $127 \times 127$  and  $1023 \times 1023$  grid, respectively.  $\gamma$  is the residual reduction factor, i.e., the factor by which the residual is reduced each PCG step. “Op. Comp.” is the operator complexity, i.e., the total number of nonzeros for each matrix in the multilevel hierarchy, relative to the number of nonzeros for the finest-level matrix. “Work” refers to the work per digit-of-accuracy, e.g., 6 work units indicates work equal to 6 finest-level matrix-vector products in order to reduce the residual by one order of magnitude. Scalability only occurs for grid-aligned anisotropy, (i.e.,  $\theta = 0$ ), thus motivating the search for a better solver. Scalability is achieved when work does not increase with the matrix size.

One possibility for improvement is more robust coarsening strategies [6, 14]. For the grid-aligned case, it is well-known that a more robust strategy semi-coarsens in the direction

---

<sup>†</sup>Department of Applied Mathematics, University of Colorado at Boulder, UCB 526, Boulder, CO, 80309, U.S.A. [jacob.schroder@colorado.edu](mailto:jacob.schroder@colorado.edu)

$\theta$	0	$\pi/16$	$2\pi/16$	$3\pi/16$	$4\pi/16$
$\gamma$	0.05, 0.05	0.21, 0.35	0.26, 0.39	0.36, 0.54	0.20, 0.40
Op. Comp.	1.5, 1.5	1.4, 1.5	1.6, 1.6	1.4, 1.4	1.8, 1.8
Work	5, 6	8, 13	11, 16	12, 21	10, 18

Table 1: SA Results, Model Rotated Anisotropic Diffusion,  
 Entries  $a, b$  Refer to Experiments with a  $127 \times 127$  and  $1023 \times 1023$  Grid, Respectively.

of strong diffusion and yields scalability, (e.g.,  $\theta = 0$ ). These advanced coarsening strategies target scalability by intelligently coarsening in the non-grid-aligned case. However, this has proved only a partial fix, as evidenced in Table 1 where [14] is used.

Another robust coarsening strategy is compatible relaxation (CR) [3], which has been applied to anisotropic diffusion problems while being coupled to energy-minimizing interpolation strategies [7, 13]. However, these two approaches, while offering improvement, also leave scalability for the non-grid-aligned case unresolved.

Geometric multigrid methods offer another path to scalability for some anisotropic problems through the use of line relaxation. In the algebraic setting, it is also possible to construct “lines” based on a strength-of-connection concept. However, because the focus of this paper is on purely algebraic methods, and because of the difficulty of effectively parallelizing line smoothers, these options are not considered. Another geometric option is AMGe [11], which exhibits improved performance for anisotropic problems, but requires availability of element stiffness matrices, which is not presumed here.

Thus, the current state does not provide an algebraic and scalable scheme for non-grid-aligned anisotropic diffusion. Given the broad need for such solvers, (e.g., Poisson operators on stretched meshes and computational fluid dynamics [12, 15]), this paper explores a promising new direction. Scalable multigrid relies on the complementary relationship between relaxation and interpolation. Therefore, the lack of scalability must be the result of some combination of inadequate relaxation, coarsening, or interpolation. Given that better coarsening has not yielded scalability and that line relaxation, as mentioned above, is impractical, this paper instead focuses on improving interpolation, with the following specific contributions. A measure developed here tracks “closeness” to the ideal interpolant [9] in order to guide the choice for the interpolation sparsity pattern. The result is a novel long-distance interpolation approach, coupled with coarse-grid injection, where the interpolation weights are determined by the energy-minimizing prolongation smoother [13]. The level of “closeness” needed is determined by comparison to interpolation for the grid-aligned case, which is known to produce scalable AMG solvers. Once the ideal interpolant is adequately approximated, the numerical results indicate that improved interpolation is sufficient for scalability, i.e., neither relaxation nor coarsening need improvement. Overall, the process outlined is a first step towards a general approach for optimizing interpolation sparsity patterns.

In Section 2, a brief overview of SA is given. In Section 3, the new interpolation strategy is motivated and then specified. In Section 4, supporting numerical results are given. In Section 5, future work is discussed and, in Section 6, concluding remarks are made.

## 2 Smoothed Aggregation Overview

An overview of SA is now given. The presentation follows standard SA, but uses newer techniques such as the energy-minimizing prolongation smoother [13] and the strength measure [14]. Also, a root-node perspective is employed to induce coarse-level injection.

Generally speaking, multigrid automatically constructs a hierarchy of coarse sets of degrees-of-freedom and interpolation operators. When relaxation and interpolation are complementary, i.e., when interpolation is accurate for modes invariant to relaxation, rapid convergence occurs. The high energy, (or algebraically smooth), error left by relaxation is accurately interpolated to the next coarser level, where it is effectively reduced. Optimality

occurs when this procedure is made recursive, with a coarsest-level of trivial size.

Here, standard V-cycles are used for the recursive solve phase. First, pre-smoothing, (e.g., Gauss-Seidel or weighted-Jacobi), occurs on the finest-level to reduce high-energy error. The residual equation is then interpolated to the first coarse-level and solved recursively. The recursive process halts once the coarsest level has been reached, and a direct solve is done. The fine-level solution is then updated using the interpolated error correction from the coarse-level. Finally, a step of post-smoothing occurs.

The SA setup phase is outlined in Algorithm 1. Level 0 is the finest-level;  $P_\ell$  denotes prolongation from level  $\ell + 1$  to level  $\ell$ ;  $P_\ell^*$  denotes restriction from level  $\ell$  to level  $\ell + 1$ ; and  $A_\ell$  is the operator on level  $\ell$ . Subscripts are dropped when the discussion is invariant to the level. SA begins with a set of provided near null-space modes,  $B_0$ , which are typically rough representations of the true near null-space. Therefore,  $B$  is pre-smoothed in Algorithm 1 for improvement, especially near the domain boundaries.

Next, the coarse-grid is constructed by aggregation, which corresponds to a non-overlapping covering,  $\mathcal{N}$ , of the matrix graph, i.e., each vertex is assigned to an aggregate, typically through a greedy procedure. The graph of  $A$  may be used, but it is more robust to use the graph of a strength-of-connection matrix  $S$ .  $S$  is based on  $A$ , but reflects the removal of entries  $(i, j)$ , between which algebraically smooth error cannot be well-approximated with  $B$  [14]. Those  $(i, j)$  removed are dubbed weak and those remaining strong.

Aggregation is then used to induce a tentative prolongation,  $T$ , that exactly interpolates  $B$  with  $T_\ell B_{\ell+1} = B_\ell$ . This is achieved by injecting  $B$  into the aggregation pattern,  $\mathcal{N}$ , i.e.,

$$T_\ell = \begin{bmatrix} Q^{(1)} & 0 & \dots & \\ 0 & Q^{(2)} & 0 & \dots \\ & & \ddots & \\ & \dots & 0 & Q^{(k)} \end{bmatrix}, \quad B_{\ell+1} = \begin{bmatrix} R^{(1)} \\ R^{(2)} \\ \vdots \\ R^{(k)} \end{bmatrix}, \quad (2)$$

where  $B^{(i)}$  is a submatrix of  $B_\ell$  obtained by taking only rows assigned to the  $i^{th}$  aggregate and  $B^{(i)} = Q^{(i)} R^{(i)}$  via a QR factorization. Thus,  $T$  is block diagonal, where the blocks correspond to only those degrees-of-freedom within an aggregate.

Prolongation smoothing then produces  $P$  by improving  $T$  with the goal of making  $\text{span}(P)$  more accurate for low energy modes. As a result, the sparsity pattern of  $P$  is typically expanded from the block diagonal form in (2). Classical SA uses one iteration of weighted-Jacobi, i.e.,  $P = (I - \omega D^{-1} A) T$ . Here, a more sophisticated energy-minimization procedure is used to smooth  $T$ , and is outlined in Section 2.1. Finally, Galerkin coarsening is used to construct the operator for the next coarse-level.

---

**Algorithm 1:** `sa_setup`( $A_0, B_0$ )

---

```

1 for  $\ell = 0$  to  $\ell_{max}$  do
2   smooth  $\eta$  times on  $A_\ell B_\ell = \mathbf{0}$ 
3    $S_\ell = \text{strength}(A_\ell)$ 
4    $\mathcal{N}_\ell = \text{aggregate}(S_\ell)$ 
5    $T_\ell, B_{\ell+1} = \text{inject\_modes}(\mathcal{N}_\ell, B_\ell)$ 
6    $P_\ell = \text{energy\_min}(A_\ell, T_\ell)$ 
7    $A_{\ell+1} = P_\ell^* A_\ell P_\ell$ 
8 return  $A_0, \dots, A_\ell, P_0, \dots, P_{\ell-1}, R_0, \dots, R_{\ell-1}$ 

```

---

## 2.1 F/C-Based Energy-Minimization Prolongation Smoothing

This section outlines the energy-minimizing prolongation smoother used here. To begin, the root-node perspective is presented which induces the  $F/C$ -style  $P$ . This perspective views each aggregate as being centered around a node, and then constructs a  $P$  that injects

from a coarse-level node to this fine-level root-node. The root-nodes and non-root-nodes are analogous to the  $C$ -points and  $F$ -points, respectively, from classical AMG. Such a  $P$  is easily constructed. First, take  $T$  and scale each column so that the entry corresponding to each root-node row is one. Second, to preserve the exact interpolation,  $T_\ell B_{\ell+1} = B_\ell$ ,  $B_{\ell+1}$  must be scaled accordingly to equal injection of  $B_\ell$ . Finally, the smoothing of  $T$  only occurs for non-root-node rows, (i.e.  $F$ -rows), thus yielding a  $P$  that can be row-permuted to the  $F/C$ -style sparsity pattern (3), where  $I$  denotes coarse-grid injection.

$$P = \begin{bmatrix} P_F \\ I \end{bmatrix}. \quad (3)$$

This approach has some attractive properties. The identity block in (3) guarantees that the columns of  $P$  are linearly independent, which is important during the energy-minimization process. Also, the ideal  $P$  (c.f. [9]) for (3) is given by  $P_{ideal} = [-A_{FF}^{-1}A_{FC}^T, I]^T$ , where  $A_{FF}$  corresponds to  $F$ -rows and  $F$ -columns of  $A$ , and  $A_{FC}$  corresponds to  $F$ -rows, but  $C$ -columns. Intuitively,  $P_{ideal}$  has “zero” energy at  $F$ -points because  $AP_{ideal}$  is zero at all  $F$ -rows. It is through this fact that the “quality” of  $P$  is later determined.

A brief description of the energy-minimizing prolongation smoother as used here, is now given. For more details, see [13]. Since  $A$  is SPD, projected CG is used to compute

$$\operatorname{argmin}_{P \in V} \sum_j \|P_{(j)}\|_A^2, \quad (4)$$

where  $P_{(j)}$  is the  $j^{th}$  column and  $V$  is a Krylov sub-space over which two constraints are enforced:  $P_\ell B_{\ell+1} = B_\ell$  and a sparsity pattern constraint.  $\tilde{V}$  has the form

$$\{Q(I \otimes A)\tilde{T}, (Q(I \otimes A))^2\tilde{T}, \dots\}, \quad (5)$$

where the  $\tilde{\cdot}$  operator denotes column-wise conversion of a matrix to a column-vector, e.g.,  $\tilde{T} = [T_{(1)}^T \ T_{(2)}^T \ T_{(3)}^T \ \dots]^T$  and  $T_{(j)}$  denotes the  $j^{th}$  column of  $T$ . The matrix,  $I$ , is the identity with size equal to the coarse-grid size and the projection operator,  $Q$ , enforces the constraints. The near null-space preservation constraint,  $P_\ell B_{\ell+1} = B_\ell$ , is enforced with an  $\ell_2$ -projection and critically guides the minimization process.

The projection,  $Q$ , enforces the sparsity pattern constraint by zeroing out those entries not present in the graph of

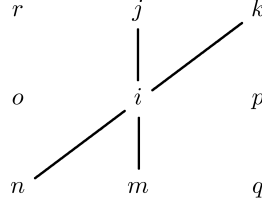
$$\begin{bmatrix} (|S|^d |T|)_F \\ I \end{bmatrix}, \quad (6)$$

where  $d \in \{1, 2, \dots\}$ ,  $|\cdot|$  denotes an element-wise absolute value, and  $(\cdot)_F$  represents restriction to  $F$ -rows. By increasing  $d$ , the sparsity pattern grows in the direction of strong connections and coarse-grid injection is preserved. This flexibility for larger sparsity patterns is crucial to the success of the proposed approach. Typically,  $2d$  iterations of the energy-minimization process are taken, with the complexity of the whole process on the order of classical prolongation smoothing [8],  $P = p_d(A)T$ , where  $p_d(A)$  is a  $d^{th}$  degree polynomial in  $A$  constructed to reduce energy, e.g., Chebyshev.

### 3 Suitable Interpolation for the Non-Grid-Aligned Case

In this section, an improved strategy for constructing  $P$  in the non-grid-aligned case is motivated. The other possible means of improving SA, better coarsening or relaxation, are not explored, for the reasons discussed in Section 1.

The key point of long-distance interpolation stencils is now motivated. Long-distance stencils are defined as expanding the sparsity pattern for  $P$  with  $d > 1$ , but typically,  $d \in \{2, 3, 4\}$ . Consider the generic strength stencil in Figure 1, where the model problem from Section 1 with  $\theta = 3\pi/16$  is again considered. Solid lines represent a strong connection. Degree-of-freedom  $i$  is not strongly connected to  $o$  or  $p$ , but it has strongly connected

Figure 1: Generic Strength-of-Connection Stencil for  $\theta = 3\pi/16$ .

neighbors  $n$  and  $k$  which are in turn strongly connected to  $o$  and  $p$ , respectively. Additionally,  $n$  and  $m$  are not strongly connected, and  $j$  and  $k$  are not strongly connected, but they are in the same aggregate whenever  $i$  is a root-node. These two situations *never* arise for grid-aligned anisotropies, or for isotropic diffusion, thus raising the following questions. Should the interpolation stencil be expanded in weak directions, i.e., in the direction of  $o$  and  $p$ , and should the stencil be of a non-standard character, i.e., not be constant-like in all directions? Both answers are yes when energy-minimization is properly employed.

The need for long-distance interpolation is now quantified for the model problems from Section 1. Specifically, the quality of  $P$  is examined through the residual of the equation that follows from the definition of  $P_{ideal}$ . Namely, that  $(AP)_F = 0$ , which implies that

$$\begin{bmatrix} A_{FF} & A_{FC} \\ A_{CF} & A_{CC} \end{bmatrix} \begin{bmatrix} P_F \\ I \end{bmatrix} = \begin{bmatrix} A_{FF}P_F + A_{FC} \\ A_{CF}P_F + A_{CC} \end{bmatrix}, \quad (7)$$

is 0 at all  $F$ -rows, if  $P = P_{ideal}$ . To measure the closeness of a given  $P$  to  $P_{ideal}$ , the residual term  $R = (AP)_F$  is formed and then the mean value of column-wise maximums is computed over all columns  $R_{(j)}$  of  $R$  with

$$\text{mean}(\{\max(R_{(0)}), \max(R_{(1)}), \dots\}). \quad (8)$$

This measure is useful in the context of the chosen model problems on regular grids, because the mean reflects the typical residual value for a column away from the boundary. Evaluating the closeness of  $P$  to  $P_{ideal}$  in this way is a specific contribution of this paper and has clear connections to evaluating the accuracy of sparse approximate inverses (SPAI) [10].

Table 2 shows these mean residual values for the first three levels in the SA hierarchy over various values of  $d$ . As  $d$  increases for the non-grid-aligned cases, the mean residual values decrease until they reach the goal of being roughly on the same order as those for the grid-aligned case of  $\theta = 0$ , which is known to produce a scalable solver. Once this occurs, a scalable SA solver is achieved, which is denoted by the gray shading. *Scalability depends on the level of approximation of  $P_{ideal}$  by  $P$ , regardless of  $\theta$ .* If standard weighted-Jacobi is used, the results are similar to  $d = 1$ . Since  $d = 4$  is the most robust value and is effective for each  $\theta$ , it becomes the standard choice in the results section.

Critical to success is adequate strength-of-connection and aggregation, which guide the expansion of the sparsity pattern according to (6). Here, the strength measure [14] is used with an appropriate drop-tolerance to produce the aggregation patterns of semi-coarsening for the (nearly) grid-aligned cases, (e.g.,  $\theta = 0, \pi/16$  in Table 2), and of larger aggregates in non-grid-aligned cases. Figure 2 depicts typical aggregates for  $\theta = 3\pi/16$  as the shaded regions containing the square nodal locations. Essentially, the “smearing” in the aggregate shapes reflects the “smearing” in the matrix stencil. Using such aggregates are critical, because they allow for the sparsity pattern of  $P$  to spread in both weak and strong directions, with the spread in weak directions happening more slowly, (as shown visually in Figure 3).

To further the motivation, columns of  $P$  are visually compared for various  $\theta$  and different prolongation settings. The column of  $P$  that corresponds to the aggregate with root-node located in the center of the domain is plotted. The four different  $P$  examined are  $P_{ideal}$ , the energy-minimizing prolongation smoother for both  $d = 1$  and 4, and the energy-minimizing

$\theta$	0	$\pi/16$	$2\pi/16$	$3\pi/16$	$4\pi/16$
$d = 1$	1.7e-3	5.4e-2	3.0e-1	2.4e-1	1.2e-1
	3.7e-3	1.3e-1	2.4e-1	1.5e-1	2.0e-1
	1.0e-2	1.4e-1	1.3e-1	4.6e-1	5.4e-1
$d = 2$	1.7e-3	5.4e-2	5.9e-2	4.4e-2	3.6e-2
	3.7e-3	2.0e-2	2.7e-2	1.0e-1	8.6e-2
	1.0e-2	5.2e-2	9.6e-2	9.3e-2	1.1e-1
$d = 3$	1.5e-3	5.1e-2	2.1e-2	2.1e-2	1.3e-2
	3.3e-3	8.6e-3	9.7e-3	7.9e-2	6.5e-2
	9.7e-3	4.6e-2	6.0e-2	4.9e-2	8.0e-2
$d = 4$	9.0e-4	4.7e-2	1.2e-2	1.6e-2	5.3e-3
	1.9e-3	6.0e-3	4.4e-3	4.6e-2	3.5e-2
	4.9e-3	2.7e-2	4.1e-2	1.7e-2	2.0e-2

Table 2: Mean Residual Terms Based on  $R = (AP)_F$ .Figure 2: Typical Aggregates for  $\theta = 3\pi/16$ .

prolongation smoother *without* coarse-grid injection for  $d = 4$ , i.e., the identity block in (6) is replaced by the sparsity pattern defined by the  $C$ -rows  $(|S|^d|T|)_C$ .

Figure 3a depicts how the columns of  $P_{ideal}$  become less local for the anisotropic case, lending support to expanding the sparsity pattern of  $P$ . Figure 3b depicts how  $d = 4$  produces a  $P$  that, at least visually, has the character of  $P_{ideal}$ . Figure 3c depicts  $d = 1$ , where the nonzeros of  $P$  are similar to the corresponding locations for  $d = 4$ , thus indicating that the expanded sparsity pattern of  $d = 4$  is a key reason for its superior performance. Figure 3d depicts  $d = 4$  *without* coarse-grid injection. Here, the resulting columns of  $P$  no longer resemble the corresponding columns of  $P_{ideal}$ . Experiments also confirm that not enforcing coarse-grid injection results in deteriorated SA performance.

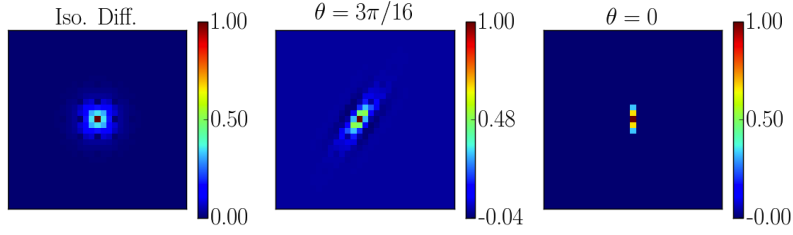
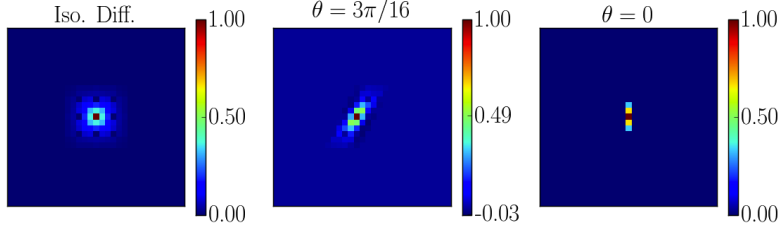
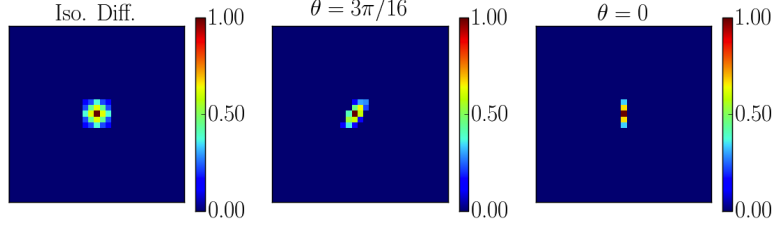
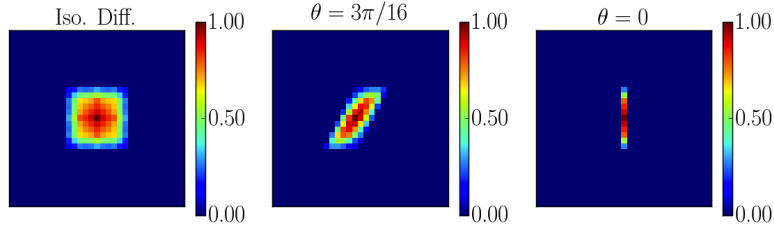
## 4 Results

Numerical evidence is now provided that the proposed strategy yields a robust solver. V(1,1)-cycles with symmetric Gauss-Seidel and a maximum coarse-grid size of 500 are used to accelerate CG.  $P$  is computed with  $d = 4$ . The near null-space mode is  $B = \mathbf{1}$ . All experiments are carried out in the software package PyAMG described in [1].

The operator complexity is reported to reassure the reader that the large coarse-grid stencils induced by  $d = 4$  do not create an unusable method. For the problems considered here, the complexity is tolerably high, i.e., 2–2.5. In part, the large coarse-grid stencils lead to faster coarsening, thus partially alleviating the adverse effect on complexity.

Table 3 depicts the results for the model problems from Section 1 and mimics the data layout of Table 1. The results indicate that  $\gamma$  and work are (nearly) scalable for all  $\theta$ .

Table 4 shows results for 2D isotropic diffusion discretized with simplices on a stretched mesh. The mesh is an unstructured triangulation of the unit circle that later has the  $y$ -coordinates stretched by a factor of 100, thus yielding non-grid-aligned anisotropic behavior. The matrix dimensions for each grid size are 13 373, 53 069, 217 143, and 869 261. Table entries are presented as  $a(b)$ , representing quantities when using energy-minimization and

(a)  $P_{ideal}$ .(b)  $d = 4$ .(c)  $d = 1$ .(d)  $d = 4$  without Coarse Grid Injection.Figure 3: Normalized Columns of  $P$  for Aggregate Centered in Domain.

$\theta$	0	$\pi/16$	$2\pi/16$	$3\pi/16$	$4\pi/16$
$\gamma$	0.05, 0.05	0.17, 0.18	0.15, 0.17	0.10, 0.14	0.18, 0.18
Op. Comp.	2.1, 2.2	2.3, 2.4	2.1, 2.2	2.3, 2.5	2.0, 2.1
Work	6, 7	11, 13	10, 12	9, 11	11, 11

Table 3: Model Rotated Anisotropic Diffusion,

Entries  $a, b$  Refer to Experiments with a  $127 \times 127$  and  $1023 \times 1023$  Grid, Respectively.

standard weighted-Jacobi, respectively. Scalability is observed, while standard SA struggles.

Table 5 depicts the results for 1D radiation transport mapped to spherical coordinates, which yields the 2D problem

$$\mathbf{b} \cdot \nabla u + \sigma u = f, \quad (9)$$

grid #	1	2	3	4
$\gamma$	0.27 (0.55)	0.30 (0.64)	0.35 (0.71)	0.34 (0.78)
Work	16 (22)	19 (29)	22 (39)	21 (53)
Op. Comp.	2.4 (1.4)	2.5 (1.5)	2.5 (1.5)	2.5 (1.4)

Table 4: Isotropic Diffusion on Stretched Unit Circle,  
Entries  $a(b)$  Refer to Experiments with the Proposed  $P$  and Weighted-Jacobi, Respectively.

where  $\mathbf{b}(x, y) = [xy, 1 - y^2]$  and  $1/\sigma$  represents the mean free path between collisions. Here, the domain is  $[0, 1] \times [-1, 1]$ , and  $\sigma = 10^{-6}$ , indicating almost no collisions. The discretizations examined are Q1 and Q2 finite elements (FE) with FOSLS (first-order system least squares) [17]. FOSLS yields a matrix that has the dominant differential term of rotated anisotropic diffusion defined by  $-\nabla \cdot [\mathbf{b}^T \mathbf{b}] \nabla$ , in addition to some lower order terms. Four grid sizes are examined,  $64 \times 128$ ,  $128 \times 256$ ,  $256 \times 512$ , and  $512 \times 1024$ . A nearly scalable solver is observed for Q1 FE, while again standard SA struggles. If W-cycles are used instead, the solver is scalable in iterations for Q1 FE. The case of Q2 FE is a current topic of research.

FE Type	Q1			Q2		
grid #	2	3	4	1	2	3
$\gamma$	0.36 (0.64)	0.37 (0.68)	0.41 (0.75)	0.57 (0.74)	0.62 (0.82)	0.69 (0.86)
Work	21 (31)	22 (37)	25 (48)	28 (41)	34 (63)	43 (83)
Op. Comp.	2.5 (1.5)	2.5 (1.5)	2.4 (1.5)	1.8 (1.3)	1.8 (1.3)	1.8 (1.3)

Table 5: Radiation Transport,  
Entries  $a(b)$  Refer to Experiments with the Proposed  $P$  and Weighted-Jacobi, Respectively.

Figure 4 shows how aggregation follows the direction of the anisotropy as it rotates according to  $\mathbf{b}$ . The lower half of the domain is shown, but the upper half is a mirror image. The aggregates are depicted as geometric shapes (lines or polygons), while the nodal locations are small squares. The blow-up of the aggregate shows a 3D plot of that aggregate's corresponding column in  $P$ , which curves according to the direction of the anisotropy.

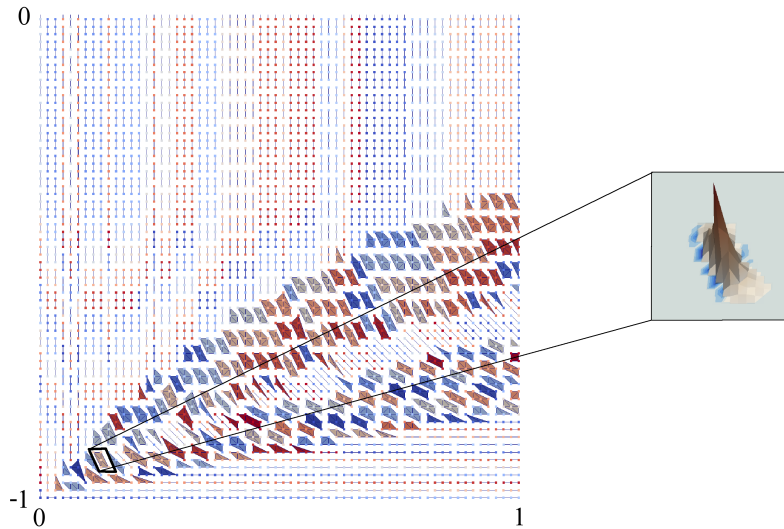


Figure 4: Aggregates for Transport Problem, with Blow-Up of Selected Column of  $P$ .



## 5 Future Work

A potential drawback of this approach is that when using coarse-grid injection, the number of functions fitted over each aggregate during the construction of  $T$ , (see (2)), must equal the supernode size <sup>1</sup>. For 3D elasticity and diffusion, supernodes are typically of size three and one respectively, thus allowing for three and one functions, respectively.

This restriction is at odds with classical SA, which has no restriction on the number of functions fitted over an aggregate. For instance, with 3D elasticity, the six rigid-body-modes are typically fit over each aggregate, i.e., three displacements and three rotations. However, the proposed approach automatically compensates for this restriction. To show this, an elasticity experiment is run with  $B$  equal to the three displacements. Then,  $e$  is computed, which measures how well  $\text{span}(P)$  approximates the rotational rigid-body-modes. Let

$$e = \sum_{i=1,2,3} \frac{\|(I - P(P^T P)^{-1} P^T) \hat{B}_{(i)}\|}{\|\hat{B}_{(i)}\|_A}, \quad (10)$$

where  $\hat{B}_{(i)}$  is  $i^{\text{th}}$  rotation and  $P$  is taken from the finest-level. Thus,  $e$  represents how well  $\text{span}(P)$  approximates  $\hat{B}$  in terms of the weak approximation property [2]. The test problem is isotropic linearized elasticity for a tripod, with a downward force applied to it, discretized with tetrahedra and linears.  $P$  has dimensions  $16341 \times 1347$  and the experiment is run with various  $d$  and, for comparison, weighted-Jacobi. Table 6 depicts these results.

ProL. Smoother:	w-Jacobi	$d = 1$	$d = 2$	$d = 3$	$d = 4$	$d = 5$
$\gamma$	0.62	0.45	0.40	0.30	0.17	0.16
$e$	1.1e-1	6.4e-2	4.0e-2	2.7e-2	2.0e-2	1.7e-2
Op. Comp.	1.2	1.2	1.4	1.6	1.8	2.0

Table 6: Residual Reduction Factors and Error in  $\text{span}(P)$  for Rotations.

As  $d$  increases, both  $\gamma$  and  $e$  improve, but at the cost of a higher operator complexity. For comparison, use of all six rigid-body-modes in  $B$  yields a solver with  $\gamma = 0.15$  and an operator complexity of 1.8. Additionally, there is more numerical evidence that larger  $d$  yields nearly scalable results for this elasticity problem. However, more experiments are clearly needed to confirm this result.

Moreover, an extension of the energy-minimizing prolongation smoother is planned so that the rotations are exactly incorporated into  $\text{span}(P)$  through a modification of the constraints. So long as there are enough nonzeros in each row of  $P$ , this goal is achievable. Also, using constraints to incorporate more modes into  $\text{span}(P)$  than there are functions fit over an aggregate is reminiscent of recent adaptive BAMG methods [4].

Last, this paper is a first step towards optimizing the sparsity pattern for  $P$ . An extension is planned that initially generates a  $P$  with  $d = 1$ , and wherever  $\max(R_j)$  is large, the sparsity pattern of column  $j$  is extended according to the strength matrix,  $S$ . Some energy-minimization iterations are then run to fill in the newly allowed nonzeros. This process repeats until each  $\max(R_j)$  falls below a threshold, or a maximum number of nonzeros is reached. Such an approach allows real-world problems, such as radiation transport, to be more efficiently handled by only allowing nonzeros where necessary. Wherever the anisotropy is (nearly) grid-aligned,  $d = 1$  is used, and elsewhere,  $d$  grows to 2, 3 or 4, depending on the non-grid-aligned anisotropy in that region of the domain. A preliminary implementation of this variable  $d$  strategy shows a savings of 3-5 work units over Table 5 for the transport problem.

<sup>1</sup>A common SA approach when multiple variables exist at each spatial location for the original matrix problem is supernodes [18]. Supernodes cause aggregation to happen in two stages at the finest-level. First, supernodes are formed by aggregating degrees-of-freedom at the same spatial location together. Second, strength-of-connection information is used to aggregate the supernodes.

## 6 Conclusions

A novel interpolation strategy of coarse-grid injection coupled with energy-minimization and long-distance stencils is developed. For the non-grid-aligned anisotropic diffusion problems examined here with linear FE, the resulting solvers appear to be scalable and are *the first such solvers known to the author*. Moreover, this indicates that neither new coarsening nor new relaxation strategies are necessary.

The sparsity pattern choice for  $P$  is guided by a proposed measure that tracks closeness to  $P_{ideal}$  and that predicts scalability. Overall, this strategy for choosing the sparsity pattern is a first step towards a general approach for optimizing interpolation sparsity patterns.

## References

- [1] W. N. Bell, L. N. Olson, and J. Schroder. PyAMG: Algebraic multigrid solvers in Python v1.0. <http://www.pyamg.org>, 2008. Release 1.0.
- [2] J. H. Bramble, J. E. Pasciak, J. Wang, and J. Xu. Convergence estimates for multigrid algorithms without regularity assumptions. *Math. of Comp.*, 57(195):23–45, 1991.
- [3] A. Brandt. General highly accurate algebraic coarsening. *Electron. Trans. Numer. Anal.*, 10:1–20, 2000.
- [4] A. Brandt. Multiscale scientific computation. In *Barth, T.J., Chan, T.F. and Haimes, R. (eds.): Multiscale and Multiresolution Methods: Theory and Application*, pages 1–96, 2001.
- [5] A. Brandt, S. F. McCormick, and J. W. Ruge. Algebraic multigrid (AMG) for sparse matrix equations. In D. J. Evans, editor, *Sparsity and Its Applications*, pages 257–284. Cambridge Univ. Press, Cambridge, 1984.
- [6] J. Brannick, M. Brezina, S. MacLachlan, T. Manteuffel, S. McCormick, and J. Ruge. An energy-based AMG coarsening strategy. *Numer. Linear Algebra Appl.*, 13(2-3):133–148, 2006.
- [7] J. Brannick and L. Zikatanov. Algebraic multigrid methods based on compatible relaxation and energy minimization. In O. Widlund and D. E. Keyes, editors, *Domain Decomposition Methods in Science and Engineering XVI*, volume 55 of *Lecture Notes in Computational Science and Engineering*, pages 15–26. Springer, Berlin, 2007.
- [8] M. Brezina and P. Vaněk. A black-box iterative solver based on a two-level Schwarz method. *Computing*, 63:233–263, 1999.
- [9] R. D. Falgout and P. S. Vassilevski. On generalizing the algebraic multigrid framework. *SIAM J. Numer. Anal.*, 42(4):1669–1693, 2004.
- [10] M. Grote and T. Huckle. Sparse approximate inverses. *SIAM J. Sci. Comput.*, 18:838–853, 1997.
- [11] J. E. Jones and P. S. Vassilevski. Amge based on element agglomeration. *SIAM J. on Sci. Comput.*, 23(1):109–133, 2001.
- [12] D. J. Mavriplis. Multigrid strategies for viscous flow solvers on anisotropic unstructured meshes. *J. of Comp. Phys.*, 145(1):141 – 165, 1998.
- [13] L. N. Olson, J. B. Schroder, and R. S. Tuminaro. A general interpolation strategy for algebraic multigrid using energy-minimization. *SIAM J. Sci. Comput.*, 2010. Submitted.
- [14] L. N. Olson, J. B. Schroder, and R. S. Tuminaro. A new perspective on strength measures in algebraic multigrid. *Numer. Linear Algebra Appl.*, 17(4):713–733, 2010.
- [15] C. W. Oosterlee. A GMRES-based plane smoother in multigrid to solve 3d anisotropic fluid flow problems. *J. of Comp. Phys.*, 130(1):41 – 53, 1997.
- [16] J. W. Ruge and K. Stüben. Algebraic multigrid (AMG). In S. F. McCormick, editor, *Multigrid Methods*, Frontiers Appl. Math., pages 73–130. SIAM, Philadelphia, 1987.
- [17] H. D. Sterck, T. A. Manteuffel, S. F. McCormick, and L. Olson. Least-squares finite element methods and algebraic multigrid solvers for linear hyperbolic PDEs. *SIAM J. on Sci. Comp.*, 26(1):31–54, 2004.
- [18] P. Vaněk, M. Brezina, and J. Mandel. Convergence of algebraic multigrid based on smoothed aggregation. *Numerische Mathematik*, 88(3):559–579, 2001.
- [19] P. Vaněk, J. Mandel, and M. Brezina. Algebraic multigrid based on smoothed aggregation for second and fourth order problems. *Computing*, 56:179–196, 1996.