Independent Quality Measures for Symmetric Algebraic Multigrid Components

James Lottes[a]

Oxford Centre for Collaborative Applied Mathematics,
Mathematical Institute, 24–29 St Giles' Oxford, OX1 3LB, England
[a]Mathematics and Computer Science Division, Argonne National
Laboratory, Argonne, IL 60439, U.S.A.

Abstract

A new algebraic multigrid (AMG) method is developed to replace a fast, parallel direct solver used for the coarse-grid problem in a massively parrallel ($P \geq 10^5$) implementation of a multilevel method, resulting in a dramatic improvement in overall efficiency. In addition to being sparse and symmetric positive definite (SPD), these coarse-grid problems are characterized by having few degrees of freedom per processor, $n/P = O(1)$. For the target processor counts, the coarse problem is large and a challenge to solve efficiently. The AMG method developed aims to produce the most efficient AMG hierarchy possible, without regard to setup costs, because the target applications require the approximate solution of a single, unchanging system hundreds of thousands of times within a single computation. The thrust of the approach rests on proposed measures of quality for each AMG component: coarsening, interpolation, and smoothing. Heuristic procedures are developed for constructing near-optimum components in turn, targeting approximations of the proposed theoretical quality measures. Crucially, the measures do not reference those components yet to be constructed. For example, the proposed measure of coarsening quality is independent of both the interpolation and the smoother. Moreover, these measures are grounded in theory; in particular a two-grid convergence bound in terms of them is proven. Numerical results comparing optimized AMG with a fast parallel direct solver intended for coarse problems show efficiency gains up to nearly two orders of magnitude. While coarse-grid problems motivated this research, the theory presented applies generally and provides a framework for deriving AMG strategies for general SPD systems.

1 Introduction

We consider the development of algebraic multigrid (AMG) methods for highly parallel solution of sparse symmetric positive-definite (SPD) systems.

Our focus is on the development of independent quality metrics that allow us to optimize the parallel solution process, without regard to setup costs. Our intended application is solution of coarse-grid systems that arise in distributed-memory implementations of multilevel Schwarz and multigrid solvers, where one is faced with repeated (perhaps approximate) solutions of systems that have distributed data and solutions and that have relatively few (tens to hundreds) degrees of freedom per processor. Such systems have been considered in the past (e.g., [4, 10]), and a review of the state of the art as of 2001 is given in [13]. The present work differs from earlier efforts in that it targets processor counts exceeding $P = 10^5$ and thus requires $O(n)$ or at most $O(n \log n)$ complexity, given that $n \geq P$. While the coarse-grid-solve problem provided the motivation for the current work, the results presented here are more general and, we believe, provide a rational framework for deriving AMG convergence strategies in the context of SPD systems.

We view an AMG iteration as consisting of the three heuristically chosen components: coarsening, interpolation weights, and smoother. The efficiency of AMG depends on the cost per iteration and the convergence rate. The components contribute more or less independently to the cost per iteration, but they seem to interact in a complex way in determining the convergence rate. We propose a technique for quantifying the “quality” of each component in a way that has no “forward” dependencies; for example, we quantify the coarsening quality without reference to interpolation or smoother. In Section 3, where we present these quality measures, we discuss how making each quantity small (smaller being better) together implies an efficient algorithm. In particular, we prove a convergence bound involving these quantities.

Equipped with this theory, we constructed heuristic procedures for each component, in each case targeting some computable approximation to our theoretical quality. Our coarsening procedure in particular might be of interest—it is a simple, inherently parallel procedure using the concept of Gershgorin discs and requiring no special treatment of positive off-diagonals. We forgo the “strength of connection” heuristic and the corresponding strong connection threshold parameter.

We discuss connections with other literature as they arise, but we provide a brief summary here. Our notion of coarsening quality, which doesn’t involve the smoother/relaxation, nonetheless strongly resembles Brandt’s idea of “compatible relaxation” [2]. It draws heavily from ideas of Brannick and Zikatanov [3], going back to a result by Demko, Moss, and Smith [6]. Our interpolation quality appears in the analysis of Falgout, Vassilevski, and Zikatanov [8] but not in the context of general smoothers. Our convergence bound theorem is closely related to a theorem of the same paper.

2 Convergence Theory

We begin by presenting a general AMG convergence theory for SPD systems, which provides the foundation for the independent quality measures proposed in the next section.

2.1 Notation

We are interested in AMG as an iterative method for solving the linear system

$$A\mathbf{x} = \mathbf{b},$$

with A a large, sparse, symmetric positive-definite (SPD) $n \times n$ matrix. We define the two-level multigrid, or "two-grid," iteration

$$\mathbf{x} - \mathbf{x}_{i+1} = E_{\mathrm{tg}}(\mathbf{x} - \mathbf{x}_i) \tag{1}$$

in terms of the two-grid error-propagation matrix

$$E_{\mathrm{tg}} := (I - BA)(I - PA_c^{-1}P^T A), \tag{2}$$

where the coarse operator $A_c := P^T AP$. The iteration is determined by the matrix B, defining the *smoother*, and by the $n \times n_c$ *prolongation* or *interpolation* matrix P. We are concerned only with the asymptotic convergence rate of (1), equal to $\rho(E_{\mathrm{tg}})$, the spectral radius of E_{tg}. As a result, our analysis also covers iterations with multiple pre- and postsmoothing steps, such as that corresponding to

$$(I - M^{-T}A)^{m_1}(I - PA_c^{-1}P^T A)(I - M^{-1}A)^{m_2},$$

which may be cyclically permuted into a form matching (2), without affecting eigenvalues. We prefer the form (2) because it is general: invertibility of B, symmetry of the "V-cycle," and iteration by simple repetition are all unnecessary constraints on the algorithm.

We restrict our attention to classical AMG, in which the coarse "C-variables" are a subset of the original variables. Selecting this subset is called *coarsening*. If we order the C-variables last, then A and P take on the block forms

$$A =: \begin{bmatrix} A_{ff} & A_{fc} \\ A_{cf} & A_{cc} \end{bmatrix}, \quad P =: \begin{bmatrix} W \\ I \end{bmatrix},$$

where we have introduced the interpolation *weights* W, which specify how P interpolates the first $n_f = n - n_c$ "F-variables" from the C-variables. Note that, following Falgout and Vassilevski [7], other AMG methods can often be cast in this classic form with the help of a change-of-basis matrix.

2.2 Hierarchical Decomposition

We introduce the two-level hierarchical basis as the columns of T and consider A transformed to this basis,

$$T := \begin{bmatrix} I & W \\ & I \end{bmatrix}, \quad \hat{A} := T^T AT = \begin{bmatrix} A_{ff} & \hat{A}_{fc} \\ \hat{A}_{fc}^T & A_c \end{bmatrix},$$

where

$$\hat{A}_{fc} := A_{ff}W + A_{fc}.$$

Note that the particular choice of weights $W = -A_{ff}^{-1}A_{fc}$, which is generally not sparse, gives $\hat{A}_{fc} = O$ and renders the decomposition orthogonal.

Because of this well-known property these weights are often termed the *ideal* weights. Of course, this orthogonal decomposition must involve the Schur complement of A_{ff},

$$S_c := A_{cc} - A_{cf} A_{ff}^{-1} A_{fc}.$$

In fact, one can easily check that

$$A_c := P^T A P = S_c + \hat{A}_{fc}^T A_{ff}^{-1} \hat{A}_{fc},$$

so that A_c reduces to the Schur complement when $\hat{A}_{fc} = O$, as expected. Moreover, from this last equation it follows immediately that for any $\mathbf{x}_c \in \mathbb{R}^{n_c}$,

$$\|P\mathbf{x}_c\|_A^2 = \|\mathbf{x}_c\|_{A_c}^2 = \|\mathbf{x}_c\|_{S_c}^2 + \|A_{ff}^{-1} \hat{A}_{fc} \mathbf{x}_c\|_{A_{ff}}^2. \tag{3}$$

The ideal weights thus minimize the energy norm of any prolongated vector (only the second term depends on W, and it vanishes when $\hat{A}_{fc} = O$).

We may also transform the other matrices we have defined to the hierarchical basis.

$$\hat{P} := T^{-1} P = \begin{bmatrix} O \\ I \end{bmatrix}, \quad \hat{B} := T^{-1} B T^{-T} =: \begin{bmatrix} \hat{B}_{ff} & \hat{B}_{fc} \\ \hat{B}_{cf} & \hat{B}_{cc} \end{bmatrix}$$

$$\hat{E}_{\text{tg}} := (I - \hat{B}\hat{A})(I - \hat{P} A_c^{-1} \hat{P}^T \hat{A}) = T^{-1} E_{\text{tg}} T$$

Note that $\hat{E}_{\text{tg}}$ and E_{tg} are similar matrices and thus have identical spectra.

2.3 Orthogonal Decomposition

The hierarchical decomposition of the previous section is not orthogonal except in the special case of the generally nonsparse ideal weights $W = -A_{ff}^{-1} A_{fc}$. Here we consider leaving the nonideal coarse basis alone and instead orthogonalizing the F-variables against the coarse subspace. We write this new basis, in hierarchical coordinates, as the columns of Q, and we consider $\hat{A}$ transformed to this basis,

$$Q := \begin{bmatrix} I & \\ -A_c^{-1} \hat{A}_{fc}^T & I \end{bmatrix}, \quad \tilde{A} := Q^T \hat{A} Q = \begin{bmatrix} S_f & \\ & A_c \end{bmatrix},$$

where

$$S_f := A_{ff} - \hat{A}_{fc} A_c^{-1} \hat{A}_{fc}^T$$

is the Schur complement of the coarse space operator A_c. Note that this construction mirrors that of the previous section for the ideal weights case. As in the previous section, we may transform the other matrices also.

$$\tilde{P} := Q^{-1} \hat{P} = \begin{bmatrix} O \\ I \end{bmatrix}, \quad \tilde{B} := Q^{-1} \hat{B} Q^{-T} =: \begin{bmatrix} \tilde{B}_{ff} & \tilde{B}_{fc} \\ \tilde{B}_{cf} & \tilde{B}_{cc} \end{bmatrix}$$

$$\tilde{E}_{\text{tg}} := (I - \tilde{B}\tilde{A})(I - \tilde{P} A_c^{-1} \tilde{P}^T \tilde{A}) = Q^{-1} \hat{E}_{\text{tg}} Q$$

Again, we see that E_{tg}, $\hat{E}_{\text{tg}}$, and $\tilde{E}_{\text{tg}}$ are all similar and have identical spectra. In particular, the two-grid asymptotic convergence rate is $\rho(\tilde{E}_{\text{tg}}) = \rho(\hat{E}_{\text{tg}}) = \rho(E_{\text{tg}})$.

2.4 Effective Smoother Component

We now introduce our first convergence theorem, in which we isolate the component of the smoother that affects the convergence. Since we have introduced the transformations to hierarchical and orthogonal decompositions, the proof is trivial. First,

$$I - \tilde{P} A_c^{-1} \tilde{P}^T \tilde{A} = I - \begin{bmatrix} O & \\ & A_c^{-1} \end{bmatrix} \begin{bmatrix} S_f & \\ & A_c \end{bmatrix} = \begin{bmatrix} I & \\ & O \end{bmatrix}.$$

That is, the coarse-grid correction zeros the coarse component of the error in the orthogonal decomposition. Next,

$$\begin{aligned} \tilde{E}_{\text{tg}} &= (I - \tilde{B}\tilde{A}) \begin{bmatrix} I & \\ & O \end{bmatrix} \\ &= \begin{bmatrix} I & \\ & O \end{bmatrix} - \begin{bmatrix} \tilde{B}_{ff} & \tilde{B}_{fc} \\ \tilde{B}_{cf} & \tilde{B}_{cc} \end{bmatrix} \begin{bmatrix} S_f & \\ & O \end{bmatrix} \\ &= \begin{bmatrix} I - \tilde{B}_{ff} S_f & O \\ -\tilde{B}_{cf} S_f & O \end{bmatrix}. \end{aligned} \tag{4}$$

Evidently, the block $\tilde{B}_{ff}$ is important. Let us relate it back to the original smoother matrix B. First, observe that

$$T^{-1} = \begin{bmatrix} I & -W \\ & I \end{bmatrix}, \quad Q^{-1} = \begin{bmatrix} I & \\ A_c^{-1} \hat{A}_{fc}^T & I \end{bmatrix}.$$

Since $\hat{B} := T^{-1} B T^{-T}$ and $\tilde{B} := Q^{-1} \hat{B} Q^{-T}$, it follows that

$$\tilde{B}_{ff} = \hat{B}_{ff} = \begin{bmatrix} I & -W \end{bmatrix} B \begin{bmatrix} I & -W \end{bmatrix}^T.$$

Note that $\begin{bmatrix} I & -W \end{bmatrix}$ is simply the projection onto the F-variables associated with the hierarchical decomposition. We have just proved the following theorem, which says that only the $\hat{B}_{ff}$ component of the smoother B affects the asymptotic convergence.

Theorem 1. *Of the n eigenvalues of E_{tg}, n_c are zero; the remaining n_f are eigenvalues of $I - \hat{B}_{ff} S_f$. That is,*

$$\lambda(E_{tg}) = \lambda(I - \hat{B}_{ff} S_f) \cup \{0^{n_c}\}, \tag{5}$$

where

$$\hat{B}_{ff} = \begin{bmatrix} I & -W \end{bmatrix} B \begin{bmatrix} I & -W \end{bmatrix}^T. \tag{6}$$

Proof. The matrices E_{tg} and $\tilde{E}_{\text{tg}}$ are similar, and $\lambda(\tilde{E}_{\text{tg}}) = \lambda(I - \hat{B}_{ff} S_f) \cup \{0^{n_c}\}$ by virtue of (4). This result establishes (5), while (6) has already been established. □

We found this to be a surprising result, for it implies that any smoother may be replaced by the "F-relaxation"

$$B \leftarrow \begin{bmatrix} \hat{B}_{ff} & \\ & O \end{bmatrix}$$

without affecting the asymptotic convergence rate of the two-level iteration, and we note that forming this F-relaxation is fairly trivial; no matrix inverses are involved in (6). We do not suggest doing this in practice, but it does raise questions about what advantages a general smoother has over an F-relaxation.

2.5 Convergence Bound

If the hierarchical decomposition is already orthogonal, because the weights are the ideal $W = -A_{ff}^{-1} A_{fc}$, then the transformation Q reduces to the identity. In particular, in this case we have $S_f = A_{ff}$. In the general case, the hierarchical decomposition should be nearly orthogonal, and S_f will approximately equal A_{ff}. To make this precise, we introduce an energy norm of the difference F of the weights W from the ideal,

$$F := W - (-A_{ff}^{-1} A_{fc}) = A_{ff}^{-1} \hat{A}_{fc}, \tag{7}$$

$$\gamma := \|A_{ff}^{\frac{1}{2}} F A_c^{-\frac{1}{2}}\|_2 = \sup_{\mathbf{v}\neq\mathbf{0}} \frac{\|F\mathbf{v}\|_{A_{ff}}}{\|\mathbf{v}\|_{A_c}}.$$

Using equation (3), we have that

$$\gamma^2 = \sup_{\mathbf{v}\neq\mathbf{0}} \frac{\|\mathbf{v}\|_{A_c}^2 - \|\mathbf{v}\|_{S_c}^2}{\|\mathbf{v}\|_{A_c}^2} < 1.$$

That is, γ characterizes the spectral equivalence between A_c and the Schur complement S_c. Lemma 2.1 of the two-level analysis of Falgout, Vassilevski, and Zikatanov [8] identifies γ as the cosine of the abstract angle between the subspaces of the hierarchical decomposition. We have chosen the notation γ to follow theirs. We are interested in the spectral equivalence between A_{ff}, the other diagonal block of $\hat{A}$, and S_f, the other Schur complement. The fact that their spectral equivalence is characterized by exactly the same constant is a generic property of Schur complements. We demonstrate it in our context in the following lemma.

Lemma 1. *The eigenvalues of $A_{ff}^{-1} S_f$ are real and bounded by*

$$1 - \gamma^2 \leq \lambda(A_{ff}^{-1} S_f) \leq 1.$$

These bounds are tight: $1 - \gamma^2$ is an eigenvalue, as is 1 whenever $n_f > n_c$.

Proof. The definition of γ gives bounds on the eigenvalues of $A_c^{-1} F^T A_{ff} F$, which are real.

$$0 \leq \inf_{\mathbf{v}\neq 0} \frac{\|F\mathbf{v}\|_{A_{ff}}^2}{\|\mathbf{v}\|_{A_c}^2} \leq \lambda(A_c^{-1} F^T A_{ff} F) \leq \sup_{\mathbf{v}\neq 0} \frac{\|F\mathbf{v}\|_{A_{ff}}^2}{\|\mathbf{v}\|_{A_c}^2} = \gamma^2$$

The upper bound is tight, by definition. Cyclic permutations of matrix products leave the spectrum invariant except for a change in the presence and/or multiplicity of the zero eigenvalue accounting for changes in dimension. Hence,

$$0 \leq \lambda(F A_c^{-1} F^T A_{ff}) \leq \gamma^2.$$

Note that the dimension changed from n_c to n_f. If this is an increase ($n_f > n_c$), the spectrum must include 0, making the lower bound tight. If we use that $F := A_{ff}^{-1}\hat{A}_{fc}$ in the definition of S_f,

$$S_f := A_{ff} - \hat{A}_{fc}A_c^{-1}\hat{A}_{fc}^T = A_{ff} - A_{ff}FA_c^{-1}F^TA_{ff},$$

we see that

$$\lambda(A_{ff}^{-1}S_f) = 1 - \lambda(FA_c^{-1}F^TA_{ff}),$$

and the result follows. □

We can use this result to remove the matrix S_f from the convergence result of Theorem 1, resulting in an inequality involving γ. We will use the following basic linear algebra fact.

Lemma 2. *For all SPD matrices X and symmetric positive semi-definite matrices A and B, all of the same size,*

$$\lambda_{\min}(AX^{-1})\lambda_{\min}(XB) \leq \lambda(AB) \leq \lambda_{\max}(AX^{-1})\lambda_{\max}(XB).$$

Proof. The second inequality is a consequence of the submultiplicative property of the Euclidean matrix norm.

$$\begin{aligned}\lambda_{\max}(AB) = \|B^{\frac{1}{2}}AB^{\frac{1}{2}}\|_2 &\leq \|B^{\frac{1}{2}}X^{\frac{1}{2}}\|_2\|X^{-\frac{1}{2}}AX^{-\frac{1}{2}}\|_2\|X^{\frac{1}{2}}B^{\frac{1}{2}}\|_2 \\ &= \lambda_{\max}(AX^{-1})\|B^{\frac{1}{2}}X^{\frac{1}{2}}\|_2^2 \\ &= \lambda_{\max}(AX^{-1})\lambda_{\max}(X^{\frac{1}{2}}BX^{\frac{1}{2}}) \\ &= \lambda_{\max}(AX^{-1})\lambda_{\max}(XB)\end{aligned}$$

The first inequality reduces to $0 \leq 0$ if either A or B is singular. Otherwise, we can apply the inequality just proved to A^{-1} and B^{-1}:

$$\lambda_{\max}(B^{-1}A^{-1}) \leq \lambda_{\max}(B^{-1}X^{-1})\lambda_{\max}(XA^{-1}).$$

Inverting both sides then yields the first inequality. □

Theorem 2. *If $\hat{B}_{ff}$ is symmetric and*

$$\rho(I - \hat{B}_{ff}A_{ff}) \leq \rho_f < 1,$$

then

$$\rho(E_{tg}) \leq 1 - (1-\gamma^2)(1-\rho_f).$$

Proof. We have that

$$0 < 1 - \rho_f \leq \lambda(\hat{B}_{ff}A_{ff}) \leq 1 + \rho_f. \tag{8}$$

In particular, $A_{ff}^{\frac{1}{2}}\hat{B}_{ff}A_{ff}^{\frac{1}{2}}$ is SPD, which implies that $\hat{B}_{ff}$ is. Thus, we may apply Lemma 2 with $A \leftarrow \hat{B}_{ff}$, $X \leftarrow A_{ff}^{-1}$, and $B \leftarrow S_f$, using the bounds from Lemma 1 and equation (8) to find that

$$(1-\rho_f)(1-\gamma^2) \leq \lambda(\hat{B}_{ff}S_f) \leq 1 + \rho_f.$$

From Theorem 1, $\lambda(E_{\text{tg}}) = \lambda(I - \hat{B}_{ff}S_f) \cup \{0^{n_c}\}$, so

$$-\rho_f \leq \lambda(E_{\text{tg}}) \leq 1 - (1-\rho_f)(1-\gamma^2).$$

□

Theorem 2 is closely related to Theorem 4.2 of the analysis by Falgout, Vassilevski, and Zikatanov [8] mentioned when we introduced γ earlier in this section. That theorem is restricted to symmetrically paired pre- and post-F-relaxations, which, in our notation, means matrices B of the form

$$B = \begin{bmatrix} M^{-1} + M^{-T} - M^{-1} A_{ff} M^{-T} & O \\ O & O \end{bmatrix}. \tag{9}$$

We used Theorem 1 to reduce the general case to an F-relaxation, whereas Falgout, Vassilevski, and Zikatanov pursue a different line of analysis to handle the general case, although a significant amount of the treatment was unified. Also, for the matrix B of (9), we have that $0 \leq \lambda(I - \hat{B}_{ff} A_{ff})$. This is not generally the case when, for example, presmoothing is not done; Theorem 2 still applies in this circumstance. We hope that readers familiar with the paper by Falgout, Vassilevski, and Zikatanov will benefit from the different point of view taken in our treatment. We also mention that the symmetry requirement of $\hat{B}_{ff}$ may be dropped from Theorem 2 by applying the theorem to a symmetrized smoother, in which case the spectral radius bound weakens to an energy norm bound.

3 Independent Quality Measures

Using the convergence theorem of the preceding section, we propose specific quantifications of the "quality" of each component of the two-grid iteration, as listed in Table 1.

Table 1: AMG Component Quality Measures

Component	Quality	Cost
Coarsening	$\kappa_f := \kappa(D_{ff}^{-\frac{1}{2}} A_{ff} D_{ff}^{-\frac{1}{2}})$	n_c/n
Interpolation	$\gamma := \|A_{ff}^{\frac{1}{2}} F A_c^{-\frac{1}{2}}\|_2$	$\text{nnz}(W)$
Smoother	$\rho_f := \rho(I - \hat{B}_{ff} A_{ff})$	cost of applying B

$$F := W - (-A_{ff}^{-1} A_{fc}), \qquad \hat{B}_{ff} := \begin{bmatrix} I & -W \end{bmatrix} B \begin{bmatrix} I & -W \end{bmatrix}^T$$

For easy reference, we have repeated definitions (7) and (6) from the previous section of the departure from the ideal weights F and the projection of the smoother to an F-relaxation $\hat{B}_{ff}$. We have not introduced the condition number κ_f until just now; D_{ff} denotes the diagonal part of A_{ff} in its definition. In all cases, smaller numbers in the table are better. Before explaining the theoretical justification of the quality measures, we highlight a few properties. First, the measures have no forward dependencies. That is, κ_f is independent of the weights W and smoother B, while γ is independent of the smoother B. We also have ρ_f independent of W in the particular case of the smoother being an F-relaxation. Second, in each row, the quality measure is in opposition to the cost indicator. For example, changing some C-variables to F-variables obviously improves the coarsening ratio but will generally increase κ_f, since the new diagonally scaled A_{ff} will include the old as a submatrix.

The justification for γ and ρ_f as quality measures is the convergence bound of Theorem 2, $\rho(E_{\text{tg}}) \leq \rho_f + \gamma^2 - \rho_f \gamma^2$. When either γ or ρ_f is made smaller, the convergence bound improves.

The justification of κ_f is that it is tied to whether a low-cost set of interpolation weights and low-cost smoother can be found that are also of high quality. In the case of the smoother, this should be clear: in particular, when κ_f is small, a few steps of damped Jacobi iteration suffice to make ρ_f small. In the case of the interpolation weights, the reason is related. It is a result by Demko, Moss, and Smith [6] that the entries of the inverse of a sparse matrix X decay exponentially,

$$[X^{-1}]_{ij} \leq c q^{|i-j|_G - 1}, \qquad q := \frac{\sqrt{\kappa(X)} - 1}{\sqrt{\kappa(X)} + 1}, \tag{10}$$

for some constant c. We have used the notation used by Brannick and Zikatanov [3], and contributed by Vassilevski, who recognized the relevance of the result to AMG. The quantity $|i-j|_G$ denotes the distance between unknowns i and j in the adjacency graph of X and can also be characterized as the smallest k for which $[X^k]_{ij}$ is nonzero; the latter notion was used in the original statement by Demko, Moss, and Smith [6]. There are interesting connections of (10) with iterative linear solution methods. The quantity $|i-j|_G$ is also the number of Jacobi iterations before the value b_j affects x_i when solving $X\mathbf{x} = \mathbf{b}$. The constant q appears in the standard convergence theory of the conjugate gradient (CG) method. This is not surprising given that the kth iteration of CG constructs an optimal polynomial $a_0 X^0 + a_1 X^1 + \cdots + a_k X^k$ approximating X^{-1}.

Now consider the ideal weights $W = -A_{ff}^{-1} A_{fc}$ for which $\gamma = 0$. These may be written

$$-D_{ff}^{-\frac{1}{2}} (D_{ff}^{-\frac{1}{2}} A_{ff} D_{ff}^{-\frac{1}{2}})^{-1} (D_{ff}^{-\frac{1}{2}} A_{fc}).$$

Because of the inverse appearing as the middle factor, these weights are not sparse. However, if we use (10) with $X = D_{ff}^{-\frac{1}{2}} A_{ff} D_{ff}^{-\frac{1}{2}}$, this factor has matrix entries decaying exponentially at a rate controlled by κ_f. In constructing a sparse approximation W to the ideal weights, we can think of dropping those entries that are small enough that γ, a measure of the error made, remains below some value. Clearly, when the coarsening is good and κ_f is small, the number of nonzeros $\text{nnz}(W)$ required to achieve a fixed γ should also be small.

Our notion of coarsening quality bears a strong resemblance to Brandt's concept of coarsening by "compatible relaxation" [2]. Indeed, κ_f characterizes the convergence rate of optimally damped Jacobi on the F-variables. However, we do not assume a smoother of this type; rather, our notion of coarsening quality is independent of the smoother. It was in the context of compatible relaxation that Brannick and Zikatanov [3] brought up the result of Demko, Moss, and Smith [6], and indeed they argue that the conditioning of A_{ff} and the convergence rate of compatible relaxation are correlated. However, the smoother is involved neither in the ideal weights nor in the theoretical measure γ of the quality of given nonideal weights, and for this reason we abandoned the compatible relaxation philosophy tying the coarsening quality to the smoother.

We have already remarked that our quantity γ is not new and is important in the context of the hierarchical basis method. It is perhaps the particular projection of the smoother to an F-relaxation, theoretically justified by Theorem 1, that makes ρ_f novel—although this projection is just the one associated with the hierarchical decomposition. We also note that ρ_f is itself a compatible relaxation convergence rate, although we use it to characterize smoother quality and not coarsening quality, as unlike κ_f it has no bearing on the sparsity of the interpolation weights.

4 Component Heuristics

In this section we present the heuristic procedures we developed to construct each component, guided by the theoretical quality measures. In each case, the heuristic procedure has a free parameter, a target value for the corresponding quality measure, that can be used to adjust the balance between cost and quality. The lack of forward dependencies in the measures is crucial here, because we can construct each component in order, without worrying about the effect of the components that have yet to be constructed.

4.1 Coarsening

Our coarsening procedure has a simple theoretical grounding. Let $X := I - D_{ff}^{-\frac{1}{2}} A_{ff} D_{ff}^{-\frac{1}{2}}$, so that $\kappa_f = \kappa(I - X)$. Notice that the diagonal of X consists of all zeros. Hence, we may write the Gershgorin disc radii associated with X as

$$r_i := \mathbf{e}_i^T |X| \mathbf{1},$$

where the absolute value of X is taken entrywise, $\mathbf{1}$ denotes the vector of all ones, and $\mathbf{e}_i$ denotes the ith column of the identity matrix. Because the discs are all centered at 0, we have the bound

$$\kappa_f \leq \frac{1 + r_{\max}}{1 - r_{\max}} \qquad \text{when} \qquad r_{\max} < 1, \tag{11}$$

where $r_{\max} := \max_i r_i$. This leads us to the following simple procedure, with R a given parameter:

1. Start with no C-variables.
2. Change those F-variables with locally maximal r_i to C-variables.
3. Recompute the remaining r_i; go back to previous step unless $r_{\max} < R$.

By "locally maximal" in step 2, we mean larger than neighbors in the adjacency graph. In the notation of the previous section we say r_i is locally maximal if $r_i \geq r_j$ whenever $|i - j|_G = 1$. We resolve ties by referring to the arbitrary ordering of unknowns.

We highlight some features of this simple algorithm. First, positive off-diagonals pose no difficulty and require no special treatment. The

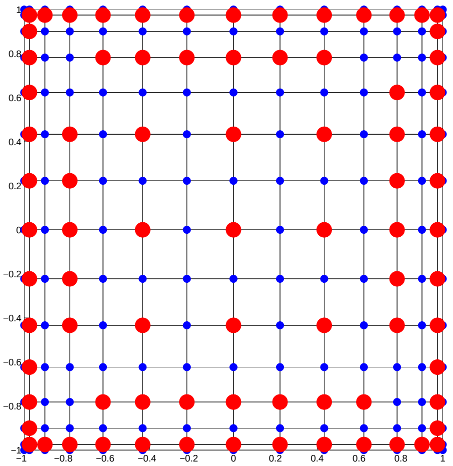

Figure 1: Algebraic coarsening of 9-pt Laplacian on Chebyshev mesh

matrix X resembles the "strength of connection" heuristic, but we don't use it to classify connections as one of "strong" or "weak." The disc radius r_i resembles the "number of strong connections" heuristic but is not integral. The procedure is inherently parallel and resembles the CLJP coarsening procedure of Cleary et al. [5]; but because r_i is not integral, we do not need to add a random number to it to create local maxima (although that's a possible strategy for resolving ties). Most important, the confidence in our algorithm comes from the bound (11) it provides on our theoretical quality measure.

In Figure 1, we show the coarsening obtained by this method for a 9-pt Laplacian on a 15×15 grid located at Chebyshev nodal coordinates in both directions. The grid exhibits strong anisotropy toward the boundaries. Note that the algebraic coarsening procedure correctly semi-coarsens where required, in the appropriate direction. In this example, we used the maximum Gershgorin disc radius $R = 0.9$, which provides the bound $\kappa_f < 19$; this coarsening achieves $\kappa_f = 3.6$.

4.2 Interpolation

We divide the problem of constructing the interpolation weights W into two subproblems: (1) finding the support, or sparsity pattern or skeleton, of W and (2) finding the numerical entries given the support. Our procedure expands the support of W iteratively, at each iteration solving subproblem (2). As such, we look at subproblem (2) first.

4.2.1 Numerical Weights

Ideally, for a fixed support, we would like to choose the entries of W in order to minimize the quality measure $\gamma := \|A_{ff}^{\frac{1}{2}} F A_c^{-\frac{1}{2}}\|_2$. A tractable substitute problem is instead to minimize the Frobenius norm $\|A_{ff}^{\frac{1}{2}} F D\|_F$, for some diagonal matrix D. The matrix D does not affect the minimizer, and moreover this is equivalent to minimizing the trace norm $\mathrm{tr}(P^T A P)$. This energy-minimizing approach was originally suggested by Wan, Chan, and Smith [14]; see also the article by Xu and Zikatanov [15]. Simply stated, with the sparsity pattern fixed in advance, the weights W are chosen to

$$\text{minimize } \mathrm{tr}(P^T A P) \qquad \text{subject to } W\mathbf{u} = -A_{ff}^{-1} A_{fc}\mathbf{u}, \tag{12}$$

where $\mathbf{u}$ is a near-null space vector specified at the C-variables only. For the discretized Laplace operator, as in our application, the appropriate choice is $\mathbf{u} = \mathbf{1}$. Note that the small singular values of A_c can dominate the norm γ but that we lose this information in the Frobenius norm approximation. Hence, the constraint, which ensures that one near-nullspace vector is interpolated ideally, is crucial. For problems of interest, this is sufficient to ensure that all singular vectors corresponding to small singular values are interpolated well enough so that γ remains small.

We find it fascinating that the vector of Lagrange multipliers $\boldsymbol{\lambda}$ for (12) is governed by a standard overlapping additive Schwarz preconditioner for A_{ff}, as remarked by Brannick and Zikatanov [3]. If we let R_i be the matrix consisting of rows of the $n_f \times n_f$ identity matrix that restricts to the support of column i of W and let

$$X_i := R_i^T (R_i A_{ff} R_i^T)^{-1} R_i,$$

then the solution to (12) is given by

$$W\mathbf{e}_i = X_i(-A_{fc}\mathbf{e}_i + u_i\boldsymbol{\lambda}),$$

where

$$X\boldsymbol{\lambda} = -A_{ff}^{-1} A_{fc}\mathbf{u} - \sum_{i=1}^{n_c} u_i X_i(-A_{fc}\mathbf{e}_i), \qquad X := \sum_{i=1}^{n_c} u_i^2 X_i. \tag{13}$$

For details of this solution, we refer the reader to the references given above. We have introduced only the straightforward generalization of letting $\mathbf{u}$ be different from $\mathbf{1}$ (and the astute reader will have noticed that X becomes some sort of weighted overlapping Schwarz preconditioner in that circumstance). One must be careful about ensuring that X is not singular, as can happen when some of the u_i are zero or the support sets are too small. In that circumstance, the remedy is to take $\lambda_i = 0$ wherever $X_{ii} = 0$ and solve only for the other λ_i.

In practice, we compute $-A_{ff}^{-1} A_{fc}\mathbf{u}$ by the Jacobi preconditioned CG method, which has a convergence rate controlled by κ_f. Because the coarsening ensured that κ_f is small, convergence is very fast. We also use the preconditioned CG to solve (13), observing that, as X is a standard preconditioner for A_{ff}, symmetrically A_{ff} is a good preconditioner for X. We go one step further with a diagonal scaling D chosen so that $DA_{ff}X$, and hence $XA_{ff}D$, has ones along the diagonal. We use the symmetric preconditioner

$$\frac{1}{2}(DA_{ff} + A_{ff}D) \approx X^{-1}, \quad D := \operatorname{diag}(d_j^{-1}), \quad d_j := \sum_{i=1}^{n_c} u_i^2 \mathbf{e}_j^T R_i^T R_i \mathbf{e}_j. \tag{14}$$

Again convergence is very fast because, diagonally scaled, A_{ff} is well-conditioned to begin with. In practice, a typical iteration count to compute $\boldsymbol{\lambda}$ to full machine precision is 15.

4.2.2 Interpolation Support

To determine the optimal support for W, we again appeal to an approximation of the theoretical quality measure γ. Here we use the approximation

$$\gamma := \|A_{ff}^{-\frac{1}{2}} \hat{A}_{fc} A_c^{-1}\|_2 \approx \|D_{ff}^{-\frac{1}{2}} \hat{A}_{fc} D_c^{-1}\|_2,$$

where D_{ff} and D_c are the diagonal parts of A_{ff} and A_c and where the reader will recall $\hat{A}_{fc} := A_{ff}W + A_{fc} = A_{ff}F$. We use a bound on the 2-norm given by Nikiforov [11],

Theorem 3 (Nikiforov [11])**.** *For all $m \times n$ complex matrices A, and integers $r \geq 0$, $p \geq 1$,*

$$\|A\|_2^{2p} \leq \max_{i, w_i^{(r)} \neq 0} \frac{w_i^{(r+p)}}{w_i^{(r)}}, \qquad \mathbf{w}^{(r)} := (|A||A|^H)^r \mathbf{1},$$

where the absolute value is taken entrywise, $\mathbf{1}$ denotes the vector of all ones, and A^H denotes the conjugate transpose.

We apply this bound by taking

$$R := |D_{ff}^{-\frac{1}{2}} \hat{A}_{fc} D_c^{-1}|, \quad \mathbf{w}^{(r)} := (RR^T)^r \mathbf{1}, \quad c_i := \sqrt{w_i^{(2)}/w_i^{(1)}},$$

so that

$$\gamma \approx \|D_{ff}^{-\frac{1}{2}} \hat{A}_{fc} D_c^{-1}\|_2 \leq c_{\max}, \qquad c_{\max} := \max_{1 \leq i \leq n_c} c_i.$$

This leads to the following simple procedure for determining the support of W, given a target value γ_{goal} of the weights quality:

1. Start with an empty skeleton for W.
2. Recompute the entries in W for the current skeleton by solving (12).
3. Recompute $\hat{A}_{fc}$, D_c, R, $\mathbf{w}^{(1)}$, $\mathbf{w}^{(2)}$, and $\mathbf{c}$.
4. If $c_{\max} \leq \gamma_{\text{goal}}$, stop.
5. Otherwise, for each column i of W such that $c_i > (1-\epsilon)\gamma_{\text{goal}}$, add to the skeleton of W the entry corresponding to the largest value in $|R|\mathbf{e}_i$, and return to step 2.

In step 5, we take $\epsilon = 0.01$. Before discussing the parameter ϵ and the complications caused by the constraint in the minimization problem of step 2, we highlight a few features of this procedure. At most one nonzero per column is added per iteration. In step 5, we use c_i as a heuristic indication of when the support of column i is insufficient. The idea is to add nonzeros only to those columns that need them. Compared to a simple drop tolerance, in which the decision for when the support of a column is sufficient is made independently from other columns, the heuristic c_i includes far more context. As a result, we found in practice that the above procedure was far more robust, achieving values of γ closer to the target γ_{goal}, and also far more efficient, producing much lower $\text{nnz}(W)$ for a given γ. For our application, setup time is immaterial, making the implications of increased complication and cost during setup worthwhile.

In practice, we start not with an empty skeleton but rather a minimal skeleton (one nonzero per row), such that it is possible to satisfy the constraint $W\mathbf{u} = -A_{ff}^{-1}A_{fc}\mathbf{u}$. In step 2, we do not attempt to solve (13) for the Lagrange multipliers exactly but simply use the approximate inverse (14) once. Without the constraint, it would be the case that $R_{ij} = 0$ wherever the skeleton of W has an entry. With the constraint present, this is no longer true. If the largest value of R happens to occur where W already has an entry, then in step 5 the procedure attempts to add an already present entry to W. We remedy this situation by treating any row where this occurs as having insufficient support to handle the constraint. We find the largest value of R in any such row and add the corresponding entry to W, constraining the search to give only new entries in W.

We originally used $\epsilon = 0$ in step 5 but noticed that the algorithm would typically quickly finish all but one or two columns. These last columns would have values of c_i just slightly above γ_{goal} and converging very slowly to γ_{goal} as nonzeros are added. Taking $\epsilon = 0.01$ appears to solve this problem, allowing the iteration to stop before the convergence stalls.

4.3 Smoother

If the smoother is an F-relaxation, that is, if B takes the form

$$B = \begin{bmatrix} B_{ff} & \\ & O \end{bmatrix},$$

then in this case $\hat{B}_{ff} = B_{ff}$. Thus, an obvious approach to minimizing $\rho_f := \rho(I - \hat{B}_{ff}A_{ff})$ is to use an F-relaxation with B_{ff} chosen as a suitable preconditioner for A_{ff}.

Note that the restriction to F-relaxations is not in itself limiting in that it is still possible to achieve arbitrarily small two-grid convergence rates. In particular, from Theorem 1 we see that the impractical choice $B_{ff} = S_f^{-1}$ results in a direct method with $E_{\text{tg}}^2 = O$. Of course, the matrix S_f involves A_c^{-1} and is not available in practice. The approach described above corresponds to using A_{ff} as a surrogate for S_f. Recall that the two matrices are spectrally equivalent by Lemma 1.

F-relaxations have at least one big advantage over general smoothers. Any standard Krylov subspace method may be used to accelerate an F-relaxation and reliably lower $\rho(I - B_{ff}A_{ff})$. In contrast, it is far from clear how to use Krylov subspace methods to improve the smoothing properties of a general smoother. Adams et al. [1] investigated the use of Chebyshev polynomials in multigrid smoothers; they use an ad hoc smallest eigenvalue parameter to ensure the polynomial does not target the slowly decaying smooth modes. We also note that combining all smoothing steps under one Krylov subspace iteration, making the overall multigrid iteration nonsymmetric, is generally able to achieve lower ρ_f than separate, symmetric pre- and postsmoothing stages.

Our choice is to do postsmoothing only, using a diagonal sparse approximate inverse (SAI) preconditioner for A_{ff}, accelerated by the Chebyshev semi-iterative method. The nonsymmetry is not an issue for our application. We choose the number of iterations such that $\rho_f \approx \gamma^2$ and the terms in the convergence bound are approximately balanced. The use of SAI preconditioners as multigrid smoothers was suggested by Tang and Wan [12]. Diagonal SAI is an easily computable alternative to Jacobi and is optimal in a certain Frobenius norm. We opted to use Chebyshev over conjugate gradient iteration so that the full multigrid iteration remains linear, and also to avoid the need for global inner products.

5 Numerical Results

We present two applications of our AMG approach, both taken from large-eddy simulations of turbulent fluid flow in a reactor core [9]. The spectral element method (SEM) Navier-Stokes code Nek5000 used for these simulations requires the solution of a global "coarse" Poisson solve for the pressure field, as part of the pressure solver. Here "coarse" means that the problem has been reduced to bilinear elements, representing a reduction in the number of unknowns by approximately a factor of N^3, where N is the polynomial order (e.g., $N = 7$ or $N = 11$). For the large problem sizes being encountered, the fast parallel direct solver [13] used for this purpose was becoming the computational bottleneck. This situation motivated the work on AMG reported here, which was used as a replacement.

First we consider the mesh of Figure 2, which is a single 2D slice from a 7-pin reactor geometry. This mesh was imported into Matlab, and test matrix A was constructed by using a standard FEM discretization of the Laplacian operator on a 9-point stencil. This toy problem, comprising 1,422 unknowns, provided a test case for the AMG heuristics presented in the previous section, which were implemented in Matlab. In Table 2 we report relevant quantities for two hierarchies. The first was constructed for an overall target convergence rate of $\rho_{\text{target}} = 0.3$, the second for $\rho_{\text{target}} = 0.05$. The full hierarchy of algebraic grids for the first case is displayed in Figure 2, with nodes at each level corresponding to C-variables rendered larger and red, and nodes corresponding to F-variables smaller and blue. In each level, m gives the number of Chebyshev iterations performed, where $\rho_{f,m=1}$ is the convergence rate of the diagonal preconditioner on its own. The column $\rho(E_{\text{mg}})$ records the full multigrid convergence rate, not just the two-grid rate. The final column, $\text{nnz}(W)/n_c$, records the average number of nonzeros per column in the interpolation weights W. Note that in each case, the AMG heuristics constructed a method with a convergence rate close to the target. Also note that the more accurate hierarchy required more nonzeros in W and more smoothing steps, reflecting the trade-off

Table 2: AMG Hierarchies for 2D Example

(a) $\rho_{\text{target}} = 0.3$, $\gamma_{\text{target}} = 0.4$

Level	n	n_c/n	$\rho_{f,m=1}$	m	γ	$\rho(E_{\text{mg}})$	$\frac{\text{nnz}(W)}{n_c}$
1	1422	0.42	0.49	2	0.34	0.31	3.6
2	594	0.30	0.55	3	0.39	0.30	7.1
3	178	0.27	0.70	3	0.44	0.26	6.3
4	48	0.27	0.72	3	0.34	0.20	6.3
5	13	0.38	0.59	3	0.20	0.16	4.2
6	5	0.20	0.67	3	0	0.11	4.0

(b) $\rho_{\text{target}} = 0.05$, $\gamma_{\text{target}} = 0.16$

Level	n	n_c/n	$\rho_{f,m=1}$	m	γ	$\rho(E_{\text{mg}})$	$\frac{\text{nnz}(W)}{n_c}$
1	1422	0.42	0.49	4	0.17	0.062	5.6
2	594	0.29	0.61	5	0.20	0.057	10.4
3	173	0.28	0.70	5	0.17	0.047	8.9
4	48	0.25	0.72	6	0.12	0.030	9.8
5	12	0.25	0.73	6	0	0.014	9.0
6	3	0.33	0.44	4	0	0.006	2.0

between cost and quality.

Next we consider a 3D problem with 417,600 unknowns. This was the "coarse" problem from a Nek5000 run on a 19-pin geometry with an original problem size of 120 million [9]. The matrix for the coarse problem was processed off-line in Matlab to generate the AMG hierarchy. This data was then read in and used by the AMG solver of Nek5000, implemented in C using MPI. We report some properties of this hierarchy in Table 3. The "?" symbols in the γ column reflect the fact that our explicit method of computing this quantity was not viable beyond a few thousand unknowns. We include an additional column in this table, the average number of nonzeros in a column of $\tilde{A}_{ff}$. Here the matrix $\tilde{A}_{ff}$, used in the Chebyshev semi-iteration for the residual update, is formed by dropping small entries in A_{ff}. Robust bounds are used to ensure the damage to ρ_f is small. This technique was used to combat the stencil growth in AMG methods, which, as Table 3 shows, our method does not escape. The actual convergence rate achieved, 0.67, was small enough so that the iteration count in Nek5000 was unaffected when the direct solver was replaced by the approximate AMG solver.

In Table 4, we list run times on the IBM BG/P at Argonne National Laboratory for Nek5000 that were reported in [9] for the 19-pin geometry with 120M unknowns. Here P is the number of processors and n/P is the average number of unknowns per processor. The "Total" column lists the overall time of the Nek5000 run, while the "Solver" column lists the time spent in the "coarse" solver, the component replaced with the approximate

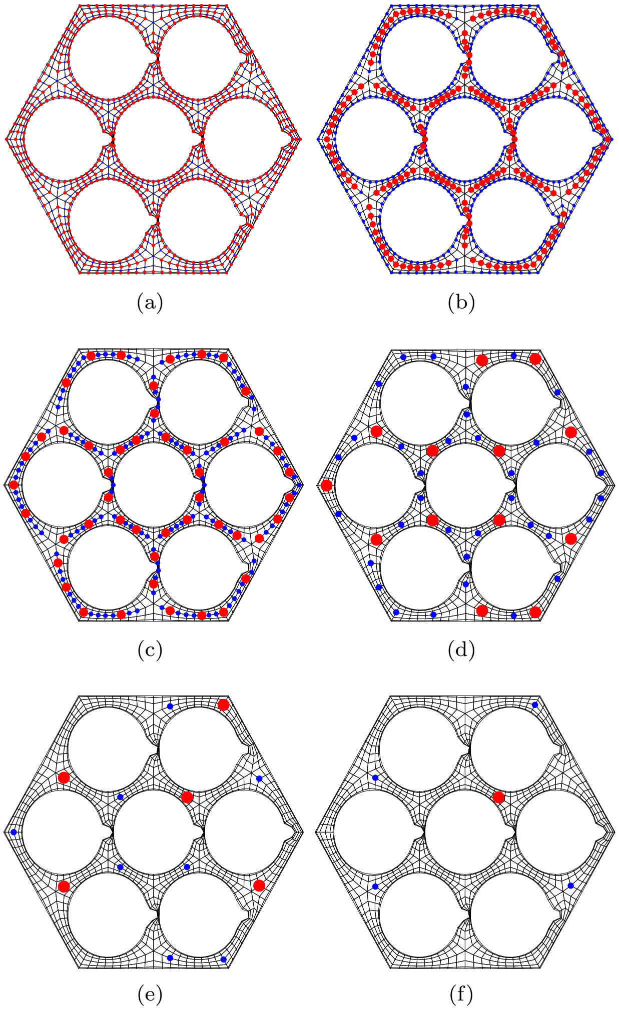

Figure 2: Six levels of the AMG hierarchy for 2D example. C-variables are shown as large red dots and F-variables as small blue dots.

Table 3: 3D Application Hierarchy; $\rho_{\text{targ}} = 0.5$, $\gamma_{\text{targ}} = 0.54$

Level	n	n_c/n	$\rho_{f,m=1}$	m	γ	$\rho(E_{\text{mg}})$	$\frac{\text{nnz}(W)}{n_c}$	$\frac{\text{nnz}(\tilde{A}_{ff})}{n_f}$
1	417600	0.36	0.80	3	?	0.67	8.3	9.4
2	151248	0.44	0.61	2	?	0.63	7.9	22.0
3	66887	0.43	0.59	2	?	0.60	8.8	30.8
4	28862	0.33	0.62	2	?	0.57	12.8	41.5
5	9471	0.22	0.68	3	?	0.55	21.4	32.8
6	2116	0.18	0.67	2	0.42	0.51	36.4	86.6
7	390	0.20	0.60	2	0.42	0.48	35.8	53.8
8	79	0.16	0.72	3	0.61	0.46	33.3	43.8
9	13	0.23	0.62	2	0.39	0.35	9.3	10.0
10	3	0.33	0.44	2	0	0.11	2.0	2.0

Table 4: BG/P Run Times (seconds)

P	n/P	Method	Solver	Total
4096	102	direct	1180	1994
4096	102	AMG 1	192	1112
4096	102	AMG 2	25	846
8192	51	AMG 2	22	460
16384	25	AMG 2	20	266

AMG solver reported on in this paper. In the first row, the run used the original direct solver. The runs for the remaining rows used two variant implementations of AMG, using the hierarchy summarized in Table 3.

Note that even on the finest level, there are only 100 unknowns per processor. The direct solver does not scale to these problem sizes and takes the bulk of the run time. The second AMG solver gives a 47-fold improvement. In considering this remarkable figure, one must keep in mind that many AMG iterations would be required to match the accuracy of the direct solve. For this application, however, only the accuracy of one iteration is required, and all of the extra accuracy provided by the direct solve is wasted. Almost all of the time in the AMG solver is spent on communication, which occurs when matrix vector products are evaluated during the multigrid iteration. The implementation uses a stand-alone general-purpose communication kernel for this task. As reported in [9], this kernel had to be rewritten in order to evaluate the AMG communication patterns efficiently. For each communication pattern, the kernel used in the “AMG 1” run selected between using standard pairwise exchanges and using an `all_reduce()`, which is implemented in hardware on the BG/P. The kernel of the “AMG 2” run also could choose a crystal-router based implementation (which has the feature of requiring maximum $\log_2 P$ messages). See [9] for details. Note that this last optimization resulted in a 7-fold improvement.

6 Conclusion

We have presented an AMG algorithm developed to replace a direct solver used as the "global coarse solve" in an application. Because this global coarse solver is invoked possibly many hundreds of thousands of times in a run, and for the same matrix, it made sense to consider using a possibly expensive setup procedure to find the absolute "best" AMG components—the best coarsening, interpolation, and smoother at each level. While it is not difficult to estimate the cost of each component, the "quality" of each is harder to isolate. The overall convergence rate, for example, depends in a complicated way on all of the components. We drew heavily on the literature to synthesize an understanding of the theory of AMG that allowed each component to be isolated, that is, characterized and able to be optimized independently from the others.

The perspective of AMG theory we presented may be summarized as follows. The convergence theory of the two-grid iteration boils down to how well the projection $\hat{B}_{ff}$ of the smoother, B, onto the F-variables approximates S_f^{-1}, the inverse of the Schur complement of the coarse operator A_c. Indeed, using S_f^{-1} as an F-relaxation results in a direct method (a Schur complement decomposition). Of course, this is completely impractical, because even forming S_f involves A_c^{-1}. The next step then is to characterize the circumstances in which a practical approximation to S_f^{-1} may be found. A first answer is that S_f will equal A_{ff} when we choose A_c to be S_c, the Schur complement of A_{ff}. This characterizes the *ideal* weights. Unfortunately, not being sparse, these are impractical. However, if they are nearly sparse, as happens when A_{ff} is well-conditioned modulo diagonal scaling, we may simply use a sparse approximation. When done in a controlled way, the resulting S_f will still *approximately* equal A_{ff}. Hence our particular characterizations of the AMG components:

- Good coarsening amounts to finding some large submatrix A_{ff}, well-conditioned modulo diagonal scaling.
- Good interpolation amounts to finding a sparse coarse basis nearly A-orthogonal to the space of F-variables.
- Good smoothing amounts to finding an operator whose projection onto the F-variables approximates S_f^{-1}, a spectrally equivalent surrogate for which is provided by A_{ff}^{-1}.

We quantified these notions to provide a metric of quality for each component presented in Table 1 and in such a way that the metrics determine a robust convergence bound on the two-grid iteration.

The heuristics we presented for the AMG components were more or less directly inspired by the theoretical measures of component quality of Table 1. In each case, finding the best component is an optimization problem involving the quality measure, and tractable substitute problems were found by approximating these measures (using bounds, substitute norms, etc.). For interpolation, the existing energy-minimizing approach of Wan, Chan, and Smith [14] fit into this framework. We augmented it with a heavyweight method for finding an optimal support, which may not be suitable for methods for which setup time is important. In contrast, the coarsening procedure we presented, based on Gershgorin discs, is novel but also simple, much simpler than compatible-relaxation-based approaches; but like those it is based on optimizing a theoretical notion of coarsening quality (related, but different in our framework). Indeed, our procedure produces a coarsening that satisfies a concrete bound on our coarsening quality metric.

The strongest evidence for the soundness of our approach is in our numerical results. Specifically, our heuristics were able to produce AMG hierarchies achieving convergence rates remarkably close to the target rate, an input parameter to the heuristics. For the 2D example with 6 levels (Table 2), the achieved contraction factors were 0.31 for the target 0.3 and 0.062 for the target 0.05.

The solver portion of the presented AMG algorithm was implemented in the Nek5000 spectral element code, for use in a highly parallel setting with a very small number of degrees of freedom per processor—that is, as the "global coarse solve." In this context, the communication involved in the matrix vector products completely dominates the solve time. The general-purpose communication library used for this purpose had to be rewritten (with Paul Fischer, see [9]) to handle the patterns involved in the AMG matrix vector products efficiently. This component was critical to the AMG solver's success. An initial version that used only pairwise exchanges actually performed *worse* than the direct solver, whereas the final version with the optimized communication kernel performed almost 50 times faster.

Acknowledgement

This work was supported by the U.S. Department of Energy under contract DE-AC02-06CH11357 and in part by Award No. KUK-C1-013-04, made by King Abdullah University of Science and Technology (KAUST). Computer time was provided by the Argonne Leadership Computing Facility, Argonne National Laboratory. The author thanks Paul Fischer, Barry Smith, and Gareth Shaw for many useful discussions.

References

[1] M. Adams, M. Brezina, J. J. Hu, and R. S. Tuminaro. Parallel multigrid smoothing: polynomial versus Gauss-Seidel. *J. Comp. Phys.*, 188(2):593–610, 2003.

[2] A. Brandt. General highly accurate algebraic coarsening. *Electron. Trans. Numer. Anal.*, 10:1–20, 2000.

[3] J. Brannick and L. Zikatanov. Algebraic multigrid methods based on compatible relaxation and energy minimization. In *Proc. of the 16th Int. Conf. on Domain Decomposition Methods*, 2005.

[4] E. Chow, R. D. Falgout, J. J. Hu, R. S. Tuminaro, and U. M. Yang. A survey of parallelization techniques for multigrid solvers. In *Parallel Processing for Scientific Computing*, SIAM Series on Software, Environment, and Tools. SIAM, 2006.

[5] A. J. Cleary, R. D. Falgout, V. E. Henson, and J. E. Jones. *Solving Irregularly Structured Problems in Parallel*, volume 1457 of *Lecture Notes in Computer Science*, chapter Coarse-grid selection for parallel algebraic multigrid, pages 104–115. Springer, 1998.

[6] S. Demko, W. F. Moss, and P. W. Smith. Decay rates for inverses of band matrices. *Math. Comp.*, 43(168):491–499, October 1984.

[7] R. D. Falgout and P. S. Vassilevski. On generalizing the algebraic multigrid framework. *SIAM J. Numer. Anal.*, 42(4):1669–1693, 2004.

[8] R. D. Falgout, P. S. Vassilevski, and L. T. Zikatanov. On two-grid convergence estimates. *Numer. Linear Algebra Appl.*, 12(5–6):471–494, 2005.

[9] P. Fischer, J. Lottes, D. Pointer, and A. Siegel. Petascale algorithms for reactor hydrodynamics. *Journal of Physics: Conference Series*, 125:012076 (5pp), 2008.

[10] W. D. Gropp and D. E. Keyes. Parallel domain decomposition and the solution of nonlinear systems of equations. In R. Glowinski, Y. A. Kuznetsov, G. Meurant, J. Périaux, and O. B. Widlund, editors, *Fourth International Symposium on Domain Decomposition Methods for Partial Differential Equations*, pages 373–381, 1991.

[11] V. Nikiforov. Revisiting Schur's bound on the largest singular value, 2007. Available at http://arxiv.org/abs/math/0702722.

[12] W.-P. Tang and W. L. Wan. Sparse approximate inverse smoother for multigrid. *SIAM J. Matrix Anal. Appl.*, 21(4):1236–1252, 2000.

[13] H. M. Tufo and P. F. Fischer. Fast parallel direct solvers for coarse grid problems. *J. Par. & Dist. Comput.*, 61:151–177, 2001.

[14] W. L. Wan, T. F. Chan, and B. Smith. An energy-minimizing interpolation for robust multigrid methods. *SIAM J. Sci. Comp.*, 21(4):1632–1649, 1999.

[15] J. Xu and L. Zikatanov. On an energy minimizing basis for algebraic multigrid methods. *Comput. Vis. Sci.*, 7(3–4):121–127, 2004.