LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

Compatible Relaxation and Coarsening in Algebraic Multigrid

J. J. Brannick, R. D. Falgout

September 24, 2009

Disclaimer

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

COMPATIBLE RELAXATION AND COARSENING IN ALGEBRAIC MULTIGRID

JAMES J. BRANNICK* AND ROBERT D. FALGOUT†

Abstract. We introduce a coarsening algorithm for algebraic multigrid (AMG) based on the concept of compatible relaxation (CR). The algorithm is significantly different from standard methods, most notably because it does not rely on any notion of strength of connection. We study its behavior on a number of model problems, and evaluate the performance of an AMG algorithm that incorporates the coarsening approach. Lastly, we introduce a variant of CR that provides a sharper metric of coarse-grid quality and demonstrate its potential with two simple examples.

Key words. algebraic multigrid, compatible relaxation

AMS subject classifications. 65F10, 65N20, 65N30

1. Introduction. Algebraic multigrid (AMG) [5, 3, 18] was introduced as a method for solving linear systems based on multigrid principles, but in a way that requires no explicit knowledge of the problem geometry. The AMG method determines coarse "grids", inter-grid transfer operators, and coarse-grid equations based solely on the matrix entries. Since the original introduction of the method, many AMG algorithms have been developed, targeting different problem classes and yielding different robustness and efficiency properties.

One common feature of all AMG algorithms is a method for choosing coarse grids, which, in the simplest context, can be thought of as a subset of the nodes of the graph of the fine-grid matrix. (We provide a more general definition later.) This paper presents an approach for selecting coarse grids based on the concept of *compatible relaxation* (CR) [4]. The approach differs significantly from most existing methods, but has the additional property that it guarantees the quality of the coarse grid, as established theoretically in [14]. In [15], a sharp AMG convergence theory was developed. Exploiting the similarity between this sharp theory and its predecessor in [14], we also present a CR approach that more accurately predicts the convergence factor of AMG.

The idea of coarsening based on CR first appeared in [4], while the first CR coarsening algorithm appeared in [17]. An outline of the algorithm in this paper was given in [14], and many of the details appeared in [6]. In [8], CR was used to choose coarse grids in an AMG algorithm based on energy-minimization. A method called compatible coarsening was used in [2] to select the fine-coarse partition in the multigraph solver. Lastly, an early version of our CR algorithm was parallelized and studied in [1].

In Section 2, we introduce some notation and summarize the theory underlying our CR-based coarsening algorithm. We then describe the algorithm in Section 3. In Section 4, we study the behavior of the algorithm on a number of model problems, and

we evaluate the performance of an AMG method that incorporates the new coarsening approach. In Section 5, we introduce a more predictive variant of CR and demonstrate its potential with two simple examples.

2. Preliminaries and Theoretical Considerations. In [14], a two-grid variational multigrid theory allowing for more general relaxation and coarsening processes was developed. Several variants of compatible relaxation (CR) were introduced, along with a supporting theory that ties CR convergence to two-grid multigrid convergence. In addition, a general algorithm was outlined for selecting coarse grids in an algebraic multigrid (AMG) method. Our aim in this section is to summarize the theory in [14], as it underpins the CR coarsening algorithms described in this paper.

Consider solving the linear system of equations

$$(2.1) \qquad A\mathbf{u} = \mathbf{f},$$

where $\mathbf{u}, \mathbf{f} \in \mathbb{R}^n$ and A is a symmetric positive definite (SPD) matrix. Define the *smoother* error propagator by

$$(2.2) \qquad I - M^{-1}A,$$

and assume that the smoother is convergent (in energy norm $\|\cdot\|_A$) so that $M + M^T - A$ is SPD. Denote the symmetrized smoother operator by

$$(2.3) \qquad \widetilde{M} = M^T(M^T + M - A)^{-1}M,$$

so that $I - \widetilde{M}^{-1}A = (I - M^{-1}A)(I - M^{-T}A)$. Let $P : \mathbb{R}^{n_c} \mapsto \mathbb{R}^n$ be the *interpolation* (or *prolongation*) operator, where $\mathbb{R}^{n_c}$ is some lower-dimensional (coarse) vector space of size n_c. The two-grid error transfer operator with no post-smoothing steps is then given by

$$(2.4) \qquad E_{TG} = (I - P(P^TAP)^{-1}P^TA)(I - M^{-1}A),$$

where P^T is the *restriction* operator and $A_c = P^TAP$ is the Galerkin *coarse-grid operator*. Note that coarse-grid correction involves an A-orthogonal projection onto range(P).

Let $R : \mathbb{R}^n \mapsto \mathbb{R}^{n_c}$ be any matrix for which $RP = I_c$, the identity on $\mathbb{R}^{n_c}$, so that PR is a projection onto range(P). We can think of R as defining the *coarse-grid variables*, i.e., $\mathbf{u}_c = R\mathbf{u}$. Also, let $S : \mathbb{R}^{n_s} \mapsto \mathbb{R}^n$ be any full-rank matrix for which $RS = 0$, where $n_s = n - n_c$. Here, the unknowns $\mathbf{u}_s = S^T\mathbf{u}$ are analogous to the fine-grid-only variables (i.e., F-points) in AMG. In addition, R and S form an orthogonal decomposition of $\mathbb{R}^n$: any $\mathbf{e}$ can be expressed as $\mathbf{e} = S\mathbf{e}_s + R^T\mathbf{e}_c$, for some $\mathbf{e}_s$ and $\mathbf{e}_c$. The next two theorems summarize the main convergence results in [14].

THEOREM 2.1. *(see Theorem 2.2 in [14])*

$$(2.5) \qquad \|E_{TG}\|_A^2 \le 1 - \frac{1}{K}, \quad where \quad K = \sup_{\mathbf{e}} \frac{\|(I - PR)\mathbf{e}\|_{\widetilde{M}}^2}{\|\mathbf{e}\|_A^2} \ge 1.$$

THEOREM 2.2. *(see Theorem 5.1 in [14])*

$$(2.6) \qquad K \le \eta K_\star, \quad where \quad \eta = \|PR\|_A, \quad K_\star = \inf_{P\,:\,RP=I_c} \sup_{\mathbf{e}} \frac{\|(I - PR)\mathbf{e}\|_{\widetilde{M}}^2}{\|\mathbf{e}\|_A^2}.$$

Theorem 2.1 gives conditions that P must satisfy in order to achieve a fast uniformly convergent multigrid method. It is clear that to make K small, then eigenvectors of A belonging to small eigenvalues must either be interpolated accurately by P or else attenuated efficiently by the smoother (since the denominator is small for these eigenvectors). For brevity, we refer to these as small eigenvectors. The choice of which small eigenvectors to eliminate by smoothing and which to eliminate by coarse-grid correction depends on the "localness" of the modes. Essentially, modes that can be eliminated by a local process (i.e., one that is equivalent to applying an operator with a comparable sparse nonzero structure to A) should be handled by the smoother.

In Theorem 2.2, the constant K is bounded by two new constants, η and $K_\star$. The significance of this theorem is that it separates the construction of P into its natural two components: coarse-grid selection and definition of P's coefficients. The constant $K_\star$ is the K in Theorem 2.1 for the "best" P possible. Hence, $K_\star$ measures the quality of the coarse grid in some sense because, if it is small, we know there exists an interpolation operator that gives good AMG convergence. Said another way, $K_\star$ measures the ability of the coarse-grid variables $R\mathbf{u}$ to represent algebraically smooth error, i.e., error not treated well by the relaxation process. Throughout the remainder of the paper, we use the term *coarse grid* as a synonym for R. Once we have a coarse grid, the expression for η gives us guidance on how to define the coefficients of P. Bounding η is both a necessary and sufficient condition for AMG (two-grid) convergence, and the requirement of accurately interpolating small eigenvectors of A arises naturally here as before. An important aspect of Theorem 2.2 is that $K_\star$ depends on the smoother, but η does not. Hence, this result suggests that, given a coarse grid, the approach for defining P's coefficients need not involve the smoother (although it does not preclude us from using the smoother to define P in practice). The expression for $K_\star$ is analyzed more carefully in the following theorem.

THEOREM 2.3. *(see Theorem 3.1 in [14]) Assume that R, S, and P satisfy $RS = 0$ and $RP = I_c$ as above. Then $K_\star$ in Theorem 2.2 is given by*

(2.7)
$$K_\star = \frac{1}{\lambda_{\min}((S^T \widetilde{M} S)^{-1}(S^T A S))},$$

and the corresponding minimizer is

(2.8)
$$P_\star = (I - S(S^T A S)^{-1} S^T A) R^T.$$

It is easy to see from (2.7) that the coarse grid must be selected such that A is spectrally similar to $\widetilde{M}$ on range(S). In addition, equation (2.8) defines the so-called *ideal interpolation* operator. Notice that, if $K_\star$ is uniformly bounded with respect to parameters such as the mesh spacing, then using $P_\star$ as the interpolation operator results in a uniformly convergent two-grid method. The form of this minimizer cannot in general lead to an efficient method, because the inverse of $S^T A S$ can in general be a dense matrix, implying that the resulting Galerkin coarse-grid operator is also a full matrix. However, we use $P_\star$ to motivate a number of the algorithmic heuristics presented later in this paper.

One way to ensure that $K_\star$ is nicely bounded in practice is through compatible relaxation (CR). The notion of compatible relaxation was introduced by Brandt in [4] as a modified relaxation scheme that keeps the coarse-grid variables invariant. Brandt stated that the convergence rate of CR is a general measure for the quality of the set of

coarse variables (for the given relaxation scheme). This statement was proved in [14] by showing that fast convergence of CR implies a small $K_\star$ (a good coarse grid). We include this result below in Theorem 2.4, but first introduce two variants of CR.

Define the primary CR iteration (represented here by its corresponding error iteration) as

$$(2.9) \qquad \mathbf{e}_s \leftarrow (I_s - M_s^{-1} A_s)\mathbf{e}_s,$$

where $M_s = S^T M S$, $A_s = S^T A S$, $\mathbf{e}_s \in \mathbb{R}^{n_s}$, and I_s denotes the identity on $\mathbb{R}^{n_s}$. Consider, for example, the classical AMG setting where the coarse-grid variables $R\mathbf{u}$ are a subset of the fine-grid variables:

$$(2.10) \qquad R = \begin{bmatrix} 0 & I_c \end{bmatrix}, \quad S = \begin{bmatrix} I_f \\ 0 \end{bmatrix}, \quad P = \begin{bmatrix} W \\ I_c \end{bmatrix}.$$

Here, the iteration in (2.9) is simply F-relaxation. However, for more elaborate forms of M or S, this iteration may become impractical (consider, for example, the M for an overlapping Schwarz method). Another form of CR is the so-called *habituated* compatible relaxation scheme due to Brandt and Livne [17], given by

$$(2.11) \qquad \mathbf{e}_s \leftarrow S^T (I - M^{-1} A) S \mathbf{e}_s,$$

where S must be normalized here so that $S^T S = I_s$. This form of CR is always computable, and it is the easiest to implement in practice because it directly involves the global smoother. Note that when using Habituated CR it is also easy to account for $\nu > 1$ relaxation sweeps in the multigrid method since we can always write $(I - M^{-1}A)^\nu$ in the form $(I - M_\nu^{-1}A)$ for some M_ν.

Define $\sigma(M) = \frac{1}{2}(M + M^T)$ to be the symmetric part of M, and let

$$(2.12) \qquad \varrho_s = \left\| (I_s - M_s^{-1} A_s) \right\|_{A_s}.$$

The next theorem establishes that ϱ_s bounds $K_\star$ and, hence, serves as a measure of the quality of the coarse grid. A similar (but somewhat weaker) bound on $K_\star$ can also be shown for the habituated form of CR.

THEOREM 2.4. *(see Theorem 5.1 in [14])*

$$K_\star \leq \frac{\Delta^2}{2 - \omega} \cdot \frac{1}{1 - \varrho_s},$$

where $\Delta \geq 1$ measures the deviation of M from its symmetric part in the sense that

$$\Delta^2 = \left\| \sigma(M)^{-1/2} M \sigma(M)^{-1/2} \right\|^2,$$

and constant ω satisfies

$$0 < \omega := \lambda_{\max}(\sigma(M)^{-1} A) < 2.$$

To use this result in practice, we must be able to estimate ϱ_s. As we show below, this can be done using the power method, i.e., by monitoring the convergence behavior

of compatible relaxation. Let $E_s = I_s - M_s^{-1} A_s$, and let $\rho(\cdot)$ denote the spectral radius of a matrix. Then

$$
\begin{aligned}
\|E_s\|_{A_s}^2 &:= \sup_{\mathbf{x}} \frac{\langle A_s E_s \mathbf{x}, \ E_s \mathbf{x} \rangle}{\langle A_s \mathbf{x}, \ \mathbf{x} \rangle} \\
&= \rho(A_s^{-1} E_s^T A_s E_s) \\
&= \rho((I_s - M_s^{-T} A_s)(I - M_s^{-1} A_s)).
\end{aligned}
$$

Thus, if $M_s = M_s^T$, then $\|E_s\|_{A_s} = \rho(E_s)$. In this setting, a straightforward approach for estimating ϱ_s is given by applying the power method to E_s. If M_s is nonsymmetric, then ϱ_s can be estimated by applying the power method to the symmetrized smoother.

For an estimate of ϱ_s to be useful, it is clear from Theorem 2.4 that we must also have "good" constants ω and Δ. In particular, ω must be uniformly bounded away from two, which is always possible by using appropriate weighting factors in the relaxation method. In the classical setting, this requirement is equivalent to satisfying a smoothing property; in general, it means that the smoother must effectively damp large eigenvectors of A. Note that this does not preclude the smoother from also damping small eigenvectors. The constant Δ must also be uniformly bounded and as close to 1 as possible. It is equal to 1 when M is symmetric. In the case where M is nonsymmetric, and for additional discussion on these two constants in general, see [14, 6].

3. Practical CR-based Coarse-Grid Selection. In this section, we present an AMG coarsening algorithm based on compatible relaxation. Most existing AMG coarsening algorithms utilize a concept called *strength of connection* to produce effective coarse grids. In classical AMG, variable u_i is said to *strongly depend* on variable u_j if the corresponding matrix coefficient a_{ij} is "large" in the sense that, for some threshold $0 < \theta_s \le 1$,

$$
\tag{3.1} -a_{ij} \ge \theta_s \max_{k \ne i}\{-a_{ik}\}.
$$

It is well known that there are potential problems with this definition of strength of connection. To see this, consider the stencil for a quadrilateral finite element discretization of the Laplacian on a 2D structured mesh that is highly stretched in the x direction:

$$
\begin{bmatrix}
-1 & -4 & -1 \\
2 & 8 & 2 \\
-1 & -4 & -1
\end{bmatrix}.
$$

The problem is strongly anisotropic in the y direction, yet this strong anisotropy is not reflected in the size of the off-diagonal entries. Although it is possible to define better metrics than (3.1) (for example, see [7]), one of our main goals in this paper is to design a coarsening algorithm that does not use strength of connection at all. This is one of the unique features of our algorithm that distinguishes it from nearly all other methods.

For the presented algorithm, we restrict ourselves to the classical AMG setting where the coarse-grid variables are a subset of the fine-grid variables. Here, R and S are defined as in (2.10), inducing a partition of the fine grid, $\mathcal{G} = \{1, ..., n\}$, into C-points, $\mathcal{C}$, and F-points, $\mathcal{F} := \mathcal{G} \setminus \mathcal{C}$. In this setting, it is useful to write A (and

other matrices as well) in the block form

$$(3.2) \qquad A = \left[\begin{array}{cc} A_{ff} & A_{fc} \\ A_{cf} & A_{cc} \end{array} \right],$$

where $A_{ff} = S^T A S$, $A_{fc} = S^T A R^T$, $A_{cf} = RAS$, and $A_{cc} = RAR^T$. The basic CR coarsening algorithm is given in Algorithm 3.1.

ALGORITHM 3.1: *CR-based coarsening* $(\theta_{cr}, \theta_{cs}, \nu)$

    3.1a   Initialize $\mathcal{C} = \emptyset$

    3.1b   Run CR$(\nu) \longrightarrow \varrho_{cr}, \mathbf{e}_s$

    3.1c   While $\varrho_{cr} > \theta_{cr}$

    3.1d   Compute *candidate set measures* $\{\gamma_i : i \notin \mathcal{C}\}$

    3.1e $\mathcal{U} = \{i \notin \mathcal{C} : \gamma_i > \theta_{cs}\}$

    3.1f $\mathcal{C} = \mathcal{C} \cup \{\text{independent set of } \mathcal{U}\}$

    3.1g   Run CR$(\nu) \longrightarrow \varrho_{cr}, \mathbf{e}_s$

    3.1h   End While

The algorithm has three main parameters: $0 < \theta_{cr}, \theta_{cs} < 1$ and ν. The threshold θ_{cr} dictates the CR convergence factor to achieve, and the threshold θ_{cs} is used to determine the so-called *candidate set* $\mathcal{U}$ of potential coarse-grid points. The parameter ν specifies the CR iterate used to define the candidate set measures γ_i. The while loop gives rise to a sequence of C-point sets: $\emptyset = \mathcal{C}_0 \subset \mathcal{C}_1 \subset ... \subset \mathcal{C}_m$, where the final set $\mathcal{C}_m$ defines the coarse grid $\mathcal{G}_c$ and the convergence of CR for this set is below the prescribed tolerance θ_{cr}. We refer to each iteration of the while loop as a *stage*. To design a specific coarse-grid selection algorithm from Algorithm 3.1, there are many choices to make. We describe the details of our algorithm next.

Running CR (steps 3.1b and 3.1g). We run CR on the homogeneous equations $A_s \mathbf{e}_s = 0$ because the error for this equation is known. Hence, the CR iteration is given by its error propagator, as in either (2.9) or (2.11). Let $\mathbf{e}_s^{(k)}$ be the CR iterate at the k-th iteration, where $\mathbf{e}_s^{(0)}$ is the initial guess (discussed below). Define

$$(3.3) \qquad \varrho_{cr}^{(1)} = 1, \quad \varrho_{cr}^{(k)} = \frac{\|\mathbf{e}_s^{(k)}\|_2}{\|\mathbf{e}_s^{(k-1)}\|_2}, \quad k > 1.$$

Then the CR iteration is stopped when at least one of two criteria are met:
    1. At least ν iterations have been done and $\varrho_{cr}^{(k)}$ has "converged".
    2. The CR iteration is converging rapidly.
In criteria 1, we consider $\varrho_{cr}^{(k)} \neq 0$ to have converged when

$$(3.4) \qquad \frac{|\varrho_{cr}^{(k)} - \varrho_{cr}^{(k-1)}|}{\varrho_{cr}^{(k)}} < 0.1.$$

The reason we check for convergence is to prevent the coarsening process from terminating prematurely. This is particularly important in light of the well-known fact that estimates such as $\varrho_{cr}^{(k)}$ generally start out small and converge upward toward

the asymptotic value. With (3.4), we usually see 2 or 3 iterations, which is quite reasonable. In criteria 2, rapid convergence is defined as

$$\varrho_{cr}^{(k)} < (0.1)\,\theta_{cr}. \tag{3.5}$$

If either criteria is met, we set $\varrho_{cr} = \varrho_{cr}^{(k)}$. If criteria 1 is met, we set $\mathbf{e}_s = \mathbf{e}_s^{(\nu)}$. If criteria 2 is met, the while loop terminates and $\mathbf{e}_s$ is not needed. Note that if $\varrho_{cr}^{(k)} = 0$, then criteria 2 is satisfied and (3.4) is not needed.

Computing candidate set measures (step 3.1d). In our current algorithm, the initial guess $\mathbf{e}_s^{(0)}$ in steps 3.1b and 3.1g is the constant vector. In general, the initial guess should be a nonzero vector representative of a near null-space component, restricted to the F-points. This is discussed in more detail below. Define $\bar{\mathbf{e}}_s$ such that $(\bar{\mathbf{e}}_s)_i = (\mathbf{e}_s)_i/(\mathbf{e}_s^{(0)})_i$. Then

$$\gamma_i = \frac{|(\bar{\mathbf{e}}_s)_i|}{\|\bar{\mathbf{e}}_s\|_\infty}, \quad i \notin \mathcal{C}. \tag{3.6}$$

Since we take $\mathbf{e}_s^{(0)} = \mathbf{1}$, we can use $\mathbf{e}_s$ instead of $\bar{\mathbf{e}}_s$ in (3.6).

On the first stage, we use $\theta_{cs} = (0.3)^\nu$ and, on all subsequent stages, θ_{cs} is a fixed parameter. This helps with boundaries, as does the use of ν in the algorithm (see Section 4.1). Another approach for dealing with boundaries was studied in [6]. There, θ_{cs} is set to $1 - \varrho_{cr}$. The idea is that when CR is converging slowly, we want to have a liberal set of points in the candidate set. On the other hand, in later stages of the algorithm when CR is converging faster, this choice of thresholding ensures that points are added to the coarse grid more sparingly.

Since the goal of the coarse-grid correction step is to compensate for deficiencies in the smoother, it makes intuitive sense to choose a coarse grid in regions where the smoothing process is inefficient. In other words, the columns of interpolation should be chosen so that they form a basis over the regions where the error is not quickly damped by relaxation. The candidate set measure γ_i in (3.6) implements this idea, but the initial guess is important. In particular, it is clear that if we choose an initial guess for which relaxation is always effective, we get no information about the coarse grid. Instead, the initial guess should be a representative of a component that interpolation must handle (e.g., the constant vector in our case), while accounting for coarse-grid points that may already have been chosen.

To further motivate our choice of initial guess, consider the two-grid error propagator E_{TG} in (2.4) and recall that the interpolation operator P has the form given in (2.10). At any stage of the algorithm, we know the dimensions of P, but not the coefficients of W. However, R^T has the same form as P (with $W = 0$), and we can assume that P is chosen so that it is better than R^T in the sense that

$$\|E_{TG}\|_A \le \left\|(I - R^T(RAR^T)^{-1}RA)(I - M^{-1}A)\right\|_A.$$

Using the fact that $(I - Q_1)(I - Q_2) = (I - Q_1)$ for any two projection operators Q_1 and Q_2 onto the same space, this implies

$$\|E_{TG}\|_A \le \left\|(I - R^T(RAR^T)^{-1}RA)\right\|_A \left\|(I - R^T R)(I - M^{-1}A)\right\|_A$$
$$= \left\|SS^T(I - M^{-1}A)\right\|_A.$$

The idea is to consider iteration $SS^T(I - M^{-1}A)$ instead of E_{TG} for forming the candidate set. This iteration does relaxation followed by a perfect coarse-grid correction at the C-points. Hence, a natural choice for an initial guess is a smooth vector

restricted to the F-points, i.e., $\mathbf{e}_s^{(0)} = S^T\mathbf{e}$ where $\mathbf{e}$ is the smooth vector ($\mathbf{e} = \mathbf{1}$ in our case). This is essentially habituated CR in (2.11). To take advantage of the fact that we are already doing CR in our algorithm, we just extend this idea to use that CR method.

Choosing an independent set (step 3.1f). The independent set algorithm is designed to distribute C-points in as uniform a fashion as possible. Define the neighborhood of points $i \in \mathcal{U}$ as

$$(3.7) \qquad \mathcal{N}_i = \{j \neq i : a_{ij} \neq 0\}.$$

Assign to each variable $i \in \mathcal{U}$ a weight ω_i given by the number of its neighbors that are not in $\mathcal{C}$ plus its candidate measure, that is $\omega_i = |\mathcal{N}_i \setminus \mathcal{C}| + \gamma_i$. Then one step of the independent set algorithm is as follows:
1. Add to $\mathcal{C}$ the point $i \in \mathcal{U}$ with maximal weight ω_i.
2. Remove from $\mathcal{U}$ the point i and the points $j \in \mathcal{N}_i \cap \mathcal{U}$.
3. For each $j \in \mathcal{N}_i$ removed in step 2, set $\omega_k = \omega_k + 1$, where $k \in \mathcal{N}_j \cap \mathcal{U}$.

This process is repeated until $\mathcal{U}$ is empty. In contrast to the independent set algorithms used in traditional approaches, this algorithm does not use strength of connection. Instead, an independent set is formed using the full graph of the matrix A, truncated by eliminating rows and columns corresponding to points not in $\mathcal{U}$. As a result, our approach generally yields more aggressive coarsening than traditional approaches, especially on increasingly coarser grids, where the stencils of the Galerkin operators tend to grow. Of course, if these stencils grow too large, then it may take many coarsening stages to obtain a grid for which CR is fast to converge. In such cases, the independent set algorithm can be based on the graph of the matrix where relatively small entries have been eliminated. We do not pursue this idea in this paper.

4. Numerical Results. In this section, we study the effectiveness of our CR-based coarsening algorithm (Algorithm 3.1) through a series of numerical experiments. In Section 4.1, we examine the grids chosen by the algorithm for several two-dimensional (2D) model problems for which suitable coarse grids are known, and in Section 4.2, we evaluate the performance of a CR-based multigrid method on both 2D and 3D problems.

Consider the general 2D diffusion equation for $u(x, y)$ given by

$$(4.1) \qquad \begin{aligned} -\nabla \cdot \mathbf{K}\nabla u + (d)u &= f & (x, y) \in \Omega, \\ \text{boundary conditions} & & (x, y) \in \Gamma, \end{aligned}$$

where $\mathbf{K}$ is a tensor of the form

$$(4.2) \qquad \mathbf{K} = \begin{bmatrix} a & c \\ c & b \end{bmatrix}, \qquad \begin{cases} a = \cos^2(\theta) + \epsilon \sin^2(\theta) \\ b = \epsilon \cos^2(\theta) + \sin^2(\theta) \\ c = (1 - \epsilon)\cos(\theta)\sin(\theta) \end{cases}.$$

The parameter $0 < \epsilon \leq 1$ specifies the strength of anisotropy in the problem, while parameter θ specifies the direction of anistropy. Based on (4.1) and (4.2), we define the following 2D test problems.

2D-Lap (2D Laplacian)
$\Omega = [0, 1] \times [0, 1]$; $\epsilon = 1$, $d = 0$; boundary conditions (BC's) = Dirichlet in x (i.e., at $x = 0$ and $x = 1$) and Neumann in y (we call this "mixed").

2D-ALap (2D Anisotropic Laplacian)
$\Omega = [0,1] \times [0,1]$; $\epsilon = 0.01$, $\theta = 0$, $d = 0$; BC's = mixed.

2D-RLap (2D Rotated Anisotropic Laplacian)
$\Omega = [0,1] \times [0,1]$; $\epsilon = 0.01$, $\theta = \pi/3$, $d = 0$; BC's = mixed.

2D-4Reg (2D Four Region Domain)
$\Omega = [0,1] \times [0,1]$; BC's = Dirichlet;

$$
\begin{array}{ll}
\epsilon = 1,\ d = 10^4 & (x,y) \in [0,1/2] \times [0,1/2], \\
\epsilon = 1,\ d = 0 & (x,y) \in (1/2,1] \times [0,1/2], \\
\epsilon = 0.01,\ \theta = 0,\ d = 0 & (x,y) \in [0,1/2] \times (1/2,1], \\
\epsilon = 0.01,\ \theta = \pi/2,\ d = 0 & (x,y) \in (1/2,1] \times (1/2,1].
\end{array}
$$

We use a finite element discretization with bilinear basis functions on meshes composed of either triangles (denoted by 'T') or quadrilaterals (denoted by 'Q'). For example, the notation 2D-ALap-Q indicates the 2D anisotropic Laplacian problem discretized with quadrilateral elements.

Unless otherwise indicated, we use a set of default parameters in our coarsening algorithm. The CR relaxation method is based on pointwise lexicographical Gauss-Seidel (GS) and the primary CR iteration in (2.9). The CR convergence threshold is set to $\theta_{cr} = 0.7$, and the candidate set threshold is set to $\theta_{cs} = 0.5$ (on all stages except the first stage, as discussed earlier). We set $\nu = 2$ so that the second CR iterate is used to define the candidate set measures.

4.1. Grids Chosen for a Set of Model Problems. We first consider problem 2D-4Reg-Q. Figure 4.1 illustrates the coarsening stages and the final coarse grid chosen by our algorithm. Initially, $\mathcal{C} = \emptyset$, $\mathcal{F} = \Omega$, and $\varrho_{cr} = 0.96$. Because $\varrho_{cr} > 0.7$, we enter the while loop in step 3.1c and proceed with the first coarsening stage. Figure 4.1(a) contains a plot of the values of the candidate measures γ_i. The candidate measures are large in three regions of the domain, signaling that coarsening is needed there. The algorithm also correctly detects that coarsening is not needed in the bottom left region of the domain where pointwise Gauss-Seidel is effective. Notice that the candidate measures along the boundaries are generally smaller and, if we set $\nu > 2$, then these smaller values would extend even further into the domain. To compensate for this (as well as to maintain good efficiency), we set ν to a relatively small number. In addition, recall that since this is the first stage of the algorithm, $\theta_{cs} = (0.3)^2 = 0.09$. The first C-point set chosen in step 3.1f is depicted by dotted squares in Figure 4.1(b). For the updated sets $\mathcal{C}$ and $\mathcal{F}$, we obtain $\varrho_{cr} = 0.91$, which leads to a second stage.

The candidate measures γ_i at stage two of the algorithm are plotted in Figure 4.1(b). Since $\theta_{cs} = 0.5$ for this stage, the algorithm signals that further coarsening is needed only in the top two regions of the domain. In addition, the indicated candidate set points are accurate, because they represent points that are neither handled well by pointwise Gauss-Seidel nor interpolated well from the current set of C-points. The result of the second stage is the coarse grid depicted in Figure 4.1(c). The new CR convergence factor is $\varrho_{cr} = 0.49$, so there are no additional stages in the algorithm. Notice that we have semi-coarsening in x in the top-left region where the problem is anisotropic in x, semi-coarsening in y in the top-right where the problem is anisotropic in y, full-coarsening in the bottom-right where the problem is isotropic, and no coarsening in the bottom-left where the mass term dominates.
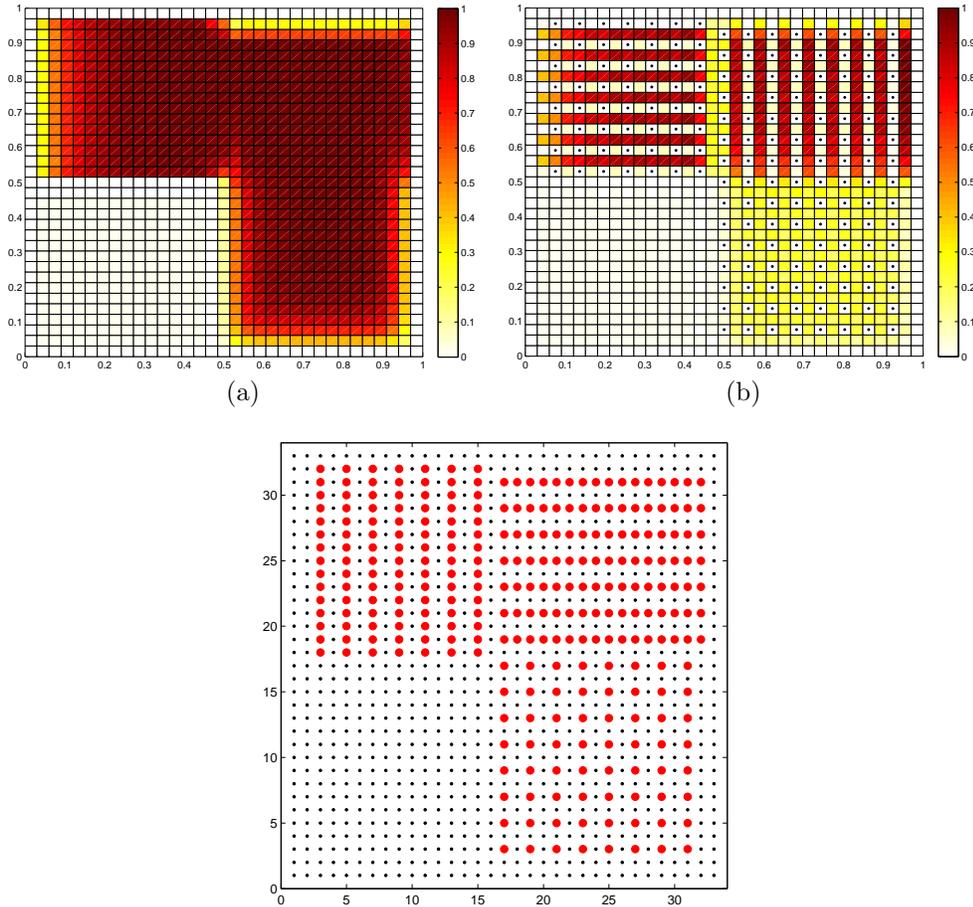
Fig. 4.1. *Candidate measures γ_i computed at step 3.1d during stage one (a) and two (b) for problem 2D-4Reg-Q. Each square depicts the value of γ_i at grid point i according to the color bar. Dotted squares indicate points currently in $\mathcal{C}$ (C-points). The final coarse grid is given by the larger red circles in (c).*

Figure 4.2 illustrates the grids chosen by our algorithm for several model problems. In most cases, the default algorithm generates the expected coarse grid. For the isotropic problems, either full-coarsening (2D-Lap-Q) or red-black coarsening (2D-Lap-T) is done. (Note that 2D-lap-Q has a 9-pt stencil and 2D-Lap-T has a 5-pt stencil.) For the grid-aligned anisotropic problems, a suitable semi-coarsening (2D-ALap-Q) or red-black coarsening (2D-ALap-T) results. For the rotated anisotropic problem 2D-RLap-T, an appropriate semi-coarsening by diagonals is done. (Note that the stencil here is the standard 5-pt stencil plus additional corner couplings to the northeast and southwest.) However, for problem 2D-RLap-Q, a less desirable full-coarsening grid results.

A better coarse grid for problem 2D-RLap-Q can be generated by simply reducing the CR convergence threshold to $\theta_{cr} = 0.5$. This produces another stage, and the resulting grid mixes semi-coarsening in x with semi-coarsening in y. (Either direction is fine for this problem.) The reason a single direction is not achieved is that lexicographical ordering of Gauss-Seidel imprints a bias onto the candidate measures.

10
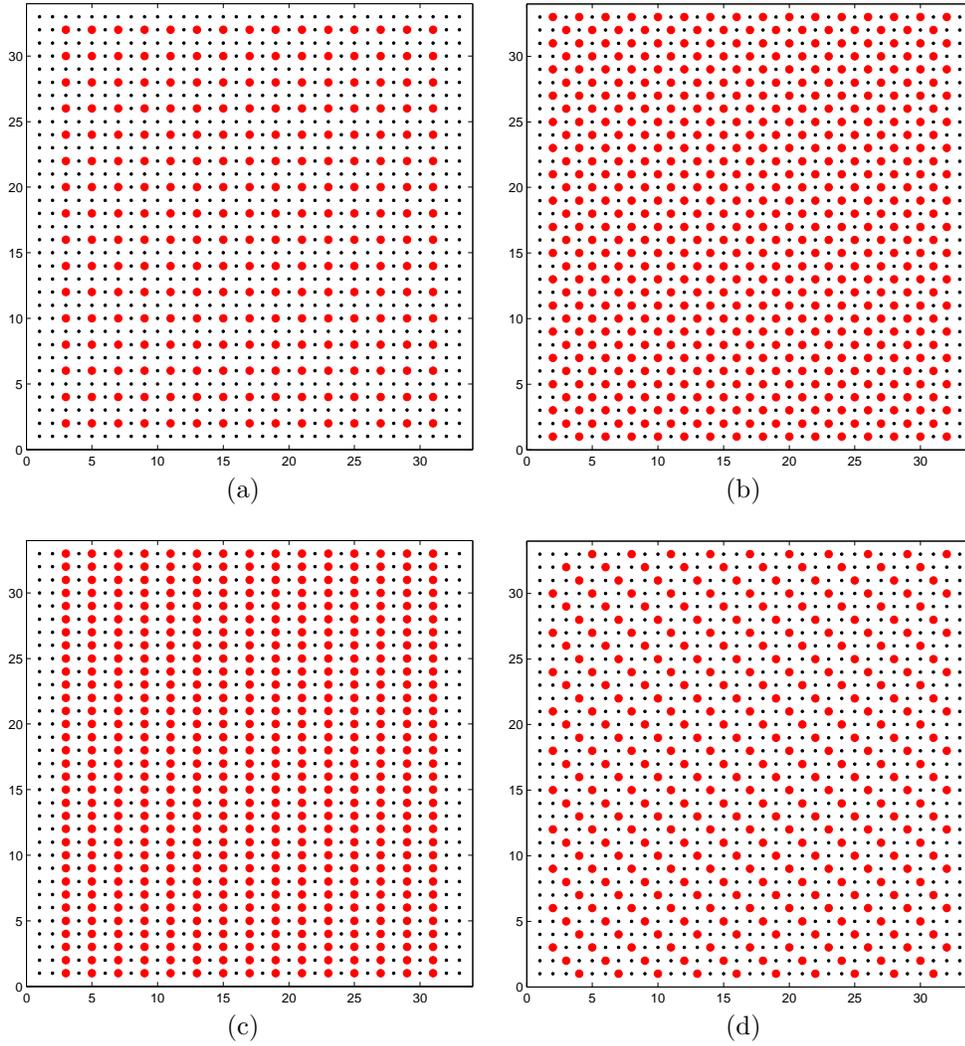
Problem	Grid	$\varrho_{cr}(0)$	$\varrho_{cr}(1)$	$\varrho_{cr}(2)$	Notes
2D-Lap-Q	(a)	0.98	0.51		
2D-Lap-T	(b)	0.99	0.00		
2D-ALap-Q	(c)	0.97	0.95	0.19	
	(a)	0.75	0.51		line ω-Jacobi ($\omega = 0.66$)
2D-ALap-T	(b)	0.98	0.00		
2D-RLap-Q	(a)	0.98	0.57		
	(c)	0.99	0.79	0.19	ω-Jacobi ($\omega = 0.75$), $\theta_{cs} = 0.75$
2D-RLap-T	(d)	0.98	0.33		

FIG. 4.2. *Coarse grids computed by Algorithm 3.1 for a variety of model problems. Each row of the table indicates the problem, the grid that was generated, the CR convergence factor initially $\varrho_{cr}(0)$ and at subsequent stages, along with notes describing changes from the default algorithm.*

Grid / Problem	2D-Lap-Q	2D-Lap-T	2D-ALap-Q	2D-ALap-T
33×33	.22/.51	.47/.00	.46/.19	.47/.00
65×65	.24/.52	.49/.00	.48/.19	.49/.00
129×129	.24/.52	.49/.00	.49/.19	.49/.00
257×257	.25/.52	.50/.00	.49/.19	.50/.00

Grid / Problem	2D-RLap-Q	2D-RLap-T	2D-4Reg-Q
33×33	.22/.57	.31/.33	.26/.49
65×65	.24/.58	.32/.32	.28/.51
129×129	.24/.58	.33/.34	-
257×257	.25/.58	.33/.33	-

TABLE 4.1
Computed values of the coarsening ratios $\delta = |\mathcal{G}|/|\mathcal{G}_c|$ and CR convergence factors ϱ_{cr} for various grid sizes.

To remedy this, a method such as symmetric Gauss-Seidel or weighted Jacobi can be used. However, with any smoother change, it may be necessary to modify the values of θ_{cr} or θ_{cs}. In Figure 4.2, we provide an example that uses weighted Jacobi with the optimal 2D smoothing parameter.

One unique aspect of our CR-based coarsening algorithm is that it complements the smoother used in the multigrid method. To see this, consider problem 2D-ALap-Q. From Figure 4.2, we see that with pointwise Gauss-Seidel, our algorithm semi-coarsens in x, as it should. However, if we simply change the smoother to line Jacobi (with optimal 1D weighting and lines in y), then the algorithm does full-coarsening. Using this combination of smoother and coarse grid is one of the oldest approaches for dealing with anisotropy. We are aware of no other AMG coarsening algorithms that are able to do this.

The results obtained by our method presented thus far have only been for a fixed problem size. The results reported in Table 4.1 indicate that our algorithm exhibits the same promising performance independently of problem size. Notice that both the values of the coarsening ratio δ and the CR-convergence factor ϱ_{cr} appear to be mesh-independent for all the problems considered so far.

4.2. Performance of a CR-based Multigrid Method. Choosing appropriate coarse grids is only one of the necessary components of an effective AMG setup scheme. Equally important is the definition of the sequence of interpolation operators, which is often intimately tied to the coarsening process. For example, the interpolation scheme used in the classical AMG algorithm [5, 3, 18] requires that each pair of strongly connected F-points, as defined by (3.1), must share a common interpolatory point. This property is explicitly enforced in the classical AMG coarsening algorithm. Our algorithm does not generate grids with this property, so classical AMG interpolation does not work well in combination with it. As mentioned earlier, our algorithm tends to coarsen more aggressively than other methods. Also, it may require significantly more stages on coarser grids if there is too much stencil growth. In this section, we describe an interpolation scheme that helps to mitigate these issues. We also apply our CR-based AMG algorithm to a number of 2D and 3D problems, and compare its performance to the BoomerAMG solver in the *hypre* library [16, 13].

One approach that has worked well for ameliorating stencil growth in the Galerkin

coarse-grid operator $A_c = P^T A P$ is to reduce the number of nonzeroes in the interpolation matrix P [20, 19]. To do this, we need a method for choosing a small but effective set of interpolatory points $\mathcal{C}_i$ for each point i on the fine grid. Most methods determine $\mathcal{C}_i$ using a strength-of-connection notion such as (3.1). Although our approach may also be interpreted in this way, it is more closely related to the idea of *interpolation truncation* [21, 19].

Recall from the theory in Section 2 that bounding η and $K_\star$ independently of parameters implies uniform two-grid convergence. Our CR-based coarsening algorithm ensures that $K_\star$ is bounded, implying a good coarse grid. For interpolation P, it is easy to show that the ideal interpolation operator $P_\star$ in (2.8) yields the minimum value of $\eta = 1$. In the classical AMG setting that we are considering in this paper, the operators R, S, and P are as in (2.10). For ideal interpolation, $W = W_\star := -A_{ff}^{-1} A_{fc}$, where A_{ff} and A_{fc} are as in (3.2). Of course, $P_\star$ is not a practical choice for interpolation because $W_\star$ is typically not sparse. Instead, we construct a local approximation to $W_\star$, then use it to define an effective interpolatory set $\mathcal{C}_i$. We then use the interpolation formula in [19] to build the coefficients of P.

To approximate $W_\star$, consider applying an iterative method to the linear system $A_{ff} W = -A_{fc}$ using an initial guess of zero. If W_a is our approximation, then the error $(W_\star - W_a)$ satisfies

$$(4.3) \qquad\qquad (W_\star - W_a) = E_f W_\star,$$

where E_f is the error propagator for the method and $W_\star$ is the initial error. Ultimately, the intention is for the iterative method to be related to compatible relaxation, but in general it can be any number of methods. For example, if we use ℓ steps of the primary CR method in (2.9), then $E_f = (I - M_{ff}^{-1} A_{ff})^\ell$. If we use conjugate gradient or a Chebyshev iteration, then E_f is some polynomial in A_{ff}. Regardless of the method, we can bound the size of the ij coefficient of the error using (4.3) as follows, where $\mathbf{e}_i$ and $\mathbf{e}_j$ are standard canonical Euclidian basis vectors and $\kappa(\cdot)$ denotes condition number:

$$
\begin{aligned}
|\,[W_\star - W_a]_{ij}\,| &= |\langle (W_\star - W_a)\,\mathbf{e}_i,\ \mathbf{e}_j\rangle| \\
&\leq \|W_\star - W_a\|\,\|\mathbf{e}_i\|\,\|\mathbf{e}_j\| \\
&= \|E_f A_{ff}^{-1} A_{fc}\| \\
&= \|(A_{ff}^{-1/2})(A_{ff}^{1/2} E_f A_{ff}^{-1/2})(A_{ff}^{-1/2})(A_{fc})\| \\
&\leq \|E_f\|_{A_{ff}}\,\|A_{ff}^{-1}\|\,\|A\| \\
&= \|E_f\|_{A_{ff}}\,\kappa(A_{ff})\,\frac{\|A\|}{\|A_{ff}\|}.
\end{aligned}
$$

If we use ℓ steps of the primary CR method in (2.9) to approximate $W_\star$, then we have

$$(4.4) \qquad\qquad |\,[W_\star - W_a]_{ij}\,| \leq \varrho_s^{\,\ell}\,\kappa(A_{ff})\,\frac{\|A\|}{\|A_{ff}\|},$$

where ϱ_s is given in (2.12) and measures the convergence factor for CR. If ϱ_s is small and M_{ff} is well-conditioned, it is easy to show that A_{ff} is well conditioned (see [8] for details). Hence, (4.4) shows that we can compute a fairly accurate approximation to $W_\star$ with only a few iterations of the CR method.

The bound in (4.4) also implies an exponential decay result similar to the one proven by Demko et al. in [11] (see also Section A.2.4 in [22]). To see this, notice

that for any ij such that $[W_a]_{ij} = 0$, we have $[W_\star - W_a]_{ij} = [W_\star]_{ij}$, so that (4.4) yields a bound for $|[W_\star]_{ij}|$. Also notice that the nonzero structure of W_a grows with the number of iterations ℓ as dictated by the specific iterative method used. In particular, define the distance function $d(i,j)$ as the largest ℓ such that $[W_a^{(k)}]_{ij} = 0$ for all $0 \leq k \leq \ell$, where $W_a^{(k)}$ is the k-th iterate of the method. Then (4.4) implies

$$(4.5) \qquad\qquad |[W_\star]_{ij}| \lesssim \varrho_s{}^{d(i,j)}.$$

If we have $\|E_f\|_{A_{ff}} \leq \varrho^\ell$ for some $\varrho < 1$, then the same basic result holds (replacing ϱ_s with ϱ) regardless of the iterative method used to generate W_a. For example, using either CG or a Chebyshev polynomial method, we can set $\varrho = 2(\sqrt{\kappa} - 1)/(\sqrt{\kappa} + 1)$ where $\kappa = \kappa(A_{ff})$. Also, since E_f is a polynomial in A_{ff} for both of these methods, then $d(i,j)$ is the usual graph distance function. With $A_{fc} = I$, this latter result is essentially the one derived in [11, 22].

The decay result in (4.5) provides some confidence that, if CR is fast to converge, then a local and sparse P with good approximation properties can be defined. Using ℓ iterations of the primary CR method, W_a is given by

$$(4.6) \qquad\qquad W_a = -\left(\sum_{k=0}^{\ell-1}(I - M_{ff}^{-1}A_{ff})^k M_{ff}^{-1}\right) A_{fc}.$$

So far in this paper, we have used lexicographical Gauss-Seidel for our CR method. Since this does not produce a local operator in (4.6), we instead use ℓ steps of the weighted Jacobi CR method (with initial guess $W_a^{(0)} = 0$)

$$(4.7) \qquad W_a^{(k)} = W_a^{(k-1)} + \omega D_{ff}^{-1}(A_{fc} - A_{ff}W_a^{(k-1)}), \quad k = 1, \ldots, \ell.$$

We choose the weight ω based on a Gershgorin bound on the largest eigenvalue of $D_{ff}^{-1}A_{ff}$ so that

$$(4.8) \qquad\qquad \omega^{-1} = \max_i \sum_j |[D_{ff}^{-1}A_{ff}]_{ij}|.$$

For some prescribed parameter $maxp$, set all but the largest $maxp$ entries in each row of W_a to zero. With this new truncated W_a and a second parameter $0 \leq \theta_a \leq 1$, we define the set of interpolation points for each point $i \in \mathcal{F}$ by

$$(4.9) \qquad \mathcal{C}_i = \{j \in \mathcal{C} : |[W_a]_{f(i),c(j)}| > \theta_a \max_k |[W_a]_{f(i),k}|\},$$

where $f(i)$ denotes the row of W_a associated with point $i \in \mathcal{F}$ and $c(j)$ denotes the column of W_a associated with point $j \in \mathcal{C}$.

Given this set of interpolatory points, we can define the coefficients of W in any number of ways. For the experiments in this paper, we use the *extended+i* method described in [19]. Specifically, we define the weights w_{ij} of W by

$$(4.10) \qquad w_{ij} = -\frac{\left(a_{ij} + \sum_{k \in \mathcal{N}_i \setminus \mathcal{C}_i}\left(\frac{a_{kj}}{\sum_{l \in \mathcal{C}_i \cup \{i\}} a_{kl}}\right)a_{ik}\right)}{\left(a_{ii} + \sum_{k \in \mathcal{N}_i \setminus \mathcal{C}_i}\left(\frac{a_{ki}}{\sum_{l \in \mathcal{C}_i \cup \{i\}} a_{kl}}\right)a_{ik}\right)}, \quad i \in \mathcal{F}, \ j \in \mathcal{C}_i,$$

where $\mathcal{N}_i$ is the set of neighbors of point i defined in (3.7). In keeping with our attempts to avoid defining strength of connection, we treat all neighbor points in
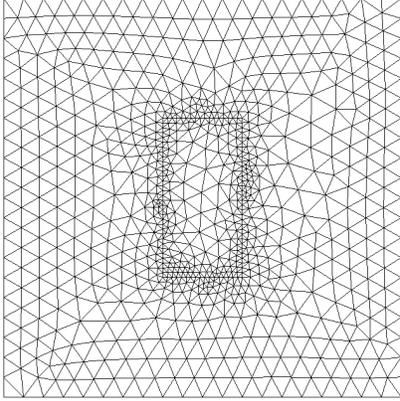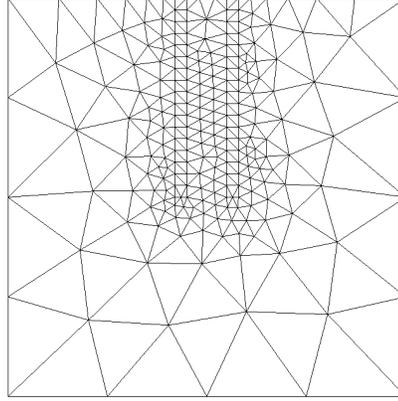
(M2)   (M3)

FIG. 4.3. *Two-dimensional unstructured meshes.*

(4.10) as simply interpolatory points (in $\mathcal{C}_i$) or non-interpolatory points. Alternative approaches for defining the interpolation weights can be found in [6], including an energy-minimization method (see also [8]) and an approach based on truncating and scaling the approximation W_a above.

For our numerical experiments, we augment the 2D test problems defined earlier with the following 2D and 3D problems based again on (4.1) and (4.2).

2D-M2-Lap (2D M2 Laplacian)
  $\Omega = [0,1] \times [0,1]$; Mesh M2 in Figure 4.3; $\epsilon = 1$, $d = 0$; BC's = Dirichlet;

2D-M3-Lap (2D M3 Laplacian)
  $\Omega = [0,1] \times [0,1]$; Mesh M3 in Figure 4.3; $\epsilon = 1$, $d = 0$; BC's = Dirichlet;

2D-M2-RLap (2D M2 Rotated Anisotropic Laplacian)
  $\Omega = [0,1] \times [0,1]$; Mesh M2 in Figure 4.3; $\epsilon = 0.01$, $\theta = \pi/3$, $d = 0$; BC's = Dirichlet;

2D-M3-RLap (2D M3 Rotated Anisotropic Laplacian)
  $\Omega = [0,1] \times [0,1]$; Mesh M3 in Figure 4.3; $\epsilon = 0.01$, $\theta = \pi/3$, $d = 0$; BC's = Dirichlet;

3D-Lap (3D Laplacian)
  $\Omega = [0,1] \times [0,1] \times [0,1]$; BC's = Dirichlet; $\mathbf{K} = 1$.

3D-ALap (3D Anisotropic Laplacian)
  $\Omega = [0,1] \times [0,1] \times [0,1]$; BC's = Dirichlet; $\mathbf{K} = \text{diag}(1, 0.1, 0.01)$.

Similarly to 2D, for 3D, we use a finite element discretization with bilinear basis functions on meshes composed of either hexahedra (denoted by 'H') or tetrahedra (denoted by 'T'). For example, the notation 3D-ALap-H indicates the 3D anisotropic Laplacian problem discretized with hexahedral elements.

15

We now compare the CR-based AMG method described above (CR-AMG) to the BoomerAMG solver in the *hypre* library [16, 13]. The purpose of this section is to demonstrate that a competitive CR-based AMG algorithm is feasible, and to illustrate the strengths and weaknesses of the current CR-AMG method. We solved homogeneous versions of the above problems to get accurate estimates of asymptotic convergence factors. We attempted to adjust the parameters in the two codes so that they differ only in the interpolation approach used.

For CR-AMG, we did at most 40 V(1,1)-cycles and stopped earlier if the norm of the error was less than 10^{-16}. We used: forward Gauss-Seidel smoothing on the down cycle; backward GS on the up cycle; $maxp = 4$ for 2D problems; $maxp = 6$ for 3D problems; $\theta_a = .25$ in (4.9); 200 iterations of GS on the coarsest grid.

For BoomerAMG, we used version 2.4.0b of the *hypre* library. We did at most 40 V(1,1)-cycles and stopped earlier if the norm of the residual was less than 10^{-16}. We used: C-F Gauss-Seidel smoothing on the down cyle; F-C GS on the up cycle; HMIS coarsening [20] with a strength threshold $\theta_s = 0.3$; extended+i interpolation [19]; $pmax = 4$ for 2D problems; $pmax = 6$ for 3D problems; interpolation truncation with factor 0.3; Gaussian elimination on the coarsest grid.

Table 4.2 reports asymptotic convergence factor estimates, number of multigrid levels, grid complexities, and operator complexities for a variety of test problems and sizes. Grid complexity is defined as the total number of grid points on all levels, divided by the number of grid points on the finest grid. Operator complexity is the total number of nonzeroes in the coefficient matrices on all levels, divided by the number of nonzeroes in A on the finest grid.

In general, CR-AMG uses fewer MG levels and has lower complexities than BoomerAMG, but this sometimes has a detrimental effect on convergence. This is particularly apparent for the rotated anisotropic Laplacian problems (the 'RLap' problems). The reason is in part because, with the default parameter settings, our CR coarsening algorithm tends to coarsen more aggressively than the algorithm in BoomerAMG. In addition, the CR method detects the fact that the condition number of the coarser-grid operators improves, and stops coarsening when it gets below the θ_{cr} threshold. Figure 4.4 illustrates both the smaller number of levels and the more aggressive coarsening for the 2D-M2-RLap problem.

For the isotropic problems (the 'Lap' problems), CR-AMG and BoomerAMG are comparable. However, for the anistropic problem 2D-ALap-Q, CR-AMG performs much better than BoomerAMG. This may be a consequence of the strength-of-connection measure in (3.1) not effectively identifying strong connections on coarser grids. Note that our choice of θ_s ensures that strong connections are identified correctly on the finest grid. For the 3D problems, CR-AMG is again comparable with BoomerAMG, except for problem 3D-ALap-T where CR-AMG achieves similar convergence factors with much smaller operator complexities and fewer grid levels.

An additional cost is incurred for setting up the current CR-AMG algorithm compared to BoomerAMG. The CR coarsening algorithm requires a few relaxations (2 or 3) for each stage. The number of stages is affected by the stencil size, so it is important to maintain small stencils on all levels. In CR-AMG, we achieve this primarily through the $maxp$ parameter. The number of stages for the results in Table 4.2 range from 0 to 2 for the 2D problems and from 0 to 4 for the 3D problems. Approaches for reducing the number of stages and/or reducing the cost of each stage is an area of current research. Another additional cost in the current CR-AMG set up algorithm is our approach for choosing the interpolatory set $\mathcal{C}_i$. For the results

Problem	Size	Asym CF	Num Levels	Grid Comp	Oper Comp
2D-Lap-Q	4225	.07 / .09	5 / 6	1.3 / 1.3	1.3 / 1.3
	16641	.08 / .11	6 / 7	1.3 / 1.3	1.3 / 1.3
	66049	.07 / .12	7 / 8	1.3 / 1.3	1.3 / 1.3
2D-Lap-T	4225	.08 / .05	6 / 7	1.6 / 1.7	2.2 / 2.2
	16641	.10 / .05	6 / 7	1.7 / 1.7	2.2 / 2.2
	66049	.14 / .07	7 / 8	1.7 / 1.7	2.2 / 2.2
2D-ALap-Q	4225	.11 / .75	5 / 8	1.8 / 1.9	1.8 / 2.0
	16641	.12 / .75	5 / 9	1.8 / 1.9	1.8 / 2.1
	66049	.14 / .75	6 /10	1.8 / 1.9	1.9 / 2.2
2D-ALap-T	4225	.08 / .09	4 / 8	1.7 / 1.9	2.1 / 2.7
	16641	.08 / .09	5 / 9	1.7 / 1.9	2.2 / 2.8
	66049	.19 / .09	6 /10	1.7 / 1.9	2.2 / 2.9
2D-RLap-Q	4225	.60 / .38	4 / 7	1.3 / 1.7	1.4 / 2.1
	16641	.75 / .39	5 / 8	1.3 / 1.7	1.4 / 2.2
	66049	.81 / .40	6 /10	1.3 / 1.7	1.4 / 2.2
2D-RLap-T	4225	.84 / .31	5 / 7	1.4 / 1.8	1.8 / 2.9
	16641	.88 / .35	6 / 8	1.4 / 1.8	1.8 / 3.0
	66049	.89 / .35	7 / 9	1.4 / 1.8	1.8 / 3.1
2D-4Reg	1089	.18 / .26	3 / 6	1.4 / 1.6	1.4 / 2.0
	4225	.67 / .63	4 / 8	1.4 / 1.7	1.5 / 2.1
2D-M2-Lap	798	.05 / .07	2 / 5	1.2 / 1.3	1.4 / 1.7
	3109	.12 / .11	4 / 6	1.3 / 1.4	1.6 / 1.9
	12273	.22 / .16	4 / 7	1.4 / 1.4	1.7 / 2.0
2D-M3-Lap	232	.03 / .08	2 / 4	1.2 / 1.2	1.3 / 1.4
	896	.06 / .13	3 / 5	1.3 / 1.4	1.6 / 1.7
	3523	.12 / .18	4 / 6	1.4 / 1.4	1.7 / 2.0
2D-M2-RLap	798	.58 / .39	2 / 6	1.2 / 1.6	1.4 / 2.1
	3109	.81 / .48	4 / 7	1.3 / 1.7	1.5 / 2.4
	12273	.95 / .57	4 / 8	1.4 / 1.8	1.6 / 2.6
2D-M3-RLap	232	.40 / .22	2 / 4	1.2 / 1.4	1.2 / 1.8
	896	.60 / .34	3 / 6	1.3 / 1.6	1.4 / 2.1
	3523	.80 / .44	3 / 7	1.3 / 1.7	1.5 / 2.3
3D-Lap-H	4913	.03 / .09	3 / 5	1.2 / 1.2	1.6 / 1.8
	35937	.03 / .23	4 / 7	1.2 / 1.3	1.7 / 2.1
3D-Lap-T	4913	.03 / .05	3 / 5	1.4 / 1.4	2.5 / 2.6
	35937	.03 / .05	4 / 6	1.5 / 1.5	2.7 / 2.9
	274625	.03 / .05	5 / 8	1.5 / 1.5	2.8 / 3.0
3D-ALap-H	4913	.40 / .25	2 / 6	1.2 / 1.5	1.5 / 1.8
	35937	.41 / .25	4 / 8	1.5 / 1.6	1.6 / 2.1
3D-ALap-T	4913	.04 / .11	3 / 7	1.5 / 1.6	2.2 / 2.8
	35937	.06 / .13	4 / 9	1.6 / 1.7	2.4 / 3.3
	274625	.17 / .13	5 /10	1.6 / 1.8	2.5 / 3.7

TABLE 4.2
Results for CR-AMG / BoomerAMG.
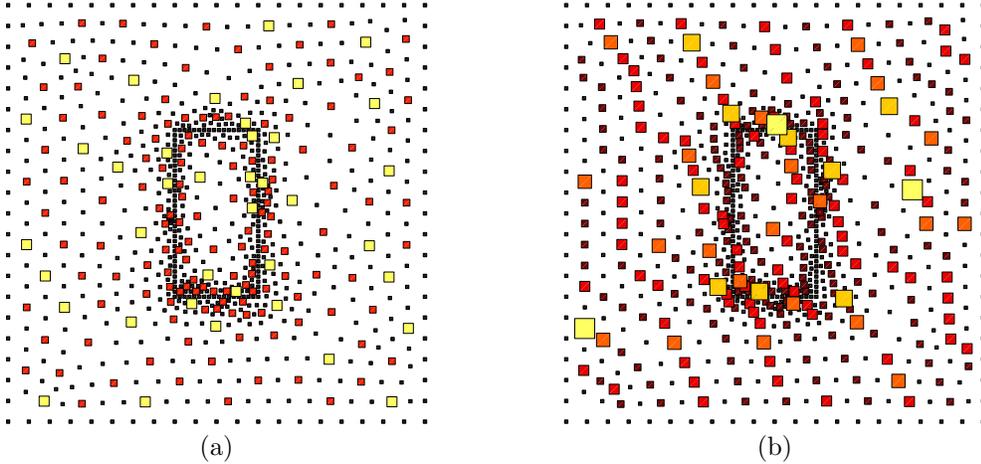
FIG. 4.4.   *Grid hierarchies produced by CR-AMG (a) and BoomerAMG (b) for the rotated anisotropic problem 2D-M2-RLap. Larger squares depict grid points on coarser-level grids.*

in the table, the number of iterations ℓ used in (4.7) was between 3 and 10. This is another area where a cheaper approach is desirable.

5. Improving the Predictive Properties of CR. In the CR-based coarsening algorithm above, the choice of stopping criteria θ_{cr} is based on empirical results. Considering the results in Table 4.1, it is obvious that although fast convergence of CR implies a coarse grid of good quality, exactly what is meant by "fast" convergence in this setting depends on the problem at hand. For these tests, our algorithm produces grids that are known to be effective in multigrid algorithms. However, the convergence factors of CR reported in this table vary significantly for each of the test problems considered. Our aim in this section is to develop new forms of CR that better predict MG convergence factors, and in turn give more accurate measurements of the quality of the coarse grid. To do this, we use a relationship between Theorem 2.1 and a sharp result in [15] to motivate more general choices of the coarse-grid variables, prescribed by the operator R. We then prove that our resulting CR methods serve also as measures of coarse-grid quality. In addition, by way of some simple examples, we demonstrate the potential of these methods to better predict two-grid multigrid convergence factors. The iteration we consider here is similar to one of the analysis tools discussed in [12].

For any SPD matrix X and any full-rank matrix B, denote the X-orthogonal projection onto range(B) by

$$(5.1) \qquad \pi_X(B) = B(B^T X B)^{-1} B^T X.$$

If $B = P$, we drop the parentheses so that $\pi_X := \pi_X(P)$. Let $\widetilde{M}$ and E_{TG} be given by (2.3) and (2.4), respectively. Then the following result, established in [15], gives the precise convergence factor of the two-grid method.

THEOREM 5.1.

$$(5.2) \qquad \|E_{TG}\|_A^2 = 1 - \frac{1}{K_\sharp}; \quad K_\sharp = \sup_{\mathbf{e}} \frac{\left\|(I - \pi_{\widetilde{M}})\mathbf{e}\right\|_{\widetilde{M}}^2}{\|\mathbf{e}\|_A^2}.$$

18

Recall Theorem 2.1, and note that the only difference between the expressions for K in (2.5) and $K_\sharp$ in (5.2) is the form of the projection. In particular, we see that $K = K_\sharp$ (i.e., the two estimates are the same), if

$$(5.3) \qquad R = (P^T \widetilde{M} P)^{-1} P^T \widetilde{M}.$$

Since compatible relaxation is based entirely on the choice of R, this suggests the possibility of defining more predictive CR methods by using a "better" R. However, the choice of R in (5.3) is rather complicated and impractical. To find a simpler form for R, note that the multigrid operator E_{TG} is invariant to post-scaling of interpolation. That is, we can replace P by PX for any nonsingular matrix X, and E_{TG} is unchanged. In particular, we can equivalently use the interpolation operator $\bar{P} = P(RP)^{-1}$ in the theory, since $R\bar{P} = I_c$ as required. Now, if we compare the numerator for K in (2.5) to the numerator for $K_\sharp$ in (5.2), it is clear that

$$(5.4) \qquad R = P^T \widetilde{M},$$

which is much simpler than in (5.3). It also makes more intuitive sense that the coarse-grid variables are best defined as the transpose of interpolation (times the smoother).

Note that this "best" R is a function of P, conflicting with the fact that we want to use R (through CR) to build P. However, (5.4) and the accompanying theory suggest that maybe we can choose any R that resembles the transpose of an effective interpolation operator. One context where this idea fits somewhat naturally is in an adaptive AMG setting [9, 10], where intermediate interpolation operators are formed as part of the adaptive cycle. For the purposes of this paper, we just assume we have a means of finding such an R, and consider how we might use it in a CR method.

Recall the primary and habituated CR methods in (2.9) and (2.11). Because we are considering more complex forms of R and (hence) S, the primary method may not be easy to compute, depending on the difficulty of inverting $S^T M S$. Since the habituated method is generally the most computable form of CR, we consider that variant below. Here, however, the theory requires S to be normalized, and this complicates things somewhat.

Define ϱ_h similarly to ϱ_s in (2.12) to be the A_s-norm of the habituated CR method in (2.11), where $A_s = S^T A S$ and S is assumed to be normalized. Replacing S by its normalized form $S(S^T S)^{-1/2}$ and recalling the notation in (5.1), we have

$$(5.5) \qquad \varrho_h = \left\| (S^T S)^{-1} S^T (I - M^{-1} A) S \right\|_{A_s}$$

$$(5.6) \qquad = \sup_{\mathbf{v}_s} \frac{\left\| \pi(S)(I - M^{-1} A) S \mathbf{v}_s \right\|_A}{\|S \mathbf{v}_s\|_A}$$

$$(5.7) \qquad = \sup_{\mathbf{v}_s} \frac{\left\| (I - \pi(R^T))(I - M^{-1} A) S \mathbf{v}_s \right\|_A}{\|S \mathbf{v}_s\|_A}.$$

The idea is to define the CR method as in the numerator of either (5.6) or (5.7), where we approximate the action of $(S^T S)^{-1}$ and $(RR^T)^{-1}$ with simple operators C_s and C_r, respectively. In particular, we are interested in the case where these operators result by applying a simple iterative method to the appropriate set of equations (e.g., $(R^T R)\mathbf{u}_r = \mathbf{f}_r$) with zero initial guess. That is, consider two habituated CR methods based on the following iteration matrices:

$$(5.8) \qquad S C_s S^T (I - M^{-1} A);$$

$$(5.9) \qquad (I - R^T C_r R)(I - M^{-1} A).$$

A result analogous to Theorem 2.4 (see [14]) shows that ϱ_h can be used to estimate the quality of the set of coarse-grid variables and to predict the convergence factor of the two-grid method. We now prove that a similar statement holds for the methods in (5.8) and (5.9) by showing that ϱ_h is bounded by the norm of these operators.

Before we state the theorem, a few comments are in order. The desired result is trivial to prove for (5.8) because the method produces iterates in range(S) as does the original habituated method in (2.11). The method in (5.9) is more difficult to deal with because it produces iterates that have components in both range(S) and range(R^T). In particular, if $\mathbf{v} = S\mathbf{v}_s + R^T\mathbf{v}_r$, then

$$(I - R^T C_r R)\mathbf{v} = R^T(I - C_r(RR^T))\mathbf{v}_r + S\mathbf{v}_s.$$

The difficulty is to bound the first term. Note that the size of this term is related to how accurately C_r approximates $(RR^T)^{-1}$, reflected below by μ_r.

THEOREM 5.2. *Assume that*

$$\mu_s := \left\| I - C_s(S^T S) \right\|_{A_s} < 1,$$
$$\mu_r := \left\| I - C_r(RR^T) \right\|_{A_r} < 1,$$

where $A_s = S^T A S$ and $A_r = RAR^T$ (e.g., assume that C_s and C_r result from convergent iterations), and let

$$\varrho_{h_s} = \left\| SC_s S^T(I - M^{-1}A) \right\|_A,$$
$$\varrho_{h_r} = \left\| (I - R^T C_r R)(I - M^{-1}A) \right\|_A.$$

Then

(5.10) $$\varrho_h \leq (1 - \mu_s)^{-1} \varrho_{h_s},$$

(5.11) $$\varrho_h \leq \varrho_{h_r} + \mu_r \left(\frac{2\gamma_m}{1 - \gamma^2} \right),$$

where $\gamma_m, \gamma \in [0,1)$ measure angles between spaces as follows:

(5.12) $\left\langle AS\mathbf{v}_s,\ M^{-T}R^T\mathbf{v}_r \right\rangle^2 \leq \gamma_m^2 \left\langle AS\mathbf{v}_s,\ S\mathbf{v}_s \right\rangle \left\langle AM^{-T}R^T\mathbf{v}_r,\ M^{-T}R^T\mathbf{v}_r \right\rangle,$

(5.13) $\left\langle AS\mathbf{v}_s,\ R^T\mathbf{v}_r \right\rangle^2 \leq \gamma^2 \left\langle AS\mathbf{v}_s,\ S\mathbf{v}_s \right\rangle \left\langle AR^T\mathbf{v}_r,\ R^T\mathbf{v}_r \right\rangle,$

for all $\mathbf{v}_s, \mathbf{v}_r$.

Proof. We first prove (5.10). From the triangle inequality, we have that

$$\begin{aligned}
\left\| \pi(S)\mathbf{v} \right\|_A &\leq \left\| SC_s S^T \mathbf{v} \right\|_A + \left\| (\pi(S) - SC_s S^T)\mathbf{v} \right\|_A \\
&= \left\| SC_s S^T \mathbf{v} \right\|_A + \left\| (I - C_s(S^T S))(S^T S)^{-1}S^T\mathbf{v} \right\|_{A_s} \\
&\leq \left\| SC_s S^T \mathbf{v} \right\|_A + \mu_s \left\| \pi(S)\mathbf{v} \right\|_A.
\end{aligned}$$

Collecting terms and letting $\mathbf{v} = (I - M^{-1}A)S\mathbf{v}_s$, we arrive at

$$\left\| \pi(S)(I - M^{-1}A)S\mathbf{v}_s \right\|_A \leq (1 - \mu_s)^{-1} \varrho_{h_s} \left\| S\mathbf{v}_s \right\|_A.$$

The result follows from (5.6).

We now prove (5.11). Similar to above, the triangle inequality yields

$$\left\| (I - \pi(R^T))\mathbf{v} \right\|_A \leq \left\| (I - R^T C_r R)\mathbf{v} \right\|_A + \mu_r \left\| \pi(R^T)\mathbf{v} \right\|_A.$$

20

Letting $\mathbf{v} = (I - M^{-1}A)S\mathbf{v}_s$ and using the fact that $\pi(S)S = S$, from (5.7) we get

$$\varrho_h \leq \varrho_{h_r} + \mu_r \left\| \pi(R^T)(I - M^{-1}A)\pi(S) \right\|_A.$$

It remains to bound $\left\| \pi(R^T)(I - M^{-1}A)\pi(S) \right\|_A$. We use the following facts for any matrix B:

$$\|B\|_A = \left\| A^{1/2}BA^{-1/2} \right\|; \quad \|B\| = \sup_{\mathbf{v},\mathbf{w}} \frac{|\langle B\mathbf{v}, \mathbf{w} \rangle|}{\|\mathbf{v}\| \|\mathbf{w}\|}; \quad \|B\| = \left\| B^T \right\|.$$

This, together with (5.12) and the fact that $\pi(R^T)\pi(S) = 0$, implies

$$
\begin{aligned}
\left\| \pi(R^T)(I - M^{-1}A)\pi(S) \right\|_A &= \left\| A^{1/2}\pi(R^T)M^{-1}A\pi(S)A^{-1/2} \right\| \\
&= \sup_{\mathbf{v},\mathbf{w}} \frac{|\langle A\pi(S)A^{-1/2}\mathbf{v}, \ M^{-T}\pi(R^T)A^{1/2}\mathbf{w} \rangle|}{\|\mathbf{v}\| \|\mathbf{w}\|} \\
&\leq \gamma_m \left\| A^{1/2}\pi(S)A^{-1/2} \right\| \left\| A^{1/2}M^{-T}\pi(R^T)A^{1/2} \right\| \\
&\leq \gamma_m \|\pi(S)\|_A \left\| \pi(R^T)M^{-1}A \right\|_A \\
&\leq \gamma_m \|\pi(S)\|_A \left\| \pi(R^T) \right\|_A \left\| M^{-1}A \right\|_A.
\end{aligned}
$$

But (5.13) holds if and only if

$$\|\pi(S)\|_A = \left\| \pi(R^T) \right\|_A \leq \sqrt{\frac{1}{1-\gamma^2}}.$$

Using this, together with the fact that $\left\| M^{-1}A \right\|_A < 2$ for a convergent smoother, proves the result. $\square$

To obtain a CR iteration whose convergence factor gives a better prediction of the performance of the resulting two-grid algorithm, Livne and Brandt suggest using a CR iteration in which the coarse-grid variables are defined as "averages" of the fine-grid variables (see [17]). The approach they propose is analogous to the habituated form of CR given by iteration (5.9) with a special choice of R. Next, using simple examples, we demonstrate the potential of such an approach.

Assume that P has the form given by (2.10) and define

$$R = \left[\begin{array}{cc} W_r^T & I_c \end{array} \right], \quad S = \left[\begin{array}{c} I_f \\ -W_r^T \end{array} \right],$$

so that $RS = 0$. Recall $\pi_X(B)$ in (5.1), and define

(5.14) $$E_X(B) = (I - \pi_X(B))(I - M^{-1}A).$$

In the following numerical experiments, we consider the CR iteration in (5.9) and analyze its ability to predict two-grid MG convergence factors. Specifically, we measure the A-norm of the CR operator, $\left\| E(R^T) \right\|_A$, and compare it to the A-norm of the two-grid MG operator, $\|E_A(P)\|_A$. We also estimate the spectral radius of these two error propagators by computing

(5.15) $$\varrho(E_X(B)) = \frac{\|\mathbf{e}^{(\nu)}\|_A}{\|\mathbf{e}^{(\nu-1)}\|_A},$$

where $\mathbf{e}^{(\nu)}$ is the ν-th iterate of the corresponding method with random initial guess. To demonstrate the potential of this CR method, we assume that a good interpolation operator P is known. Then, guided by the result in (5.4), we let $R^T = P$.

We consider two test problems from Section 4: 2D-Lap-Q (Laplacian) and 2D-ALap-Q (anisotropic Laplacian). We use the standard interpolation operators for these problems, so R^T has the following stencils:

$$\frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}, \qquad \frac{1}{2} \begin{bmatrix} & 0 & \\ 1 & 2 & 1 \\ & 0 & \end{bmatrix}.$$

(problem 2D-Lap-Q)   (problem 2D-ALap-Q).

We use Gauss-Seidel as the smoother.

Before presenting the results of our numerical experiments, we state an identity for the A-norm of the difference of $\pi_X(B)$ and $\pi_A(B)$. In our setting where $R^T = P$, the result implies that if $\left\|\pi(R^T)\right\|_A$ is close to one, then the A-orthogonal projection $\pi_A(P)$ should be well approximated by the corresponding L_2-orthogonal projection $\pi(R^T)$, indicating that the MG operator $E_A(P)$ should be well approximated by the CR operator $E(R^T)$.

THEOREM 5.3. *(see [6] for details) Define $\pi_X(B)$ as in (5.1). Then*

$$\|\pi_X(B) - \pi_A(B)\|_A^2 = \|\pi_X(B)\|_A^2 - 1.$$

Table 5.1 compares A-norms of the CR and two-grid MG operators for our two test problems. Since the error propagation matrices $E_X(B)$ are nonsymmetric, we compute the A-norm estimates as follows:

$$\|E_X(B)\|_A = \lambda_{\max}(A^{-1} E_X(B)^T A E_X(B))^{1/2}.$$

Note that $\left\|\pi(R^T)\right\|_A \approx 1.1$ for all grid sizes for both problems. Thus, the above theorem suggests that $\|E_A(P)\|_A$ is well approximated by $\left\|E(R^T)\right\|_A$, and the results in the table confirm this.

Table 5.2 compares spectral radius estimates for the CR and two-grid MG operators. Here, we include results using a small number of iterations m of diagonally-scaled CG to approximate the action of $(RR^T)^{-1}$. The results in the table indicate that the spectral radius estimates of the CR operator $E(R^T)$ closely approximate those of the MG operator $E_A(P)$. We also see that a small m is more than sufficient. This is because RR^T is well conditioned, having a condition number of about 4 for problem 2D-Lap-Q and about 2 for 2D-ALap-Q, for all three problems sizes. However, we see that the spectral radius estimates do not provide accurate estimates of the A-norms in Table 5.1. Of course, since the operators are nonsymmetric, theory only guarantees that the A-norm is an upper bound for the spectral radius.

To avoid this nonsymmetry problem, a better approach in practice might be to estimate the spectral radius of the CR operator

$$(I - M^{-T}A)(I - R^T C_r R)(I - M^{-1}A).$$

Here we get 0.1713 with $C_r = (RR^T)^{-1}$ for problem 2D-Lap-Q on a 32×32 grid, while the A-norm of the symmetric two-grid MG method is 0.1707 (the square of $\|E_A(P)\|_A = .4131$ in Table 5.1). In addition, with C_r determined by m steps of GS, the spectral radius estimate for CR is 0.186 for $m = 1$ and 0.173 for $m = 3$. By comparison, the CR method used in Section 4 gives a prediction of 0.5.

Problem	Grid	$\left\|E(R^T)\right\|_A$	$\left\|E_A(P)\right\|_A$
2D-Lap-Q	16×16	.39	.39
	32×32	.42	.41
	64×64	.43	.43
2D-ALap-Q	16×16	.58	.57
	32×32	.61	.60
	64×64	.61	.61

TABLE 5.1
A-norms of the CR and two-grid MG operators for two problems.

Problem	ν	$\varrho(E(R^T))$			$\varrho(E_A(P))$
		$m = 1$	$m = 2$	$m = 5$	
2D-Lap-Q	3	.26	.26	.26	.23
	10	.26	.26	.27	.25
	15	.27	.28	.28	.27
2D-ALap-Q	3	.44	.45	.45	.43
	10	.46	.47	.48	.44
	15	.47	.48	.48	.47

TABLE 5.2
Estimates of the spectral radii of the CR and two-grid MG operators for two problems on a 32×32 grid. For $\varrho(E(R^T))$, m diagonally-scaled CG iterations are used to approximate the action of $(RR^T)^{-1}$. For $\varrho(E_A(P))$, a direct method is used to compute $(RAR^T)^{-1}$.

6. Conclusions and Future Research. We introduced a coarsening algorithm for algebraic multigrid (AMG) based on the concept of compatible relaxation (CR), studied its behavior on a number of model problems, presented the performance results of an AMG method that utilizes the coarsening approach, and demonstrated the potential of a more predictive variant of CR that provides a sharper metric of coarse-grid quality. Two unique aspects of the method are that it does not use strength of connection and it naturally complements the smoother used in the AMG solver.

We are exploring several research avenues with respect to CR. We would like to reduce the sensitivity of the algorithm to the parameters θ_{cr} and θ_{cs}. The most important of these is θ_{cr}, which can be reduced to improve the quality of the coarse grid. However, doing this sometimes results in too many coarse-grid points, and it usually increases the number of stages required. This latter effect is also an issue for parallel computing, our main topic of current research. The independent set algorithms needed in parallel tend to produce even more aggressively coarsened grids than those illustrated in this paper. Another issue in parallel is that we cannot use lexicographical Gauss-Seidel to generate the candidate set, because it does not parallelize well. Finally, we are also investigating the use of CR for problems that come from systems of PDEs. Here, the candidate set measure in (3.6) must be modified.

REFERENCES

[1] D. M. ALBER AND L. N. OLSON, *Parallel coarse-grid selection*, Numer. Linear Algebra Appl., 14 (2007), pp. 611–643. 1

[2] R. E. BANK, *Compatible coarsening in the multigraph algorithm*, Advances in Engineering Software, 38 (2007), pp. 287–294. 1

[3] A. BRANDT, *Algebraic multigrid theory: The symmetric case*, Appl. Math. Comput., 19 (1986), pp. 23–56. 1, 12

[4] ———, *General highly accurate algebraic coarsening*, Electronic Transactions on Numerical Analysis, 10 (2000), pp. 1–20. 1, 3

[5] A. BRANDT, S. F. MCCORMICK, AND J. W. RUGE, *Algebraic multigrid (AMG) for sparse matrix equations*, in Sparsity and Its Applications, D. J. Evans, ed., Cambridge University Press, Cambridge, 1984. 1, 12

[6] J. BRANNICK, *Adaptive algebraic multigrid coarsening strategies*, PhD thesis, University of Colorado at Boulder, Boulder, Colorado, USA, 2005. 1, 5, 7, 15, 22

[7] J. BRANNICK, M. BREZINA, S. MACLACHLAN, T. MANTEUFFEL, S. MCCORMICK, AND J. RUGE, *An energy-based AMG coarsening strategy*, Numer. Linear Algebra Appl., 13 (2006), pp. 133–148. 5

[8] J. BRANNICK AND L. ZIKATANOV, *Algebraic multigrid methods based on compatible relaxation and energy minimization*, in Domain Decomposition Methods in Science and Engineering XVI, O. B. Widlund and D. E. Keyes, eds., vol. 55 of Lecture Notes in Computational Science and Engineering, Springer-Verlag, 2007, pp. 15–26. 1, 13, 15

[9] M. BREZINA, R. FALGOUT, S. MACLACHLAN, T. MANTEUFFEL, S. MCCORMICK, AND J. RUGE, *Adaptive smoothed aggregation (αSA)*, SIAM J. Sci. Comput., 25 (2004), pp. 1896–1920. UCRL-JC-151149. 19

[10] ———, *Adaptive algebraic multigrid*, SIAM J. Sci. Comput., 27 (2006), pp. 1261–1286. UCRL-JRNL-203501. 19

[11] S. DEMKO, W. MOSS, AND P. SMITH, *Decay rates of inverse band matrices*, Math. Comp., 43 (1984), pp. 491–499. 13, 14

[12] B. DISKIN, J. THOMAS, AND R. MINECK, *On quantitative analysis methods for multigrid solutions*, SIAM J. Sci. Comput., 27 (2005), pp. 108–129. 18

[13] R. D. FALGOUT, J. E. JONES, AND U. M. YANG, *The design and implementation of hypre, a library of parallel high performance preconditioners*, in Numerical Solution of Partial Differential Equations on Parallel Computers, A. M. Bruaset and A. Tveito, eds., vol. 51 of Lecture Notes in Computational Science and Engineering, Springer-Verlag, 2006, ch. 8, pp. 267–294. UCRL-JRNL-205459. 12, 16

[14] R. D. FALGOUT AND P. S. VASSILEVSKI, *On generalizing the AMG framework*, SIAM J. Numer. Anal., 42 (2004), pp. 1669–1693. UCRL-JC-150807. 1, 2, 3, 4, 5, 20

[15] R. D. FALGOUT, P. S. VASSILEVSKI, AND L. T. ZIKATANOV, *On two-grid convergence estimates*, Numer. Linear Algebra Appl., 12 (2005), pp. 471–494. UCRL-JRNL-203843. 1, 18

[16] *hypre: High performance preconditioners.* http://www.llnl.gov/CASC/hypre/. 12, 16

[17] O. LIVNE, *Coarsening by compatible relaxation*, Numer. Linear Algebra Appl., 11 (2004), pp. 205–227. 1, 4, 21

[18] J. W. RUGE AND K. STÜBEN, *Algebraic multigrid (AMG)*, in Multigrid Methods, S. F. McCormick, ed., vol. 3 of Frontiers in Applied Mathematics, SIAM, Philadelphia, PA, 1987, pp. 73–130. 1, 12

[19] H. D. STERCK, R. D. FALGOUT, J. W. NOLTING, AND U. M. YANG, *Distance-two interpolation for parallel algebraic multigrid*, Numer. Linear Algebra Appl., 15 (2008), pp. 115–139. Special issue on Multigrid Methods. UCRL-JRNL-230844. 13, 14, 16

[20] H. D. STERCK, U. M. YANG, AND J. J. HEYS, *Reducing complexity in parallel algebraic multigrid preconditioners*, SIAM J. Matrix Anal. Appl., 27 (2006), pp. 1019–1039. 13, 16

[21] K. STÜBEN, *An introduction to algebraic multigrid*, in Multigrid, U. Trottenberg, C. Oosterlee, and A. Schüller, eds., Academic Press, San Diego, CA, 2001, pp. 413–528. 13

[22] P. S. VASSILEVSKI, *Multilevel Block Factorization Preconditioners: Matrix-based Analysis and Algorithms for Solving Finite Element Equations*, Springer, New York, 2008. 13, 14