NUMERICAL LINEAR ALGEBRA WITH APPLICATIONS
Numer. Linear Algebra Appl. 0000; **00**:1–25
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nla

Robust and Adaptive Multigrid Methods: comparing structured and algebraic approaches

S.P. MacLachlan[1,*], J.D. Moulton[2], and T.P. Chartier[3]

[1] *Department of Mathematics, Tufts University, 503 Boston Avenue, Medford, MA 02155, USA.*
[2] *Applied Mathematics and Plasma Physics, MS B284, Los Alamos National Laboratory, Los Alamos, NM 87545, USA*
[3] *Department of Mathematics, Davidson College, Box 6908, Davidson, NC 28035, USA.*

SUMMARY

Although there have been significant advances in robust algebraic multigrid methods in recent years, numerical studies and emerging hardware architectures continue to favor structured-grid approaches. Specifically, implementations of logically structured robust variational multigrid algorithms, such as the Black Box Multigrid (BoxMG) solver, have been shown to be 10 times faster than AMG for three-dimensional heterogeneous diffusion problems on structured grids [1]. BoxMG offers important features such as operator-induced interpolation for robustness, while taking advantage of direct data access and bounded complexity in the Galerkin coarse-grid operator. Moreover, since BoxMG uses a variational framework, it can be used to explore advances of modern adaptive AMG approaches in a structured setting. In this paper, we show how to extend the adaptive multigrid methodology to the BoxMG setting. This extension not only retains the favorable properties of the adaptive framework, but also sheds light on the relationship between BoxMG and AMG. In particular, we show how classical BoxMG can be viewed as a *special case* of classical AMG, and how this viewpoint leads to a richer family of adaptive BoxMG approaches. We present numerical results that explore this family of adaptive methods and compare its robustness and efficiency to the classical BoxMG solver.

Received ...

KEY WORDS: Multigrid, Adaptive Multigrid, Algebraic Multigrid

1. INTRODUCTION

While multigrid methods have been actively studied since the seminal work of Brandt in the 1970's [2,3], the field remains one of active research, with recent advances in robustness from the adaptive multigrid framework [4–7] and application to a wide variety of new problems [8–12]. Additionally, the challenges of evolving computer architectures have driven research into numerical methods that are naturally suited to GPUs and other accelerated architectures [13–15]. In this paper, we consider the implementation of the adaptive multigrid methodology

*Correspondence to: S. P. MacLachlan, Department of Mathematics, Tufts University, 503 Boston Avenue, Medford, MA 02155, USA. scott.maclachlan@tufts.edu

Contract/grant sponsor: The work of SPM was partially supported by the National Science Foundation, under grant DMS-0811022. The work of JDM was funded by the Department of Energy at Los Alamos National Laboratory under contracts DE-AC52-06NA25396 and the DOE Office of Science Advanced Computing Research (ASCR) program in Applied Mathematical Sciences. The work of TPC was partially supported by a research fellowship from the Alfred P. Sloan Foundation and the Department of Energy, under grant DE-FG02-04ER25590.

for structured-grid robust multigrid approaches, such as the Black Box Multigrid algorithm (BoxMG) [16, 17]. We also note several connections between BoxMG and the Algebraic Multigrid algorithm (AMG) [18–20].

The Black Box Multigrid method (BoxMG), which was introduced in [17] (largely following from [16]) and further developed in [21–24], is a robust multigrid solver (or, more correctly, family of algorithms) that is known to be effective for the solution of various PDEs discretized on logically structured two- or three-dimensional grids. Much like AMG (and as its name implies), these algorithms are intended to function as "black boxes", with the user providing only the fine-grid discretization, right-hand side, and an initial guess for the solution. The key difference between BoxMG and AMG is that BoxMG uses a fixed coarse-grid structure, based on geometric coarsening, while the coarse grids in AMG are typically computed based on either graph algorithms [18, 19, 25–27] or compatible relaxation principles [28–30]. Thus, BoxMG may be efficiently implemented using structured data representations allowing direct addressing, while AMG typically relies on unstructured data storage and indirect addressing. As with AMG, interest in simulating challenging applications, such as electrical activation in the heart [31], models of tumor growth [32], geophysical electromagnetic simulation [33], and flow in heterogeneous porous media [34–36], has driven the algorithm's use and continued development.

While much of the development of robust multigrid approaches in the last decade has been focused on algebraic multigrid techniques and their parallelization [25–27,37–41], recent trends in computer architectures, particularly towards many-core and accelerated architectures that achieve their best performance operating on structured data, suggest that it is worthwhile to revisit structured robust algorithms like BoxMG. On these architectures, the performance hit taken by unstructured data accesses and indirect addressing may be so significant that it is more efficient to consider larger, logically structured grids with regular data access patterns than optimal-accuracy unstructured refinements that necessitate AMG. For example, logically structured body-fitted grids may be used to handle complex shapes, and irregular domains may be embedded in rectangular domains.

A major goal of this paper is to extend the recently developed adaptive multigrid framework [4–7] to the structured BoxMG setting. While classical AMG significantly expanded the scope of problems to which multigrid can be applied, important classes of problems, such as those of quantum chromodynamics [42–44], still pose difficulties for classical AMG. For such problems, the implicit assumptions made by AMG regarding the errors that pointwise Gauss-Seidel relaxation is slow to reduce are not satisfied. Adaptive multigrid methods, then, are designed to avoid making such fixed assumptions by dynamically assessing the nature of these slow-to-converge errors in order to construct effective multigrid components. Similarly, the construction of the interpolation operators in classical BoxMG is based on assumptions regarding the slow-to-converge errors in the BoxMG relaxation, either pointwise or linewise Gauss-Seidel, and the adaptive methodology offers similar improvements to the robustness of the BoxMG algorithm. Two important consequences of this extension are the use of the adaptive framework with a linewise Gauss-Seidel smoother, allowing better treatment of anisotropic problems, and the ability to remove some of the logical complexities in the classical BoxMG interpolation scheme, which allow for efficient treatment of a wide variety of boundary conditions (see Section 4).

A natural consequence of the extension of adaptive multigrid methods (which have, thus far, been developed only for algebraic multigrid algorithms) to the BoxMG framework is a new perspective on the relationship between the classical and adaptive AMG and BoxMG interpolation formulae. This relationship is also explored here. In particular, we see that the BoxMG interpolation formulae can be viewed as special cases of the AMG formulae, based on non-standard definitions of strong and weak connections. This perspective, then, suggests a family of logically structured robust multigrid algorithms, based on BoxMG, AMG, or combinations of the two. These approaches are also explored.

The remainder of this paper is organized as follows. Section 2.1 reviews the classical BoxMG interpolation formulae, while the adaptive variants of these formulae are introduced in Section

DOI: 10.1002/nla

2.2. The adaptive multigrid setup cycling scheme is reviewed in Section 2.3. The connections between BoxMG and AMG interpolation are discussed in Section 3. Numerical results are presented in Section 4, while Section 5 gives conclusions.

2. STRUCTURED AND ADAPTIVE INTERPOLATION APPROACHES

The structured-grid interpolation used in BoxMG [17] is based on the assumptions that both the fine and coarse grids are *logically rectangular*; that is, that the fine-grid matrix has either a 5-point or 9-point stencil that matches the connectivity pattern of a finite-difference or finite-element discretization on a tensor-product Cartesian mesh. While there are important practical differences in implementation if the fine-grid stencil is a 5-point (finite-difference) stencil, there is no change in the formulae for the resulting interpolation operators, aside from the use of zero coefficients for the missing entries in the matrix. Thus, we focus here on the 9-point connectivity case that is typical of quadrilateral finite-element discretizations. We also focus here on standard geometric multigrid coarsening of these meshes, by a factor of two in each direction. While coarsening by higher ratios is also possible [45,46], the standard coarsening by factors of 2^d closely matches the classical AMG framework [18,20] to which we will compare.

Historically, BoxMG predates AMG, with the concept of operator-induced interpolation originally introduced in [16,17]. Thus, complications in the multigrid treatment of the PDE, including jumps in diffusion coefficients or unequal grid spacings, are automatically included in the definition of the interpolation operator. Physically, for a standard diffusion problem, the BoxMG interpolation operator can be seen to be based on the approximation of continuity of normal fluxes across jumps in the diffusion coefficient that are aligned with the mesh [36]; in Section 3, we show that this approach can also be equated to a standard AMG definition of interpolation for a fixed choice of the coarse grid and non-standard choice of the definition of "strong connections" in the matrix.

In the discussion that follows, we make use of a compass-based notation for connections in the matrix, A, as given in Figure 1. In this notation, the row of the matrix equation $A\mathbf{x} = \mathbf{b}$ associated with node (i,j) is given by

$$\begin{aligned} a_{i,j}^{SW} x_{i-1,j-1} + a_{i,j}^{S} x_{i,j-1} + a_{i,j}^{SE} x_{i+1,j-1} + a_{i,j}^{W} x_{i-1,j} + a_{i,j}^{O} x_{i,j} \\ + a_{i,j}^{E} x_{i+1,j} + a_{i,j}^{NW} x_{i-1,j+1} + a_{i,j}^{N} x_{i,j+1} + a_{i,j}^{NE} x_{i+1,j+1} = b_{i,j}. \end{aligned}$$

Within BoxMG, just as within AMG, the goal of the definition of interpolation is to faithfully represent this equation for an error that yields a small residual, typically assumed to be zero:

$$\begin{aligned} a_{i,j}^{SW} e_{i-1,j-1} + a_{i,j}^{S} e_{i,j-1} + a_{i,j}^{SE} e_{i+1,j-1} + a_{i,j}^{W} e_{i-1,j} + a_{i,j}^{O} e_{i,j} \\ + a_{i,j}^{E} e_{i+1,j} + a_{i,j}^{NW} e_{i-1,j+1} + a_{i,j}^{N} e_{i,j+1} + a_{i,j}^{NE} e_{i+1,j+1} = 0, \end{aligned} \tag{1}$$

although the inclusion of the residual in an affine interpolation process was noted to be important in certain cases in [17].

2.1. Classical BoxMG Interpolation

The classical BoxMG interpolation operator involves four cases:

1. (i,j) **is a coarse point:** In this case, shown at the left of Figure 2, interpolation simply preserves the coarse-grid value: $\left(P\mathbf{e}^C\right)_{i,j} = e^C_{I,J}$, where (I,J) is the coarse-grid index for the node corresponding to (i,j) on the fine grid.
2. $(i-1,j)$ **and** $(i+1,j)$ **are coarse points:** In this case, shown in the middle of Figure 2, point (i,j) lies embedded in a coarse-grid line in the x-direction, and the goal is to define a one-dimensional interpolation formula from coarse-grid points (I,J) and $(I+1,J)$, corresponding to fine-grid points $(i-1,j)$ and $(i+1,j)$, respectively, to (i,j).

DOI: 10.1002/nla

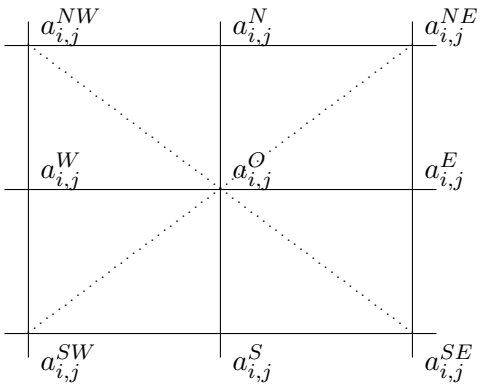


Figure 1. Compass-Based Notation for the Stencil at Node (i, j)

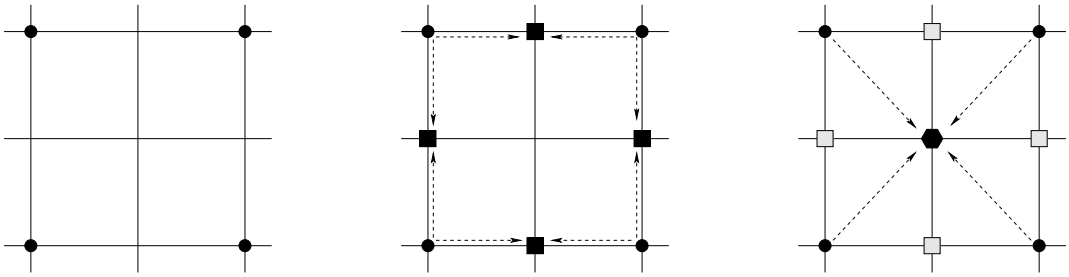


Figure 2. The schematic [45] shows injection of coarse-grid points (left) and interpolation of two different classes of fine-grid points: points embedded in coarse-grid lines (middle), and points located at the logical center of a coarse-grid cell (right).

The BoxMG approach is to simply collapse the stencil in the y-direction onto the line with fixed y-coordinate, j. Thus, in Equation (1), we assume that the error is approximately constant in the y direction and write

$$e_{k,j} \approx e_{k,j-1} \approx e_{k,j+1}, \text{ for } k = i-1, i, i+1, \tag{2}$$

giving

$$\left(a_{i,j}^{S} + a_{i,j}^{O} + a_{i,j}^{N}\right) e_{i,j} = -\left(a_{i,j}^{SW} + a_{i,j}^{W} + a_{i,j}^{NW}\right) e_{I,J}^{C} - \left(a_{i,j}^{SE} + a_{i,j}^{E} + a_{i,j}^{NE}\right) e_{I+1,J}^{C}, \tag{3}$$

where we have used the injected errors at the coarse-grid points, $e_{i-1,j} = e_{I,J}^{C}$ and $e_{i+1,j} = e_{I+1,J}^{C}$. Thus, the interpolation formula is given by

$$(P\mathbf{e}^{C})_{i,j} = -\frac{a_{i,j}^{SW} + a_{i,j}^{W} + a_{i,j}^{NW}}{a_{i,j}^{S} + a_{i,j}^{O} + a_{i,j}^{N}} e_{I,J}^{C} - \frac{a_{i,j}^{SE} + a_{i,j}^{E} + a_{i,j}^{NE}}{a_{i,j}^{S} + a_{i,j}^{O} + a_{i,j}^{N}} e_{I+1,J}^{C}. \tag{4}$$

Writing $\bar{a}_{i,j}^{Oy} = a_{i,j}^{S} + a_{i,j}^{O} + a_{i,j}^{N}$, $\bar{a}_{i,j}^{W} = a_{i,j}^{SW} + a_{i,j}^{W} + a_{i,j}^{NW}$, and $\bar{a}_{i,j}^{E} = a_{i,j}^{SE} + a_{i,j}^{E} + a_{i,j}^{NE}$, this can be compactly written as $(P\mathbf{e}^{C})_{i,j} = -\frac{\bar{a}_{i,j}^{W}}{\bar{a}_{i,j}^{Oy}} e_{I,J}^{C} - \frac{\bar{a}_{i,j}^{E}}{\bar{a}_{i,j}^{Oy}} e_{I+1,J}^{C}$.

3. $(i, j-1)$ **and** $(i, j+1)$ **are coarse points:** This case is analogous to the previous one, with the roles of the x- and y-coordinates switched. Thus, taking (I, J) and $(I, J+1)$ to be the coarse-grid indices corresponding to fine-grid points $(i, j-1)$ and $(i, j+1)$, respectively, we make the approximations that

$$e_{i,k} \approx e_{i-1,k} \approx e_{i+1,k}, \text{ for } k = j-1, j, j+1.$$

DOI: 10.1002/nla

Substituting these in Equation (1) and rearranging, as above, gives

$$(P\mathbf{e}^C)_{i,j} = -\frac{a_{i,j}^{SW} + a_{i,j}^{S} + a_{i,j}^{SE}}{a_{i,j}^{W} + a_{i,j}^{O} + a_{i,j}^{E}} e_{I,J}^C - \frac{a_{i,j}^{NW} + a_{i,j}^{N} + a_{i,j}^{NE}}{a_{i,j}^{W} + a_{i,j}^{O} + a_{i,j}^{E}} e_{I,J+1}^C. \tag{5}$$

4. $(i-1, j-1)$**,** $(i+1, j-1)$**,** $(i-1, j+1)$**, and** $(i+1, j+1)$ **are coarse points:** In this case, shown at the right of Figure 2, the fine-grid point (i, j) lies at the center of a coarse-grid cell with nodes (I, J), $(I+1, J)$, $(I, J+1)$, and $(I+1, J+1)$ corresponding to fine-grid nodes $(i-1, j-1)$, $(i+1, j-1)$, $(i-1, j+1)$, and $(i+1, j+1)$, respectively. In this case, there is no clear preferential geometric direction for collapsing the stencil, since each connection to the fine-grid neighbors of (i, j) is equidistant (both geometrically and grid-wise) from two coarse-grid points and (i, j) itself. However, as interpolation has already been determined for each of these neighbors, we can leverage the definitions of interpolation given above to eliminate them from Equation (1), writing

$$\begin{aligned} a_{i,j}^{SW} e_{i-1,j-1} + a_{i,j}^{S} \left(P\mathbf{e}^C\right)_{i,j-1} + a_{i,j}^{SE} e_{i+1,j-1} + a_{i,j}^{W} \left(P\mathbf{e}^C\right)_{i-1,j} + a_{i,j}^{O} e_{i,j} \\ + a_{i,j}^{E} \left(P\mathbf{e}^C\right)_{i+1,j} + a_{i,j}^{NW} e_{i-1,j+1} + a_{i,j}^{N} \left(P\mathbf{e}^C\right)_{i,j+1} + a_{i,j}^{NE} e_{i+1,j+1} = 0, \end{aligned}$$

or

$$\begin{aligned} a_{i,j}^{O} e_{i,j} = -a_{i,j}^{SW} e_{I,J}^C - a_{i,j}^{S} \left(P\mathbf{e}^C\right)_{i,j-1} - a_{i,j}^{SE} e_{I+1,J}^C - a_{i,j}^{W} \left(P\mathbf{e}^C\right)_{i-1,j} \\ - a_{i,j}^{E} \left(P\mathbf{e}^C\right)_{i+1,j} - a_{i,j}^{NW} e_{i,J+1}^C - a_{i,j}^{N} \left(P\mathbf{e}^C\right)_{i,j+1} - a_{i,j}^{NE} e_{I+1,I+1}^C. \end{aligned} \tag{6}$$

Denoting the coefficient of P interpolating from coarse-grid node (K, L) to fine-grid node (k, ℓ) by $p_{(k,\ell),(K,L)}$ (in a slight modification of the usual row-column indexing notation), we can then expand the entries of $P\mathbf{e}^C$ that appear above as

$$\begin{aligned} \left(P\mathbf{e}^C\right)_{i,j-1} &= p_{(i,j-1),(I,J)} e_{I,J}^C + p_{(i,j-1),(I+1,J)} e_{I+1,J}^C, \\ \left(P\mathbf{e}^C\right)_{i,j+1} &= p_{(i,j+1),(I,J+1)} e_{I,J+1}^C + p_{(i,j+1),(I+1,J+1)} e_{I+1,J+1}^C, \\ \left(P\mathbf{e}^C\right)_{i-1,j} &= p_{(i-1,j),(I,J)} e_{I,J}^C + p_{(i-1,j),(I,J+1)} e_{I,J+1}^C, \\ \text{and } \left(P\mathbf{e}^C\right)_{i+1,j} &= p_{(i+1,j),(I+1,J)} e_{I+1,J}^C + p_{(i+1,j),(I+1,J+1)} e_{I+1,J+1}^C. \end{aligned}$$

Making these substitutions into Equation (6) implicitly gives the interpolation formula

$$\begin{aligned} \left(P\mathbf{e}^C\right)_{i,j} = &- \frac{a_{i,j}^{SW} + a_{i,j}^{S} p_{(i,j-1),(I,J)} + a_{i,j}^{W} p_{(i-1,j),(I,J)}}{a_{i,j}^{O}} e_{I,J}^C \\ &- \frac{a_{i,j}^{SE} + a_{i,j}^{S} p_{(i,j-1),(I+1,J)} + a_{i,j}^{E} p_{(i+1,j),(I+1,J)}}{a_{i,j}^{O}} e_{I+1,J}^C \\ &- \frac{a_{i,j}^{NW} + a_{i,j}^{N} p_{(i,j+1),(I,J+1)} + a_{i,j}^{W} p_{(i-1,j),(I,J+1)}}{a_{i,j}^{O}} e_{I,J+1}^C \\ &- \frac{a_{i,j}^{NE} + a_{i,j}^{N} p_{(i,j+1),(I+1,J+1)} + a_{i,j}^{E} p_{(i+1,j),(I+1,J+1)}}{a_{i,j}^{O}} e_{I+1,J+1}^C. \end{aligned} \tag{7}$$

To achieve added robustness, for problems on grids of dimensions other than $2^N + 1$, that include lower-order terms, or that involve a wide variety of boundary conditions, the BoxMG algorithm supplements these definitions with a heuristic switch for diagonally dominant rows of the matrix [45]. Thus, each of the definitions in Equations (4), (5), and (7) has a generalized form that makes use of this switch. For example, at fine-grid points embedded in horizontal coarse-grid lines, Equation (4) becomes

$$(P\mathbf{e}^C)_{i,j} = -\frac{\bar{a}_{i,j}^{W}}{\mathcal{A}_{i,j}^{Oy}} e_{I,J}^C - \frac{\bar{a}_{i,j}^{E}}{\mathcal{A}_{i,j}^{Oy}} e_{I+1,J}^C.$$

Prepared using nlaauth.cls
DOI: 10.1002/nla

Taking $\mathcal{A}_{i,j}^{Oy} = \bar{a}_{i,j}^{Oy}$, as in Equation (4) above, is referred to as *averaging* interpolation. Although this is an effective option for many situations, there are also situations where this leads to poor performance, particularly for diagonally dominant rows of the matrix. An alternate approach, defining $\mathcal{A}_{i,j}^{Oy} = \bar{a}_{i,j}^{W} + \bar{a}_{i,j}^{E}$, gives interpolation coefficients that exactly preserve the constant in interpolation, even when the stencil is diagonally dominant. This formulation is referred to as *constant-preserving* interpolation; although this approach is effective in many situations, it is not always successful. Most notably, *constant-preserving* interpolation fails when the grids are not of dimension $2^N + 1$ in each direction, as extrapolation is used at the right and/or top boundaries.

The solution implemented in BoxMG is to use a heuristic switch to choose between these two formulations. Letting $\omega^y = \bar{a}_{i,j}^{W} + \bar{a}_{i,j}^{E}$, the denominator of the interpolation weights is defined as,

$$\mathcal{A}_{i,j}^{Oy} = \begin{cases} \bar{a}_{i,j}^{Oy} & a_{i,j}^{O} > (1+\epsilon^y)\omega^y \\ \omega^y & a_{i,j}^{O} \leq (1+\epsilon^y)\omega^y \end{cases},$$

where $\epsilon^y = \min(|\bar{a}_{i,j}^{W}|/a_{i,j}^{O}, |\bar{a}_{i,j}^{E}|/a_{i,j}^{O})$. An analogous switch is defined for the fine-grid points embedded in y-lines, and for fine-grid points in the interior of coarse-grid cells. This formulation is referred to as *switched* interpolation.

2.2. Adaptive BoxMG interpolation

Adjusting the interpolation formulae in Equations (4), (5), and (7) to account for modified information about the near-null space of matrix A can be done following a similar approach as in [6]. In (4) and (5), connections between the fine-grid point (i, j) and its fine-grid neighbors are collapsed based on the assumption that the unwanted values of the fine-grid error vector could be accurately approximated directly by the values at their neighbors on the coarse grid, or by the error at point (i, j) itself. This is a natural assumption when the matrix, A, comes from a standard discretization of an elliptic differential operator, for which the constant vector gives a good local representative of the near-null space of the matrix. As such, this is closely related to the standard AMG assumption that algebraically smooth errors vary slowly along strong connections [18].

When the constant vector doesn't give a good representation of the near-null space of the matrix, adaptive multigrid methods [4–6, 11, 47–50] make use of so-called *prototype* vectors, that are assumed to give an accurate local picture of the variation in the near-null space of the matrix or, equivalently, in the slow-to-converge errors for simple pointwise relaxation schemes, such as weighted Jacobi or Gauss-Seidel. Because we consider such adaptivity within the setting of the scalar BoxMG algorithm (and, in particular, not its systems variant [22]), we only consider the case where such information is provided in the form of a single prototype vector, $\mathbf{z}$, although recent experience with the scalar gauged-Laplacian operator from quantum chromodynamics suggests that this may limit the applicability of the resulting algorithm [47].

Thus, given a single prototype vector, $\mathbf{z}$, we propose to modify the interpolation rules given in Equations (4) and (5) based on indirect fitting of $\mathbf{z}$. When point (i, j) is embedded in a coarse-grid line in the x-direction, instead of making the approximations in Equation (2) that lead to (4), we define local scaling factors based on the prototype,

$$w_{k,j}^{S} = \frac{z_{k,j-1}}{z_{k,j}} \text{ and } w_{k,j}^{N} = \frac{z_{k,j+1}}{z_{k,j}}, \text{ for } k = i-1, i, i+1,$$

and make weighted approximations to the error in the y direction,

$$e_{k,j-1} \approx w_{k,j}^{S} e_{k,j} \text{ and } e_{k,j+1} \approx w_{k,j}^{N} e_{k,j}, \text{ for } k = i-1, i, i+1.$$

These factors locally capture the slope of the error transverse to the direction of interpolation, leading to the definition of a rich interpolation space that appropriately represents the prototype vector. With these approximations, the overall approximation of Equation (1)

Prepared using nlaauth.cls
DOI: 10.1002/nla

becomes

$$\begin{aligned}\left(w^S_{i,j}a^S_{i,j} + a^O_{i,j} + w^N_{i,j}a^N_{i,j}\right)e_{i,j} = &-\left(w^S_{i-1,j}a^{SW}_{i,j} + a^W_{i,j} + w^N_{i-1,j}a^{NW}_{i,j}\right)e^C_{I,J}\\ &-\left(w^S_{i+1,j}a^{SE}_{i,j} + a^E_{i,j} + w^N_{i+1,j}a^{NE}_{i,j}\right)e^C_{I+1,J},\end{aligned}$$

which gives the interpolation formula,

$$(P\mathbf{e}^C)_{i,j} = -\frac{w^S_{i-1,j}a^{SW}_{i,j} + a^W_{i,j} + w^N_{i-1,j}a^{NW}_{i,j}}{w^S_{i,j}a^S_{i,j} + a^O_{i,j} + w^N_{i,j}a^N_{i,j}}e^C_{I,J} - \frac{w^S_{i+1,j}a^{SE}_{i,j} + a^E_{i,j} + w^N_{i+1,j}a^{NE}_{i,j}}{w^S_{i,j}a^S_{i,j} + a^O_{i,j} + w^N_{i,j}a^N_{i,j}}e^C_{I+1,J}.$$

Rewriting this in terms of only entries in A and $\mathbf{z}$ gives

$$\begin{aligned}(P\mathbf{e}^C)_{i,j} = &-\frac{z_{i,j}\left(a^{SW}_{i,j}z_{i-1,j-1} + a^W_{i,j}z_{i-1,j} + a^{NW}_{i,j}z_{i-1,j+1}\right)}{z_{i-1,j}\left(a^S_{i,j}z_{i,j-1} + a^O_{i,j}z_{i,j} + a^N_{i,j}z_{i,j+1}\right)}e^C_{I,J}\\ &-\frac{z_{i,j}\left(a^{SE}_{i,j}z_{i+1,j-1} + a^E_{i,j}z_{i+1,j} + a^{NE}_{i,j}z_{i+1,j+1}\right)}{z_{i+1,j}\left(a^S_{i,j}z_{i,j-1} + a^O_{i,j}z_{i,j} + a^N_{i,j}z_{i,j+1}\right)}e^C_{I+1,J}.\end{aligned} \tag{8}$$

Similarly, to replace the interpolation formula given in Equation (5) for the case where point (i,j) is embedded in a coarse-grid line in the y-direction, we define the scaling factors,

$$w^W_{i,k} = \frac{z_{i-1,k}}{z_{i,k}} \text{ and } w^E_{i,k} = \frac{z_{i+1,k}}{z_{i,k}}, \text{ for } k = j-1, j, j+1,$$

and make weighted approximations to the error in the x direction,

$$e_{i-1,k} \approx w^W_{i,k}e_{i,k} \text{ and } e_{i+1,k} \approx w^E_{i,k}e_{i,k}, \text{ for } k = j-1, j, j+1.$$

Then, following the analogous derivation as in Equation (8), we can derive the adaptive version of Equation (5),

$$(P\mathbf{e}^C)_{i,j} = -\frac{w^W_{i,j-1}a^{SW}_{i,j} + a^S_{i,j} + w^E_{i,j-1}a^{SE}_{i,j}}{w^W_{i,j}a^W_{i,j} + a^O_{i,j} + w^E_{i,j}a^E_{i,j}}e^C_{I,J} - \frac{w^W_{i,j+1}a^{NW}_{i,j} + a^N_{i,j} + w^E_{i,j+1}a^{NE}_{i,j}}{w^W_{i,j}a^W_{i,j} + a^O_{i,j} + w^E_{i,j}a^E_{i,j}}e^C_{I,J+1}.$$

Again rewriting in terms of the entries of A and $\mathbf{z}$ gives

$$\begin{aligned}(P\mathbf{e}^C)_{i,j} = &-\frac{z_{i,j}\left(a^{SW}_{i,j}z_{i-1,j-1} + a^S_{i,j}z_{i,j-1} + a^{SE}_{i,j}z_{i+1,j-1}\right)}{z_{i,j-1}\left(a^W_{i,j}z_{i-1,j} + a^O_{i,j}z_{i,j} + a^E_{i,j}z_{i+1,j}\right)}e^C_{I,J}\\ &-\frac{z_{i,j}\left(a^{NW}_{i,j}z_{i-1,j+1} + a^N_{i,j}z_{i,j+1} + a^{NE}_{i,j}z_{i+1,j+1}\right)}{z_{i,j+1}\left(a^W_{i,j}z_{i-1,j} + a^O_{i,j}z_{i,j} + a^E_{i,j}z_{i+1,j}\right)}e^C_{I,J+1}.\end{aligned} \tag{9}$$

While modification of the definition of interpolation to fine-grid nodes that lie at the centers of coarse-grid cells, as given in (6), is possible (see Section 3), it is natural to keep the general form, and simply replace the values of $\left(P\mathbf{e}^C\right)_{i,j-1}$, $\left(P\mathbf{e}^C\right)_{i,j+1}$, $\left(P\mathbf{e}^C\right)_{i-1,j}$, and $\left(P\mathbf{e}^C\right)_{i+1,j}$ with those from Equations (8) and (9).

An immediate consequence of these definitions is a scaling-invariance property of the adaptive BoxMG interpolation scheme given in Equations (8) and (9). Just as in classical AMG [6], the performance of classical BoxMG is known to suffer significantly when the system matrix, A, is rescaled, even in the simplest ways; however, the adjustments in (8) and (9) are sufficient to "undo" the effects of scaling.

Theorem 1. *Let A be a given matrix with regular 9-point connectivity structure (i.e., a non-zero pattern contained within that of the standard 9-point tensor-product bilinear finite-element stencil on a regular mesh), and let S be a non-singular diagonal matrix. Define $\hat{A} = SAS$, and $\mathbf{z} = S^{-1}\mathbf{1}$ to be the prototype vector for adaptive BoxMG interpolation, so that $z_{i,j} = 1/s_{(i,j),(i,j)}$ is the value of the prototype at node (i,j). Let P denote the classical*

Prepared using nlaauth.cls
DOI: 10.1002/nla

BoxMG interpolation operator for matrix A, as given in Equations (4), (5), *and* (7). *Assume that all of*

$$\begin{aligned}
&a^O_{i,j}\\
&a^W_{i,j} + a^O_{i,j} + a^E_{i,j}\\
&a^S_{i,j} + a^O_{i,j} + a^N_{i,j}\\
&\hat{a}^W_{i,j} z_{i-1,j} + \hat{a}^O_{i,j} z_{i,j} + \hat{a}^E_{i,j} z_{i+1,j}\\
&\hat{a}^S_{i,j} z_{i,j-1} + \hat{a}^O_{i,j} z_{i,j} + \hat{a}^N_{i,j} z_{i,j+1}
\end{aligned}$$

are nonzero for each node (i,j) where these appear in the interpolation formulae, so that both the classical BoxMG interpolation operator for A and the adaptive BoxMG interpolation operator for $\hat{A}$ based on prototype $\mathbf{z}$, $\hat{P}$, are well-defined. Then, $\hat{P} = S^{-1}PS_c$, where S_c is the diagonal matrix whose values are given by taking the values of S at only the coarse-grid nodes.

Proof
We prove this result by considering each of the four cases discussed above.

1. (i,j) **is a coarse point:** In this case, both $\left(P\mathbf{e}^C\right)_{i,j} = e^C_{I,J}$ and $\left(\hat{P}\mathbf{e}^C\right)_{i,j} = e^C_{I,J}$. Note, however, that the entry on the diagonal of S_c corresponding to (I,J) is the same as the entry on the diagonal of S corresponding to (i,j), so that the effects of the left and right scalings in $S^{-1}PS_c$ on these rows cancel out.
2. $(i-1,j)$ **and** $(i+1,j)$ **are coarse points:** In this case, we have two formulae for the classical and adaptive cases, Equations (4) and (8), respectively, giving

$$\begin{aligned}
(P\mathbf{e}^C)_{i,j} =& -\frac{a^{SW}_{i,j} + a^W_{i,j} + a^{NW}_{i,j}}{a^S_{i,j} + a^O_{i,j} + a^N_{i,j}} e^C_{I,J} - \frac{a^{SE}_{i,j} + a^E_{i,j} + a^{NE}_{i,j}}{a^S_{i,j} + a^O_{i,j} + a^N_{i,j}} e^C_{I+1,J}\\
\text{and } (\hat{P}\mathbf{e}^C)_{i,j} =& -\frac{z_{i,j}\left(\hat{a}^{SW}_{i,j} z_{i-1,j-1} + \hat{a}^W_{i,j} z_{i-1,j} + \hat{a}^{NW}_{i,j} z_{i-1,j+1}\right)}{z_{i-1,j}\left(\hat{a}^S_{i,j} z_{i,j-1} + \hat{a}^O_{i,j} z_{i,j} + \hat{a}^N_{i,j} z_{i,j+1}\right)} e^C_{I,J}\\
&- \frac{z_{i,j}\left(\hat{a}^{SE}_{i,j} z_{i+1,j-1} + \hat{a}^E_{i,j} z_{i+1,j} + \hat{a}^{NE}_{i,j} z_{i+1,j+1}\right)}{z_{i+1,j}\left(\hat{a}^S_{i,j} z_{i,j-1} + \hat{a}^O_{i,j} z_{i,j} + \hat{a}^N_{i,j} z_{i,j+1}\right)} e^C_{I+1,J}.
\end{aligned}$$

   Noting that the scaling, $\hat{A} = SAS$, gives $\hat{a}^{SW}_{i,j} = a^{SW}_{i,j} z^{-1}_{i,j} z^{-1}_{i-1,j-1}$, $\hat{a}^W_{i,j} = a^W_{i,j} z^{-1}_{i,j} z^{-1}_{i-1,j}$, and so forth, we see that many of the terms, $z_{k,\ell}$, in the adaptive case cancel with the scaling in $\hat{A}$, ultimately giving

$$\hat{P}\mathbf{e}^C_{i,j} = -\frac{z_{i,j}\left(a^{SW}_{i,j} + a^W_{i,j} + a^{NW}_{i,j}\right)}{z_{i-1,j}\left(a^S_{i,j} + a^O_{i,j} + a^N_{i,j}\right)} e^C_{I,J} - \frac{z_{i,j}\left(a^{SE}_{i,j} + a^E_{i,j} + a^{NE}_{i,j}\right)}{z_{i+1,j}\left(a^S_{i,j} + a^O_{i,j} + a^N_{i,j}\right)} e^C_{I+1,J}.$$

   From this, we see that $\hat{p}_{(i,j),(I,J)} = \frac{z_{i,j}}{z_{i-1,j}} p_{(i,j),(I,J)}$ and that $\hat{p}_{(i,j),(I+1,J)} = \frac{z_{i,j}}{z_{i+1,j}} p_{(i,j),(I+1,J)}$. Further noting that the entry of S_c corresponding to (I,J) is $z^{-1}_{i-1,j}$ and that corresponding to $(I+1,J)$ is $z^{-1}_{i+1,j}$ (since $(i-1,j)$ and $(i+1,j)$ are the fine-grid points corresponding to these coarse-grid indices), establishes that $\hat{p}_{(i,j),(I,J)} = \left(S^{-1}PS_c\right)_{(i,j),(I,J)}$ and $\hat{p}_{(i,j),(I+1,J)} = \left(S^{-1}PS_c\right)_{(i,j),(I+1,J)}$.
3. $(i,j-1)$ **and** $(i,j+1)$ **are coarse points:** This case is essentially identical to the previous case, except that we now identify coarse points (I,J) and $(I,J+1)$ with fine points $(i,j-1)$ and $(i,j+1)$, respectively. Following the analogous steps, Equation (9) becomes

$$(\hat{P}\mathbf{e}^C)_{i,j} = -\frac{z_{i,j}\left(a^{SW}_{i,j} + a^S_{i,j} + a^{SE}_{i,j}\right)}{z_{i,j-1}\left(a^W_{i,j} + a^O_{i,j} + a^E_{i,j}\right)} e^C_{I,J} - \frac{z_{i,j}\left(a^{NW}_{i,j} + a^N_{i,j} + a^{NE}_{i,j}\right)}{z_{i,j+1}\left(a^W_{i,j} + a^O_{i,j} + a^E_{i,j}\right)} e^C_{I,J+1}.$$

Prepared using nlaauth.cls
DOI: 10.1002/nla

As before, we see the relationship that $\hat{p}_{(i,j),(I,J)} = \frac{z_{i,j}}{z_{i,j-1}} p_{(i,j),(I,J)}$ and $\hat{p}_{(i,j),(I,J+1)} = \frac{z_{i,j}}{z_{i,j+1}} p_{(i,j),(I,J+1)}$; a similar argument shows that $\hat{p}_{(i,j),(I,J)} = \left(S^{-1}PS_c\right)_{(i,j),(I,J)}$ and $\hat{p}_{(i,j),(I,J+1)} = \left(S^{-1}PS_c\right)_{(i,j),(I,J+1)}$.

4. $(i-1,j-1)$**,** $(i+1,j-1)$**,** $(i-1,j+1)$**, and** $(i+1,j+1)$ **are coarse points:** In this case, the formula for $\hat{A}$ follows that for A, with entries from $\hat{P}$ replacing those of P:

$$\begin{aligned}\left(\hat{P}\mathbf{e}^C\right)_{i,j} = &- \frac{\hat{a}^{SW}_{i,j} + \hat{a}^{S}_{i,j}\hat{p}_{(i,j-1),(I,J)} + \hat{a}^{W}_{i,j}\hat{p}_{(i-1,j),(I,J)}}{\hat{a}^{O}_{i,j}} e^C_{I,J} \\ &- \frac{\hat{a}^{SE}_{i,j} + \hat{a}^{S}_{i,j}\hat{p}_{(i,j-1),(I+1,J)} + \hat{a}^{E}_{i,j}\hat{p}_{(i+1,j),(I+1,J)}}{\hat{a}^{O}_{i,j}} e^C_{I+1,J} \\ &- \frac{\hat{a}^{NW}_{i,j} + \hat{a}^{N}_{i,j}\hat{p}_{(i,j+1),(I,J+1)} + \hat{a}^{W}_{i,j}\hat{p}_{(i-1,j),(I,J+1)}}{\hat{a}^{O}_{i,j}} e^C_{I,J+1} \\ &- \frac{\hat{a}^{NE}_{i,j} + \hat{a}^{N}_{i,j}\hat{p}_{(i,j+1),(I+1,J+1)} + \hat{a}^{E}_{i,j}\hat{p}_{(i+1,j),(I+1,J+1)}}{\hat{a}^{O}_{i,j}} e^C_{I+1,J+1},\end{aligned}$$

where the coarse-grid indices (I,J), $(I+1,J)$, $(I,J+1)$, and $(I+1,J+1)$ refer to fine-grid points $(i-1,j-1)$, $(i+1,j-1)$, $(i-1,j+1)$, and $(i+1,j+1)$, respectively. Making the appropriate substitutions for entries in $\hat{A}$ and $\hat{P}$, we have

$$\begin{aligned}\hat{p}_{(i,j),(I,J)} &= -\frac{\hat{a}^{SW}_{i,j} + \hat{a}^{S}_{i,j}\hat{p}_{(i,j-1),(I,J)} + \hat{a}^{W}_{i,j}\hat{p}_{(i-1,j),(I,J)}}{\hat{a}^{O}_{i,j}} \\ &= -\frac{a^{SW}_{i,j} z^{-1}_{i,j} z^{-1}_{i-1,j-1}}{a^{O}_{i,j} z^{-2}_{i,j}} - \frac{a^{S}_{i,j} z^{-1}_{i,j} z^{-1}_{i,j-1} p_{(i,j-1),(I,J)} \frac{z_{i,j-1}}{z_{i-1,j-1}}}{a^{O}_{i,j} z^{-2}_{i,j}} \\ &\qquad - \frac{a^{W}_{i,j} z^{-1}_{i,j} z^{-1}_{i-1,j} p_{(i-1,j),(I,J)} \frac{z_{i-1,j}}{z_{i-1,j-1}}}{a^{O}_{i,j} z^{-2}_{i,j}} \\ &= -\frac{z_{i,j}\left(a^{SW}_{i,j} + a^{S}_{i,j} p_{(i,j-1),(I,J)} + a^{W}_{i,j} p_{(i-1,j),(I,J)}\right)}{z_{i-1,j-1} a^{O}_{i,j}} = \frac{z_{i,j}}{z_{i-1,j-1}} p_{(i,j),(I,J)}.\end{aligned}$$

Similar calculations for the other coefficients show that

$$\begin{aligned}\hat{p}_{(i,j),(I+1,J)} &= \frac{z_{i,j}}{z_{i+1,j-1}} p_{(i,j),(I+1,J)}, \\ \hat{p}_{(i,j),(I,J+1)} &= \frac{z_{i,j}}{z_{i-1,j+1}} p_{(i,j),(I,J+1)}, \\ \text{and } \hat{p}_{(i,j),(I+1,J+1)} &= \frac{z_{i,j}}{z_{i+1,j+1}} p_{(i,j),(I+1,J+1)}.\end{aligned}$$

As before, this shows that these coefficients are also scaled as given by $\hat{P} = S^{-1}PS_c$.

□

The desired corollary to Theorem 1 is that the performance of BoxMG based on adaptive interpolation applied to $\hat{A}\hat{\mathbf{x}} = \hat{\mathbf{b}}$ is equivalent to that of classical BoxMG applied to $A\mathbf{x} = \mathbf{b}$; however, the truth of this depends on both the choice of smoother and availability of entries in S. If the error-propagation operator for the BoxMG smoother applied to A is $I - MA$, while that applied to $\hat{A}$ is $I - \hat{M}\hat{A}$, then the performance of BoxMG based on adaptive interpolation applied to $\hat{A}\hat{\mathbf{x}} = \hat{\mathbf{b}}$ depends on how $\hat{M}$ relates to S. For standard smoothers, such as Jacobi and Gauss-Seidel, $\hat{M}^{-1} = SM^{-1}S$ and, so, $I - \hat{M}\hat{A} = S^{-1}(I - MA)S$. The same relation holds for line smoothers, or any other smoother where M^{-1} is chosen by taking a fixed subset of the nonzero elements of A. Under the assumption that $\hat{M}^{-1} = SM^{-1}S$, it can be seen that the overall solver performance of BoxMG based on adaptive interpolation applied to $\hat{A}\hat{\mathbf{x}} = \hat{\mathbf{b}}$ is

DOI: 10.1002/nla

equivalent to that of classical BoxMG applied to $A\mathbf{x} = \mathbf{b}$, when the diagonal scaling S is known exactly. This, of course, is not a practical assumption, although results in Section 4 show that a good enough approximation can usually be found such that the performance doesn't suffer very much.

2.3. Adaptive setup procedure

In this paper, we make use of a simple adaptive setup cycling scheme, as was considered in [6], where the prototype vector is initialized to be a random vector, and an initial series of setup V-cycles is performed. On each level of these cycles, a fixed number of pre-relaxations is performed on the homogeneous problem, $A\mathbf{z} = \mathbf{0}$ (with the current prototype on that level used as the initial guess), an adaptive interpolation operator, P, is formed based on the resulting improved prototype, which is then injected to give an initial prototype vector on the next coarser grid along with the computed Galerkin coarse-grid operator, P^TAP. After these steps, the process moves to the next coarser grid and begins again, until the coarsest grid is reached. No computation is done on the coarsest grid (although further relaxation or an exact eigensolve might be used to improve the process [6, 7, 49]). On the upward traversal of the V-cycle, the coarser-grid prototype is interpolated to the finer grid based on the appropriate adaptive interpolation operator defined on the downward traversal, again a fixed number of relaxations on $A\mathbf{z} = \mathbf{0}$ is performed with the interpolated prototype as the initial guess, but no update is computed for the interpolation operators. (Thus, if we are performing a fixed number of cycles, we skip this relaxation on the final cycle to avoid unnecessary work.) As was noted in [6], this is not typically the most efficient approach to the adaptive cycling when only one or two relaxations are used on each level in the setup V-cycle, as we consider in Section 4; rather, this serves to illustrate the overall performance of the adaptive approach.

After these setup cycles are performed, we take the interpolation and Galerkin coarse-grid operators defined on the last adaptive cycle and use these in the solve phase of the algorithm. The solve phase makes use of standard multigrid V-cycles in the usual way. Here, we follow the BoxMG solution algorithm and make use of either pointwise or linewise Gauss-Seidel relaxation and Galerkin coarse-grid correction.

3. CONNECTIONS BETWEEN BOXMG AND AMG

The equivalence given in Theorem 1 is effectively the same as that observed for classical and adaptive AMG in [6], and relies on similar cancellations occurring when the scaled coefficients of $\hat{A}$ are introduced into the adaptive interpolation formula. Thus, it is natural to investigate the connection between both the classical and adaptive interpolation formulae for BoxMG and AMG. Here, we show that BoxMG can be viewed as a "special case" of AMG, based on geometric choices of the definitions of strong connections, rather than the typical algebraic choice based on classical M-matrix properties [18].

The primary difference between the coarsening algorithms employed by BoxMG and AMG is in the selection of coarse grids, with BoxMG using standard geometric coarsening, while AMG uses an algebraic approach based on the graph of the matrix, A, filtered based on "strong connections". If A has a standard 9-point stencil with all connections deemed to be strong, then the standard "fully coarsened" grid used by BoxMG is an allowable coarse grid for the classical AMG coloring algorithm [51]. While this standard coarsening can also arise in other situations, our goal here is, instead, to examine the interpolation processes rather than the coarse-grid selection algorithms. However, it is interesting to note some of the fundamental differences between BoxMG and AMG that result from these very different approaches to coarsening. First, since BoxMG uses a geometric coarsening that is independent of the problem, some fine-scale features of the problem, such as discontinuities in the continuum diffusion coefficients, will not be preserved on coarser grids. In these cases, the operator-induced interpolation must capture these features and, through the Galerkin coarse-grid operator,

Prepared using nlaauth.cls
DOI: 10.1002/nla

represent their influence on coarser grids. In contrast, the algebraic approach to coarsening in classical AMG naturally attempts to preserve these features on coarser grids and, hence, tends to coarsen along features but not across them. Similarly, for anisotropic problems, AMG aims to achieve the semicoarsened grids needed for efficient multigrid solution using pointwise smoothers, while BoxMG defaults to using the coupled line relaxation (and, in 3D, plane relaxation) approaches needed for efficient multigrid solution using full geometric coarsening.

Based on this structured coarse grid, the BoxMG interpolation formulae are given above in Section 2. In contrast, the AMG interpolation formulae are determined based on only two cases, when a fine-grid point is also a coarse-grid point, and when it isn't. When node (i,j) is both a fine-grid and coarse-grid point, then the AMG interpolation is the same as BoxMG, $\left(P\mathbf{e}^C\right)_{i,j} = e^C_{I,J}$, where (I,J) is the coarse-grid index corresponding to fine-grid point (i,j).

When (i,j) doesn't correspond to a coarse-grid point, the definition of AMG interpolation begins with the same small-residual assumption that gives Equation (1). From here, the neighboring points, indexed by $d \in \{SW, S, SE, W, E, NW, N, NE\}$, are divided into the strongly connected coarse-grid, $C_{i,j}$, and fine-grid, $F_{i,j}$, neighbors and the weakly connected neighbors, $W_{i,j}$. Then, Equation (1) can be rewritten as

$$a^O_{i,j} e_{i,j} = -\sum_{d\in C_{i,j}} a^d_{i,j} e_{(i,j)+\vec{d}} - \sum_{d'\in F_{i,j}} a^{d'}_{i,j} e_{(i,j)+\vec{d'}} - \sum_{d''\in W_{i,j}} a^{d''}_{i,j} e_{(i,j)+\vec{d''}},$$

where we use the notation $e_{(i,j)+\vec{d}}$ to denote the error at the grid point in direction d from (i,j); for example, $e_{(i,j)+\overrightarrow{SW}} = e_{i-1,j-1}$. For directions $d \in C_{i,j}$, node $(i,j)+\vec{d}$ is a coarse-grid node, and so these values can be used directly in interpolation. The goal in the AMG definition of interpolation is to "collapse" connections to $d \in F_{i,j} \cup W_{i,j}$ onto those in $C_{i,j}$.

Weak connections, $d'' \in W_{i,j}$, occur when $a^{d''}_{i,j}$ is small enough that the approximation of $e_{(i,j)+\vec{d''}}$ is unimportant, so long as a significant error in scaling isn't made. In this case, it is typical to assume that $e_{(i,j)+\vec{d''}} \approx e_{i,j}$, so that the connection is said to be collapsed onto the diagonal. For strong connections between (i,j) and one of its fine-grid neighbors, $d' \in F_{i,j}$, AMG interpolation makes the approximation that the error at node $(i,j)+\vec{d'}$ can be written as a linear combination of the errors at the coarse-grid points $d \in C_{i,j}$,

$$e_{(i,j)+\vec{d'}} \approx \sum_{d\in C_{i,j}} w^{d',d}_{i,j} e_{(i,j)+\vec{d}}.$$

To determine the coefficients, $w^{d',d}_{i,j}$, we again start with a modified form of Equation (1),

$$a^O_{(i,j)+\vec{d'}}\, e_{(i,j)+\vec{d'}} = -\sum_{d\in C_{i,j}} a^{d-d'}_{(i,j)+\vec{d'}}\, e_{(i,j)+\vec{d}} - \sum_{d''\notin C_{i,j}\cup\{d'\}} a^{d''-d'}_{(i,j)+\vec{d'}}\, e_{(i,j)+\vec{d''}}, \tag{10}$$

where the notation $d-d'$ refers to the net direction between nodes $(i,j)+\vec{d'}$ and $(i,j)+\vec{d}$, when d' and d are directions measured from node (i,j). For example, if $d' = SW$ and $d = W$, then $d-d' = N$, the direction from $(i-1,j-1)$ to $(i-1,j)$. Note that $d-d'$ is not well-defined for all pairs of directions (e.g., if $d' = SW$ and $d = N$); the coefficient $a^{d-d'}_{(i,j)+\vec{d'}}$ is taken to be zero if $d-d'$ is not well defined.

To turn Equation (10) into an interpolation weighting to collapse the connection to $a^{d'}_{i,j}$, we introduce an approximation, where the second sum on the right-hand side is discarded, and the coefficient of $e_{(i,j)+\vec{d'}}$ is modified to compensate. In classical AMG, this is done based on the assumption that the slow-to-converge errors of relaxation (the algebraically smooth errors) locally match the constant vector, so that the interpolation from $d \in C_{i,j}$ to $(i,j)+\vec{d'}$ should also preserve the constant vector. With this assumption, the weights $w^{d',d}_{i,j}$ for $d \in C_{i,j}$ are

DOI: 10.1002/nla

given by

$$w_{i,j}^{d',d} = \frac{a_{(i,j)+\vec{d}'}^{d-d'}}{\sum_{d''\in C_{i,j}} a_{(i,j)+\vec{d}'}^{d''-d'}}. \tag{11}$$

Putting these all together gives the approximation of Equation (1) of

$$\left(a_{i,j}^{O} + \sum_{d''\in W_{i,j}} a_{i,j}^{d''}\right) e_{i,j} = -\sum_{d\in C_{i,j}} a_{i,j}^{d} e_{(i,j)+\vec{d}} - \sum_{d'\in F_{i,j}} a_{i,j}^{d'} \sum_{d\in C_{i,j}} \frac{a_{(i,j)+\vec{d}'}^{d-d'}}{\sum_{d''\in C_{i,j}} a_{(i,j)+\vec{d}'}^{d''-d'}} e_{(i,j)+\vec{d}}, \tag{12}$$

which can easily be transformed into an interpolation operator after consolidating terms on the right-hand side and dividing by the coefficient on the left-hand side.

While Equation (12) and its derivation appear to be completely unrelated to the BoxMG formulae derived in Section 2, there are, nonetheless, similarities in the treatment of connections from node (i,j) to fine-grid neighbors in Equation (1). Considering Equation (3), we see that this has similar form, with the term $a_{i,j}^{S} + a_{i,j}^{O} + a_{i,j}^{N}$ replacing the sum on the left-hand side, and simple sums of coefficients on the right-hand side. Note, however, that in this case, nodes $(i+1,j+1)$ and $(i+1,j-1)$ have only a single connection to $d \in C_{i,j}$, for $d = E$, so that $w_{i,j}^{NE,E} = w_{i,j}^{SE,E} = 1$ and, similarly, $w_{i,j}^{NW,W} = w_{i,j}^{SW,W} = 1$. Thus, one observation is that the BoxMG interpolation formula in Equation (4) is the same as the AMG interpolation formula in this case, where nodes $(i, j\pm 1)$ are treated as weakly connected neighbors $(S, N \in W_{i,j})$, and $(i-1, j\pm 1)$ and $(i+1, j\pm 1)$ are treated as strongly connected fine-grid neighbors $(SW, NW, SE, NE \in F_{i,j})$, with $(i\pm 1, j)$ as coarse-grid neighbors, $W, E \in C_{i,j}$.

This equivalence is, however, somewhat unsatisfactory, as there is no apparent justification in treating the connections to nodes $(i, j\pm 1)$ as weak connections. A more satisfying point of view arises from considering an analogue of Equation (10) in which collapsing strong fine-grid connections is allowed to both (i,j) and $d \in C_{i,j}$:

$$a_{(i,j)+\vec{d}'}^{O}\, e_{(i,j)+\vec{d}'} = -\sum_{d\in C_{i,j}\cup\{O\}} a_{(i,j)+\vec{d}'}^{d-d'}\, e_{(i,j)+\vec{d}} - \sum_{d''\notin C_{i,j}\cup\{O,d'\}} a_{(i,j)+\vec{d}'}^{d''-d'}\, e_{(i,j)+\vec{d}''},$$

leading to the analogue of Equation (11) for $d \in C_{i,j}\cup\{O\}$,

$$w_{i,j}^{d',d} = \frac{a_{(i,j)+\vec{d}'}^{d-d'}}{\sum_{d''\in C_{i,j}\cup\{O\}} a_{(i,j)+\vec{d}'}^{d''-d'}}.$$

With this definition of $w_{i,j}^{d',d}$, the modified AMG interpolation weights and the classical BoxMG interpolation weights coincide when the definition of strong connections used in determining the AMG interpolation weights is based on the simple criterion of geometric distance. For each fine-grid neighbor of (i,j), (k,ℓ), we say that (k,ℓ) strongly depends on node $(k',\ell') \in C_{i,j}\cup(i,j)$ if and only if the geometric distance between (k,ℓ) and (k',ℓ') is minimal over all points in $C_{i,j}\cup(i,j)$. Note that we specifically use geometric distance here rather than graph distance, since the two are not the same for a standard 9-point stencil. A similar conclusion holds for the analogue of Equation (5).

For the case where (i,j) is in the center of a coarse-grid cell, given in Equation (7), the inclusion of node (i,j) in the distribution does not give equivalence between the AMG and BoxMG approaches. The "usual" indirect interpolation weights, given by Equation (11) may give this equivalence, depending on the relationship between the weights prescribed by Equation (11) and the BoxMG weights given in Equations (4) and (5). When these weights are the same (as, for instance, they will be in cases where both result in linear interpolation along cell edges, such as for a constant-coefficient diffusion problem), then the BoxMG and

Prepared using nlaauth.cls
DOI: 10.1002/nla

AMG interpolation rules coincide. A plausible rule to distinguish between the cases where (i, j) should and shouldn't be included in the distribution is based on whether or not node (k, ℓ) is strongly connected (based on geometric distance) to only node (i, j) or to some nodes in $C_{i,j}$. When the strong connection is only to node (i, j), as in the horizontal and vertical line interpolation cases, the generalized formula should be used, since the denominator in the classical formula (Equation (11)) is zero. This occurs because the fully coarsened grid enforced by BoxMG violates the typical AMG assumption that every pair of strongly connected fine-grid points should have a common coarse-grid neighbor that is strongly connected to each of the fine-grid points. When this assumption is not violated, as in the case when (i, j) is a coarse-grid cell center, the classical AMG collapsing should be used.

This connection between BoxMG and AMG is, to our knowledge, a new perspective on these two robust multigrid approaches. While BoxMG has long been acknowledged as an efficient and robust multigrid solution algorithm, outperforming AMG by a factor of 10 in solving 3D heterogeneous diffusion equations in [1], this connection provides the opportunity to investigate a broader family of structured robust multigrid methods, in both the classical and adaptive frameworks. Among the possible "hybrid" approaches of structured coarsening with algebraic interpolation are

1. Using classical or adaptive AMG interpolation on structured coarsening meshes,
2. Using classical or adaptive AMG interpolation for interpolation along coarse-grid cell edges, in place of Equations (4) and (5), combined with BoxMG interpolation for coarse-grid cell centers, and
3. Using classical or adaptive BoxMG interpolation for interpolation along coarse-grid cell edges, combined with classical or adaptive AMG interpolation for coarse-grid cell centers.

In all of these cases, because of the explicit grid structure, we can derive fully explicit formulae if we use geometric or graph-based definitions of strength of connection. Even with algebraic definitions of strength of connection, however, we can still make use of the known coarsening structure to make use of direct data access in achieving significant speedup over naïve AMG implementations. These hybrid approaches are investigated in Section 4.1.2.

4. NUMERICAL RESULTS

In this section, we explore the performance of adaptive BoxMG interpolation operators, in comparison to the classical BoxMG algorithm, as well as with interpolation based on AMG and adaptive AMG principles. We consider two examples where the fine-grid matrix comes from a bilinear finite-element discretization of the second-order form of Darcy's Law, $-\nabla \cdot \mathcal{K}(\mathbf{x})\nabla p(\mathbf{x}) = q(\mathbf{x})$ on $[0, 1]^2$, with variable permeability $\mathcal{K}(\mathbf{x})$. In the first examples, $\mathcal{K}(\mathbf{x})$ has a simple two-scale periodic structure that allows easy comparison and analysis of the numerical performance of these algorithms. The second set of examples are more realistic, using a geostatistical approach to generate layered media, with corresponding values of $\mathcal{K}(\mathbf{x})$. Finally, we consider test problems based on five-point finite-difference discretizations of both isotropic and anisotropic diffusion problems, with a focus on the effects of non-standard boundary conditions.

4.1. Periodic Permeability

Our first tests focus on the simple case of a periodic, two-scale diffusion problem, with a structured pattern of high-permeability square inclusions against a homogeneous background, as depicted on the left of Figure 3. This permeability is constructed based on a 4×4 tiling of the unit cell shown on the right of Figure 3, where $\mathcal{K}(\mathbf{x}) = 1000$ inside the dark regions and $\mathcal{K}(\mathbf{x}) = 1$ in the background medium. Note that with a 64×64 element finest grid, this permeability field is perfectly resolved on the given finest grid, but it is not accurately resolved on any coarser grid in the multigrid hierarchy.

Prepared using nlaauth.cls
DOI: 10.1002/nla

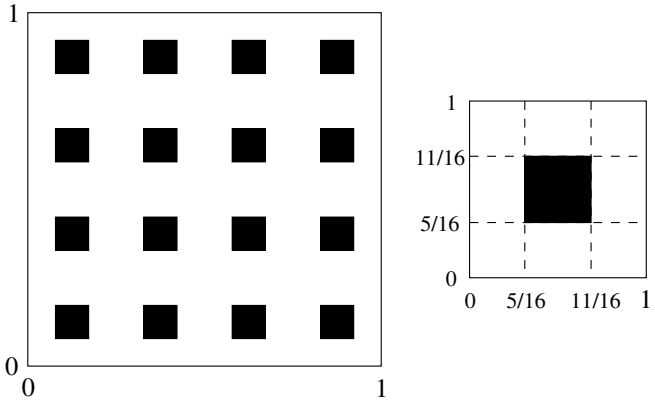


Figure 3. Periodic permeability field. At left, the 4×4 tiling of a unit cell (shown at right) that is used in the numerical examples that follow.

4.1.1. Neumann Boundary Conditions A natural first experiment is to see if the adaptive BoxMG approach can recover the performance seen by the classical BoxMG approach in the case where the ideal prototype vector is the constant vector. While classical BoxMG and AMG effectively use the constant vector as the prototype for all problems, in the adaptive setup stage (regardless of the final choice of adaptive interpolation), the constant vector arises as the ideal prototype only if it is the (unique) eigenvector associated with the largest (in modulus) eigenvalue of the relaxation process. A natural way to guarantee that this is the case is to consider the singular problem that arises from considering Neumann boundary conditions on all edges of the domain. The (unique) zero eigenvalue of the system matrix, A, becomes the (unique) unit eigenvalue of the semi-convergent relaxation process, $I - MA$, for standard choices of M, such as in weighted Jacobi or Gauss-Seidel relaxation. This, then, in the adaptive viewpoint, makes the associated eigenvector (the constant vector) the ideal vector on which to base interpolation.

The obvious disadvantage of considering the case with only Neumann Boundary conditions is the singularity of the resulting linear system, and its effect on the convergence of the resulting multigrid V-cycle iteration. To address this, we employ two strategies. Following classical BoxMG [24], for problems with a non-zero right-hand side (i.e., in the case where we are interested in the solution of a PDE and not in measuring asymptotic convergence factors), the singularity is treated only on the coarsest grid. Here, we modify the last diagonal entry of the coarsest-grid matrix to make it artificially non-singular before factoring it; after solving this perturbed coarsest-grid system in each V-cycle, we add a projection step, making the coarsest-grid correction orthogonal to the coarsest-grid constant vector, to ensure that the coarse-grid correction process does not introduce components in the direction of the null space. For the case of a non-zero right-hand side, this approach is sufficient to restore optimal multigrid convergence to the level of machine precision; however, it does not allow for testing of asymptotic convergence rates by applying the multigrid V-cycle to a problem with a zero right-hand side (with a random initial guess). In that case, we add an additional projection step against the finest-grid constant vector, to eliminate the effects of the singularity on the measurement of the performance of the resulting multigrid methods. We note that both of these approaches rely on the assumption that the null-space vector is known with sufficient accuracy to make these projections effective. In the case of the projection on the coarsest-grid, this means that the pre-image of the finest-grid null-space is assumed to be known; this may be reasonable for standard finite-element problems, such as are typically considered, but adds an additional challenge for problems where this is not the case.

Our first test is to examine how the adaptive process converges to a steady-state, and how that convergence can be related to the case of classical BoxMG. In Table I, we compare the

DOI: 10.1002/nla

Table I. Average and last-cycle convergence factors ($\overline{\rho}$ and ρ_{100}, resp.) for adaptive BoxMG, as well as Rayleigh quotients for prototype vectors used, as a function of number of adaptive setup cycles performed.

# of setup cycles	1	2	3	4	5	6	Classical
$\overline{\rho}$	0.89	0.81	0.37	0.081	0.077	0.077	0.077
ρ_{100}	0.99	0.91	0.42	0.091	0.087	0.087	0.087
RQ	4.10	$4.54e{-2}$	$1.82e{-3}$	$1.61e{-4}$	$1.47e{-5}$	$7.23e{-7}$	0

convergence factors of the resulting BoxMG V-cycle (using the adaptive BoxMG interpolation described above) to that of classical BoxMG applied to this problem. We report both the convergence factor averaged over 100 iterations, $\overline{\rho}$, and the convergence factor measured for the 100^{th} iteration, ρ_{100}, for each test. Additionally, we measure the Rayleigh quotient of the final prototype vector used on the finest grid. For all experiments with this problem, we make use of 1 pre-relaxation and 1 post-relaxation in the adaptive setup cycles.

Two important trends can be seen in Table I. First, the performance of the resulting V-cycles using the adaptive setup improves uniformly until it matches that of the classical (non-adaptive) BoxMG algorithm. Secondly, the Rayleigh Quotient of the adaptive prototype vector is uniformly decreasing, by over an order of magnitude per cycle, to zero, the Rayleigh Quotient of the true null-space vector.

Another way to monitor the convergence of the adaptive cycle is to look at the convergence of entries in the resulting interpolation operators. While this could be measured in an entrywise manner (e.g., using the Frobenius norm) at each level, this convergence is complicated by the effects of the changing Galerkin coarse-grid operators with each adaptive cycle. Instead, we make use of an approach from [52] to visualize the adaptation of the entries in the interpolation operator in terms of evolving coarse-scale basis functions.

On each scale, k (counting $k = k_f$ to be the finest grid, with decreasing k indicating depth in the multigrid hierarchy), we recursively define a set of basis functions $\{\psi_j^k\}$, starting from the standard bilinear basis functions used on the finest scale. Thus, on the finest scale, $k = k_f$, we have the given set of basis functions, $\{\psi_j^{k_f}\}$, for all fine-scale nodes, j, used to define the fine-scale matrix by

$$a_{ij} = \int_\Omega \left\langle \mathcal{K}(\mathbf{x}) \nabla \psi_j^{k_f}, \nabla \psi_i^{k_f} \right\rangle d\Omega.$$

Writing interpolation, P_{k-1}^k, from level $k-1$ to level k as $\left(P_{k-1}^k\right)_{ij} = p_{ij}^k$, we can compute the Galerkin coarse-grid operator, $A_{k-1} = \left(P_{k-1}^k\right)^T A_k P_{k-1}^k$, as

$$\begin{aligned}(A_{k-1})_{ij} &= \sum_{l,m} p_{li}^k (A_k)_{lm} p_{mj}^k = \sum_{l,m} p_{li}^k \, p_{mj}^k \int_\Omega \langle K(\mathbf{x}) \nabla \psi_m^k, \nabla \psi_l^k \rangle d\Omega \\ &= \int_\Omega \left\langle K(\mathbf{x}) \nabla \left(\sum_m p_{mj}^k \, \psi_m^k \right), \nabla \left(\sum_l p_{li}^k \, \psi_l^k \right) \right\rangle d\Omega.\end{aligned}$$

Thus, we can recursively define multiscale basis functions on level $k-1$ as

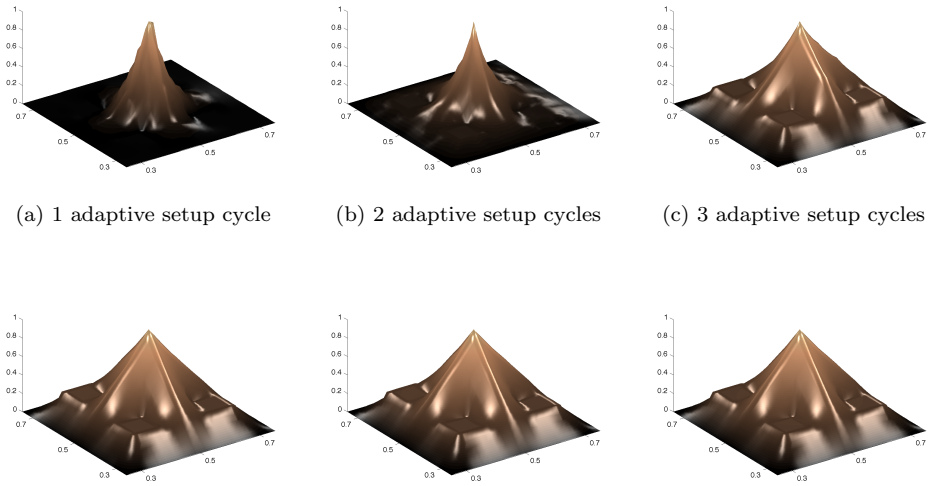
$$\psi_j^{k-1} = \sum_m p_{mj}^k \psi_m^k, \tag{13}$$

and write the discrete coarse-grid operator in the form

$$(A_{k-1})_{ij} = \int_\Omega \left\langle \mathcal{K}(\mathbf{x}) \nabla \psi_j^{k-1}, \nabla \psi_i^{k-1} \right\rangle d\Omega.$$

We can then use the recursive definition (13) to define coarse-scale basis functions associated with both the adaptive interpolation operators and the classical BoxMG interpolation operator

Prepared using nlaauth.cls
DOI: 10.1002/nla



(a) 1 adaptive setup cycle (b) 2 adaptive setup cycles (c) 3 adaptive setup cycles

(d) 4 adaptive setup cycles (e) 5 adaptive setup cycles (f) Classical BoxMG

Figure 4. Multiscale basis functions from the 4×4 coarse grid associated with the node at $(0.5, 0.5)$ for the 64×64 element fine-grid discretization of the diffusion equation with periodic permeability given as in Figure 3.

(which was investigated from this viewpoint in [34]). Figure 4 shows the basis functions associated with the experiments in Table I. Note that after only one or two adaptive setup cycles, the basis functions shown bear little resemblance to that of classical BoxMG, nor do they reflect the fine-scale permeability structure in any meaningful way. Thus, it is not surprising that the multigrid V-cycle based on these interpolation operators is ineffective. While there is little qualitative difference between the basis function after three setup cycles and those after four or five setup cycles (or from classical BoxMG), it is clear that the evolution from three to four adaptive setup cycles makes the difference between achieving reasonable convergence factors and optimal convergence factors. Thus, while the defects in Figure 4c are small, they are important in the resulting multigrid convergence factors.

4.1.2. Comparing adaptive AMG and BoxMG interpolation As discussed above, while both the classical and adaptive AMG interpolation formulae make implicit assumptions on the relationship between the coarse and fine grids, we can still pair these approaches with the regular coarse grids used by BoxMG. Table II gives results for both adaptive AMG and classical AMG (Method 1 in Section 3), for comparison with the adaptive and classical BoxMG results in Table I. Several notable differences occur between these two sets of results. First, comparing the performance of classical BoxMG with classical AMG, we see that classical AMG has slightly worse performance, likely due to the mismatch between the interpolation formula and choice of coarse grids. The adaptive process for AMG, however, converges slightly faster than that for BoxMG; excellent convergence is already seen after 3 adaptive setup cycles, and the adaptive AMG performance matches that of classical AMG after 4 adaptive setup cycles. This faster adaptive convergence is mirrored in the reduction of the Rayleigh Quotient of the prototype vector, which also converges to zero at a slightly quicker rate than in the adaptive BoxMG results.

Two interesting variants on these approaches occur when we consider hybrid interpolation schemes that use one of the adaptive BoxMG and AMG interpolation formulae for interpolating along edges in the coarse mesh and the other for interpolating to coarse-grid cell centers,

DOI: 10.1002/nla

Table II. Average and last-cycle convergence factors ($\overline{\rho}$ and ρ_{100}, resp.) for adaptive AMG (Method 1 in Section 3), as well as Rayleigh quotients for prototype vectors used, as a function of number of adaptive setup cycles performed.

# of setup cycles	1	2	3	4	5	6	Classical
$\overline{\rho}$	0.89	0.73	0.13	0.094	0.095	0.095	0.095
ρ_{100}	0.99	0.81	0.15	0.11	0.11	0.11	0.11
RQ	4.10	$3.79e{-2}$	$8.85e{-4}$	$2.23e{-5}$	$1.62e{-6}$	$1.85e{-7}$	0

Table III. Average and last-cycle convergence factors ($\overline{\rho}$ and ρ_{100}, resp.) using adaptive AMG for interpolation along edges and BoxMG for interpolation to cell centers (Method 2 in Section 3), as well as Rayleigh quotients for prototype vectors used, as a function of number of adaptive setup cycles performed.

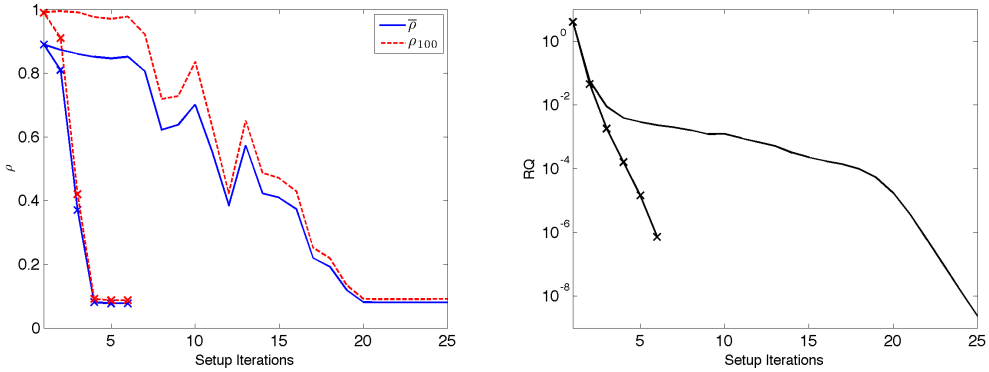
# of setup cycles	1	2	3	4	5	6	Classical
$\overline{\rho}$	0.89	0.69	0.098	0.091	0.091	0.091	0.092
ρ_{100}	0.98	0.76	0.11	0.10	0.10	0.10	0.10
RQ	4.10	$3.59e{-2}$	$4.87e{-4}$	$4.74e{-6}$	$2.24e{-7}$	$2.09e{-8}$	0

Methods 2 and 3 in Section 3. Table III shows results for one of these variants, using adaptive AMG for interpolation along the coarse-grid mesh edges and BoxMG for interpolation to coarse-grid cell centers (Method 2 in Section 3). These results are quite comparable to those in Table II, both in terms of the asymptotic convergence rates (which are slightly better than using adaptive AMG alone) and the Rayleigh Quotients of the prototype vectors. This slight improvement in performance may be due to the use of the adaptive AMG interpolation coefficients in the definition of the BoxMG interpolation coefficients for the cell centers, ensuring that $(AP\mathbf{e}^C)_{i,j} = 0$ for a fine-grid node, (i,j), that is at the center of a coarse-grid element. For the singular problem considered here, this may allow better recovery of the true null-space vector from the adaptive process than the fit by adaptive AMG, which gives a misfit that depends on the fine-grid values of $A\mathbf{z}$ [6].

In contrast, using adaptive BoxMG interpolation along coarse-grid mesh edges and adaptive AMG for coarse-grid cell centers (Method 3 in Section 3) yields an adaptive process that converges much more slowly than any of the cases considered above. Figure 5 shows the convergence of the adaptive process, measured in terms of the obtained V(1,1)-cycle convergence factors and the Rayleigh Quotients of the evolving prototype, over 25 adaptive setup cycles. While the convergence of the Rayleigh Quotient is almost monotonic (there is a slight increase in the RQ between the ninth and tenth adaptive setup cycles, but it is otherwise always decreasing), it decreases at a much slower rate than is achieved with the other interpolation approaches. The reason for this is not immediately apparent, but may be due to poor compatibility between the adaptive BoxMG approach along edges (which includes collapsing to the diagonal of some fine-fine connections that would be considered to be strong by AMG) and the adaptive AMG approach that interpolates to the cell centers completely independently of the interpolation for the edges. Coupled with this slow convergence of the prototype vector is an irregular convergence of the asymptotic convergence factors of the resulting V-cycles. Such irregular convergence highlights a current downside in the adaptive multigrid methodology, that reduction of the Rayleigh Quotient of a prototype vector does not guarantee improvement in the asymptotic convergence of the resulting multigrid V-cycle. Despite these issues, however, the adaptive process does converge, after about 20 cycles, to an interpolation operator that yields $\overline{\rho} = 0.080$ and $\rho_{100} = 0.090$, values that are very similar to those achieved by classical BoxMG and by the adaptive BoxMG method, shown in Table I.

These examples highlight the use of the Rayleigh Quotient of the evolving prototype as a potential measure of the convergence of the adaptive cycle. While reduction of the Rayleigh

Prepared using nlaauth.cls
DOI: 10.1002/nla



(a) $\overline{\rho}$ and ρ_{100} (b) Rayleigh quotients of prototype vectors

Figure 5. Average and last-cycle convergence factors ($\overline{\rho}$ and ρ_{100}, resp.) using adaptive BoxMG for interpolation along edges and adaptive AMG for interpolation to cell centers (Method 3 in Section 3), as well as Rayleigh quotients for prototype vectors used, as a function of number of adaptive setup cycles performed. For comparison, lines marked with the $\times$ symbols give results from Table I for adaptive BoxMG.

Quotient does not always lead to a reduction in the resulting multigrid V-cycle performance, optimal V-cycle performance does result from uniformly small values of the Rayleigh Quotient. In the above tables and figure, we see that the estimates of the asymptotic convergence factors are uniformly substandard if the Rayleigh Quotient of the prototype is much larger than $O(10^{-4})$, while they are uniformly optimal for smaller values. For non-singular problems, this in line with what was observed in early experiments with the adaptive methodology [53]; for the singular problem considered here, the relevant tolerance is likely to depend on the second-smallest eigenvalue, which is expected to be about $\left(\frac{1}{64}\right)^2 \approx 2.44 \times 10^{-4}$.

4.1.3. Dirichlet and Mixed Boundary Conditions In order to test the robustness of the adaptive BoxMG approach to symmetric diagonal scalings, as predicted in Theorem 1, we switch to considering the non-singular problem with full Dirichlet boundary conditions. This is to avoid confusion between the effects of the adaptive process and those of the additional projections included for the singular full-Neumann case. Table IV shows results for three different definitions of the scaling matrix, S: the unscaled case, $S = I$, choosing S so that the resulting scaled matrix, $\hat{A} = SAS$ has all ones on the diagonal, and choosing S randomly, by taking $s_{(i,j),(i,j)} = 10^{u_{i,j}}$ where the parameters $u_{i,j}$ are independently chosen from a uniform distribution on $[0, 1]$. We compare adaptive BoxMG with two variants of classical BoxMG, the switched version (described in Section 2.1) and the constant-preserving interpolation that forces each row in interpolation to have coefficients that sum to one. For these tests, we continue to consider a problem with random initial guess and zero right-hand side, so that we can effectively measure approximations to the asymptotic convergence factor of the multigrid cycle.

Results for the unscaled case in Table IV already show different behavior than those in the full Neumann case (cf. Table I). In particular, while the adaptive BoxMG process is quick to converge, it does not converge to a solver that is comparable to the classical BoxMG performance. This is likely due to the treatment of the Dirichlet boundary conditions in the system matrix. While AMG (and adaptive AMG) automatically eliminate any Dirichlet boundary points included in the discretized operator, BoxMG does not and, instead, retains some boundary points on all coarse grids. Thus, in contrast to the experiments with adaptive AMG in [6], there is a distinct difference in the performance of adaptive BoxMG for the Dirichlet and Neumann problems. Similar difficulties arise when forcing classical BoxMG to

DOI: 10.1002/nla

Table IV. Average and last-cycle convergence factors ($\overline{\rho}$ and ρ_{100}, resp.) using adaptive BoxMG interpolation for scaled versions of the periodic test problem with Dirichlet boundary conditions.

# of setup cycles	Unscaled		Unit Diagonal		Random scaling	
	$\overline{\rho}$	ρ_{100}	$\overline{\rho}$	ρ_{100}	$\overline{\rho}$	ρ_{100}
1	0.906	0.993	0.900	0.994	0.903	0.995
2	0.756	0.831	0.796	0.886	0.878	0.977
3	0.229	0.252	0.240	0.261	0.386	0.423
4	0.204	0.227	0.211	0.227	0.203	0.228
5	0.204	0.227	0.211	0.227	0.202	0.227
Classical	0.0599	0.0668	0.898	0.992	0.900	0.9996
Constant-Preserving	0.635	0.700	0.900	0.998	0.900	0.9997

Table V. Iteration counts for classical and adaptive BoxMG to reduce the L2 norm of the residual by a relative factor of 10^{14} for the scaled periodic permeability problem with combined Dirichlet and Neumann boundary conditions. > 500 denotes a run that did not meet this criterion in 500 iterations.

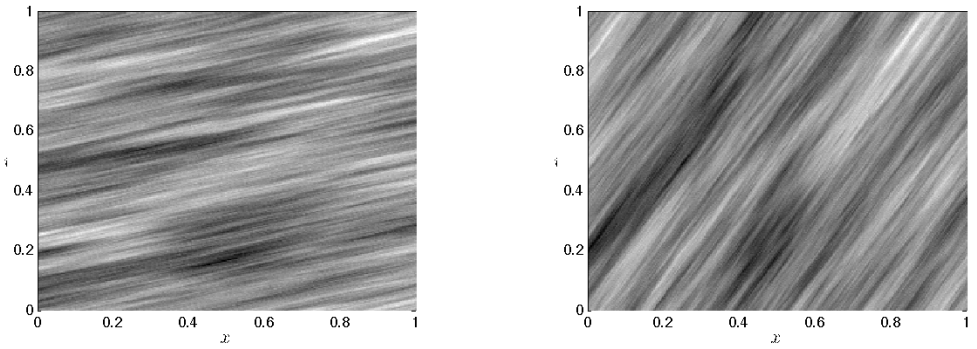
# of setup cycles	1	2	3	4	5	6	Classical
Unscaled	> 500	240	28	12	12	12	11
Unit Diagonal	> 500	370	39	18	17	17	> 500
Random Scaling	> 500	> 500	46	14	13	13	> 500

make use of the constant-preserving interpolation, which leads to much poorer performance than either classical (switched-interpolation) BoxMG or the adaptive BoxMG process. The adaptive process shows its advantages (and verifies Theorem 1) for the two non-trivially scaled examples, which both lead to near-total failure of the classical BoxMG approaches, while the adaptive approach recovers essentially the same convergence factor (both averaged, $\overline{\rho}$, and last cycle, ρ_{100}) in all cases.

Finally, we consider a problem with mixed Dirichlet and Neumann boundary conditions and a nonzero right-hand side vector. For this problem, we again consider the second-order form of Darcy's Law, $-\nabla \cdot \mathcal{K}(\mathbf{x})\nabla p(\mathbf{x}) = 0$, with homogeneous Neumann boundary conditions, $(\mathcal{K}(\mathbf{x})\nabla p(\mathbf{x})) \cdot \mathbf{n}(\mathbf{x}) = 0$, for points, $\mathbf{x}$, on the top and bottom edges of the square, where $\mathbf{n}(\mathbf{x})$ is the outward unit normal vector at $\mathbf{x}$. Dirichlet boundary conditions are imposed on the left and right edges of the domain, with fixed pressure, $p(\mathbf{x}) = 1$ on the left edge and $p(\mathbf{x}) = 0$ on the right edge. Thus, instead of computing the approximations ($\overline{\rho}$ and ρ_{100}) to the asymptotic convergence factors, we simply compute the number of iterations needed to reach convergence, defined as a reduction in the L2 norm of the residual by a relative factor of 10^{14}.

Table V shows iteration counts needed for the classical and adaptive BoxMG algorithms to achieve these reductions for the same scalings as considered in Table IV. For these problems, for each choice of S, we apply the standard BoxMG V(1,1) cycle based on adaptive (or classical) BoxMG interpolation to the problem $SAS\mathbf{y} = S\mathbf{b}$. Thus, the stopping criterion for each problem is somewhat different, given the effect of the scaling by S on the L2 norm of the residual; this is seen in the similar, but not identical iteration counts in Table V. As expected, the classical BoxMG approach is only successful for the unscaled problem, while V-cycles based on only one or two adaptive setup cycles are also slow to converge. For all three problems, reasonable solvers are achieved with three setup cycles, while there is only small (if any) improvement beyond the fourth setup cycle.

Prepared using nlaauth.cls
DOI: 10.1002/nla



(a) Principle axis of 15° (b) Principle axis of 60°

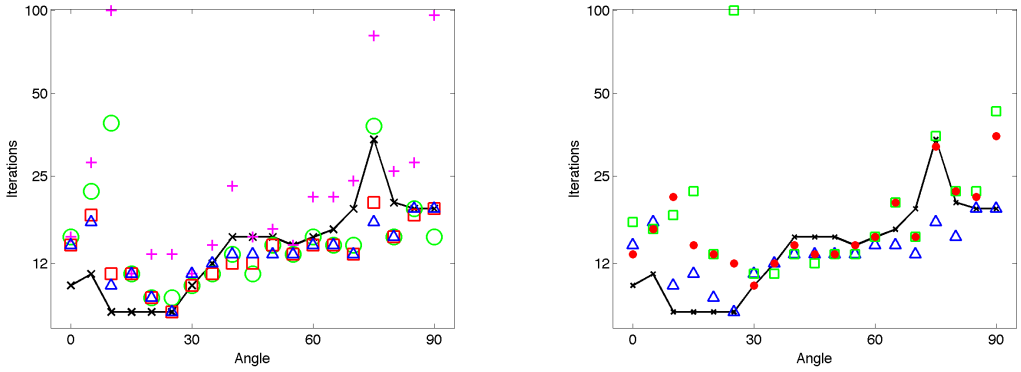
Figure 6. Geostatistical permeability fields, $\mathcal{K}(\mathbf{x})$, with principle axes of anisotropy of 15° (left) and 60° (right). Permeability values range from approximately 10^{-3} (white) to 10^{3} (black).

4.2. Geostatistical Examples

We now apply the adaptive BoxMG and AMG algorithms to a much more difficult class of heterogeneous diffusion problems, with much more realistic structures for $\mathcal{K}(\mathbf{x})$. To generate $\mathcal{K}(\mathbf{x})$, we use the geostatistical techniques implemented in the GSLIB software package [54]. We select a fine scale of 256×256 elements and, on each element, we generate a scalar constant permeability value. These values are selected by choosing a principle axis of statistical anisotropy, at 5° increments between 0 and 90 degrees relative to the positive x-axis, and generating a permeability field such that $\log_{10}(\mathcal{K}(\mathbf{x}))$ is normally distributed with mean 0, variance 4, and with correlation lengths of 0.8 along the principle axis and 0.04 in the direction orthogonal to this axis. Figure 6 shows grayscale images of two of these permeability fields, corresponding to principle axes of anisotropy of 15° and 60° degrees. Note that the fields do not correspond to simple rotations of one-another. An independent field is generated for each angle.

Iteration counts for solving the discretization of $-\nabla \cdot \mathcal{K}(\mathbf{x})\nabla p(\mathbf{x}) = 0$ with the same mixed Dirichlet and Neumann boundary conditions as in Section 4.1.3 are plotted in Figure 7. Here, solution is taken to mean a reduction in the L2 norm of the residual by a relative factor of 10^{10}, and a maximum of 100 iterations are allowed. As a benchmark, we first consider the performance of classical BoxMG (with the switched version of interpolation). Classical BoxMG successfully solves all of these realizations within 100 iterations, but the actual iteration counts are fairly angle-dependent, with relatively fewer iterations for smaller angles and more iterations for larger angles, peaking at 75° with 34 iterations. For the adaptive BoxMG tests, we use V(2,2) setup cycles (because V(1,1) setup cycles showed poor performance for some angles). While a single V(2,2) setup cycle never yielded a successful solver, two cycles were sufficient for all but one angle (10°). Four or five adaptive BoxMG V(2,2) setup cycles were sufficient to achieve steady-state of the adaptive solvers, with iteration counts similar to classical BoxMG. Notably, there is less angle-dependent variation in the iteration counts for these solvers, with variation from 8 to 19 iterations to solution for five setup cycles, compared to the maximum of 34 iterations needed by classical BoxMG. Adaptive AMG performs less well than adaptive BoxMG; with six V(2,2) setup cycles, adaptive AMG is notably worse than adaptive BoxMG, particularly for large angles. While increasing the work in the adaptive AMG setup process to six V(4,4) setup cycles improves these results somewhat (particularly the outlying data point where the V(2,2) setup approach failed for 25°), it is clear that adaptive BoxMG generally outperforms adaptive AMG for these problems. This is not surprising, however, as BoxMG (and its adaptive counterpart) is designed to treat these strongly varying permeability fields

DOI: 10.1002/nla



(a) Adaptive BoxMG (b) Adaptive AMG

Figure 7. Iteration counts for solving problems with geostatistical media. In black in both plots, marked by the solid lines and × symbol, are the iteration counts for classical BoxMG to reduce the residual by a relative factor of 10^{10}. At left are the results for adaptive BoxMG; + symbols (in magenta) mark 2 adaptive V(2,2) setup cycles, ◯ symbols (in green) mark 3 adaptive setup cycles, □ symbols (in red) mark 4 adaptive V(2,2) setup cycles, and △ symbols (in blue) mark 5 adaptive V(2,2) setup cycles. At right are the results for adaptive AMG, with the classical BoxMG and last adaptive BoxMG (with 5 adaptive V(2,2) setup cycles) provided for reference; □ symbols (in green) mark 6 adaptive V(2,2) setup cycles, while • symbols (in red) mark 6 adaptive V(4,4) setup cycles. Solve phases that failed to converge within 100 iterations are marked as 100 iterations.

with structured coarsening, while both classical and adaptive AMG would naturally treat these fields with unstructured coarsening schemes.

4.3. Vacuum Boundary Conditions

A powerful feature of the classical BoxMG code is its robustness with respect to grid dimension and boundary conditions. Specifically, BoxMG uses the heuristic switch described in Section 2.1 to treat grids that are not of dimension $2^N + 1$ efficiently, and to ensure that the performance of Neumann, Dirichlet, and Robin (mixed) boundary conditions are comparable. In this section, we consider a simple problem with a vacuum boundary condition on one side of the domain that illustrates the performance of this heuristic switch. In addition, this example demonstrates the ability of the adaptive BoxMG algorithm to perform as well as or better than the heuristic switch. We use the adaptive setup cycle described in Section 2.3, with 5 V(2,1) setup cycles, and a random vector for the initial prototype. To quantify the performance of each method we report the average convergence factor, $\overline{\rho}$, which is computed as the average over 20 iterations of the resulting multigrid cycle.

The vacuum boundary condition is a specific Robin boundary condition that arises in the P_1 approximation of the neutron transport equation. Physically, it represents a boundary on which neutrons can leave the domain, but do not enter. Here, we consider a standard 5-point vertex-based discretization of the example introduced in [45]:

$$\begin{aligned} -\nabla \cdot (\mathcal{K}\nabla p) &= 0 \text{ in } \Omega = (0, X_R) \times (0, Y_T)\,, \\ (\mathcal{K}\nabla p) \cdot \mathbf{n} &= 0 \text{ on } (\{0\} \times (0, Y_T)) \cup ((0, X_R) \times \{0\}) \cup (\{X_R\} \times (0, Y_T))\,, \\ (\mathcal{K}\nabla p) \cdot \mathbf{n} + \frac{1}{2} p &= 0 \text{ on } (0, X_R) \times \{Y_T\}\,, \end{aligned}$$

where $\mathcal{K}$ is a diagonal diffusion tensor. The finite-difference discretization leads to a symmetric M-matrix that has a diagonally dominant stencil along the top boundary, $y = Y_T$. The mesh spacing, h, and the y-component of the diffusion tensor, $\mathcal{K}_{yy}$, control the strength of this

DOI: 10.1002/nla

Table VI. Convergence rates for the anisotropic problem with grids of dimension $2^N + 1$ are shown. In this case, the diagonally dominant point remains on all coarser grids, and averaging interpolation is ineffective.

Grid Dimension	$\overline{\rho}^{cp}$	$\overline{\rho}^{avg}$	$\overline{\rho}^{sw}$	$\overline{\rho}^{adapt}$
17×17	0.01003	0.56821	0.01003	0.00985
33×33	0.03348	0.53651	0.03348	0.03604
65×65	0.04854	0.50340	0.04854	0.04716
129×129	0.04957	0.47413	0.04957	0.05049
257×257	0.04993	0.45050	0.04993	0.04977

diagonal dominance. To demonstrate the importance of the this seemingly benign diagonal dominance, we follow [45] and consider two variations of this problem.

First, we consider an anisotropic problem that demonstrates the failure of the averaging interpolation and the solid performance of the other formulations. In addition, this is the first demonstration of adaptive BoxMG with line relaxation. Consider the constant-coefficient anisotropic problem on the unit square defined by

$$\mathcal{K} = \begin{bmatrix} 1 & 0 \\ 0 & 100 \end{bmatrix}, \quad X_R = Y_T = 1\,.$$

As noted earlier, combining anisotropic diffusion with standard coarsening dictates that we use y-line relaxation. The failure of averaging interpolation arises on grids that are dimension $2^N + 1$ as the diagonally dominant stencil remains on all coarse grids. Specifically, the averaging interpolation along the top boundary introduces a significant error, due largely to the strength of $\mathcal{K}_{yy}$, and this error is immediately propagated to the neighboring fine-grid points located on the interior of coarse-grid cells. This failure is shown in Table VI, where the averaging interpolation yields $\overline{\rho} \approx 0.5$, while other formulations all have $\overline{\rho} \approx 0.05$. We also note that the adaptive cycling procedure used to construct the adaptive BoxMG interpolation performs well with line relaxation, producing convergence rates nearly identical to the classical BoxMG code.

The second example is an isotropic problem that demonstrates the failure of the constant-preserving interpolation, as well as the ability of the adaptive BoxMG interpolation to outperform classical BoxMG in some cases. We continue to use y-line relaxation to further demonstrate the capability of the adaptive setup to use non-point smoothers. This constant-coefficient isotropic problem uses a large domain and is defined by

$$\mathcal{K} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad X_R = Y_T = 128\,.$$

With the large domain, we have $h \gg \mathcal{K}_{xx} = \mathcal{K}_{yy}$, making the stencil diagonally dominant on the top boundary. We consider two sequences of grids. In the first sequence, we consider grids with dimensions 2^N, such that the last point in each direction (right,top) is lost on each successively coarser grid. This leads to extrapolation at the right and top boundaries. The failure of constant-preserving interpolation for this hierarchy of grids is shown in Table VII, where $\overline{\rho}^{cp} \approx 0.5$ while the other formulations yield the expected $\overline{\rho} \approx 0.05$. Next, we consider grids of dimension $2^N + 1$ for this problem. In this case, like the anisotropic problem above, the boundary point remains on all coarser grids. So, given identical grid hierarchies, it is interesting to examine the similarities and differences between this problem and the anisotropic one. As before, the constant-preserving interpolation does very well, with $\overline{\rho}^{cp} \approx 0.055$ (see Table VIII). This success is primarily because only points on the boundary have diagonally dominant stencils, and constant-preserving interpolation is most appropriate there. However, in this case, the averaging interpolation does not fail and, instead, gives $\overline{\rho}^{avg} \approx 0.12$. Here, unlike the strongly anisotropic case, the error introduced using averaging interpolation along

Prepared using nlaauth.cls
DOI: 10.1002/nla

Table VII. Convergence rates for the isotropic, large domain, problem are shown. Grids of dimension 2^N lead to extrapolation at the right and top boundaries.

Grid Dimension	$\overline{\rho}^{cp}$	$\overline{\rho}^{avg}$	$\overline{\rho}^{sw}$	$\overline{\rho}^{adapt}$
16×16	0.47194	0.05503	0.05503	0.05543
32×32	0.50807	0.05479	0.05479	0.05576
64×64	0.51016	0.05558	0.05558	0.05524
128×128	0.49309	0.05505	0.05505	0.05483
256×256	0.47210	0.05506	0.05506	0.05511

Table VIII. Convergence rates for the isotropic, large domain, problem are shown. The grids are of dimension $2^N + 1$ and, hence, keep the diagonally dominant point on all coarser grids (no extrapolation at top and right boundaries).

Grid Dimension	$\overline{\rho}^{cp}$	$\overline{\rho}^{avg}$	$\overline{\rho}^{sw}$	$\overline{\rho}^{adapt}$
17×17	0.05735	0.06901	0.06098	0.05626
33×33	0.08535	0.08976	0.07764	0.05572
65×65	0.05492	0.11529	0.10516	0.05542
129×129	0.05489	0.12839	0.12212	0.05490
257×257	0.05492	0.12076	0.10895	0.05506

the top boundary is quite small. Moreover, based on the strength of the diagonal dominance, the heuristic switch tends to use averaging interpolation and, hence, leads to $\overline{\rho}^{sw} \approx 0.12$ as well. In contrast, the adaptive interpolation, which is generated by the adaptive setup cycle, yields $\overline{\rho}^{adapt} \approx 0.055$. Thus, this example demonstrates that the adaptive interpolation may outperform the heuristic switch that is used in classical BoxMG.

5. CONCLUSIONS

This paper presents two main contributions: the extension of the adaptive multigrid framework to a structured, robust multigrid algorithm, BoxMG, and insight into the relationship between the interpolation operators used in classical algebraic multigrid (AMG) and BoxMG. For the former, we show, both analytically and numerically, that the adaptive BoxMG approach inherits the diagonal-scaling invariance properties of the adaptive AMG algorithms. Further, the adaptive BoxMG approach naturally treats highly heterogeneous media and a wide variety of boundary conditions without the logical complexity needed for robustness in the classical BoxMG approach; it also naturally complements the use of line relaxation strategies for anisotropic problems. For the latter, we present a simple relationship between the classical AMG interpolation operators on structured grids and those used by BoxMG and investigate hybrid approaches based on these two points of view. Taken together, these results lead to improved insight into and performance from the robust structured-grid context of BoxMG, which has the potential to be very useful in the many-core and accelerated architecture paradigms, where the computational advantages of structured-grid approaches over AMG are significant.

REFERENCES

1. MacLachlan SP, Tang JM, Vuik C. Fast and robust solvers for pressure-correction in bubbly flow problems. *J. Comput. Phys.* 2008; **227**(23):9742–9761.

DOI: 10.1002/nla

2. Brandt A. Multi–level adaptive technique (MLAT) for fast numerical solution to boundary value problems. *Proceedings of the Third International Conference on Numerical Methods in Fluid Mechanics*, *Lecture Notes in Physics*, vol. 18, Cabannes H, Teman R (eds.), Springer–Verlag: Berlin, 1973; 82–89.
3. Brandt A. Multi–level adaptive solutions to boundary–value problems. *Math. Comp.* 1977; **31**:333–390.
4. Brezina M, Falgout R, MacLachlan S, Manteuffel T, McCormick S, Ruge J. Adaptive smoothed aggregation (αSA). *SIAM J. Sci. Comput.* 2004; **25**(6):1896–1920.
5. Brezina M, Falgout R, MacLachlan S, Manteuffel T, McCormick S, Ruge J. Adaptive smoothed aggregation (αSA) multigrid. *SIAM Rev.* 2005; **47**(2):317–346.
6. Brezina M, Falgout R, MacLachlan S, Manteuffel T, McCormick S, Ruge J. Adaptive algebraic multigrid. *SIAM J. Sci. Comput.* 2006; **27**:1261–1286.
7. Brandt A, Brannick J, Kahl K, Livshits I. Bootstrap AMG. *SIAM Journal on Scientific Computing* 2011; **33**(2):612–632.
8. H De Sterck, Manteuffel TA, McCormick SF, Nguyen Q, Ruge J. Multilevel adaptive aggregation for Markov chains, with application to web ranking. *SIAM J. Sci. Comput.* 2008; **30**(5):2235–2262.
9. Borzì A. High-order discretization and multigrid solution of elliptic nonlinear constrained optimal control problems. *J. Comput. Appl. Math.* 2007; **200**(1):67–85.
10. Börgers C, MacLachlan S. An angular multigrid method for monoenergetic particle beams in Flatland. *J. Comp. Phys.* 2010; **229**:2914–2931.
11. Brezina M, Manteuffel T, McCormick S, Ruge J, Sanders G. Towards adaptive smoothed aggregation (αSA) for nonsymmetric problems. *SIAM J. Sci. Comput.* 2010; **32**:14–39.
12. Haber E, MacLachlan S. A fast method for the solution of the Helmholtz equation. *Journal of Computational Physics* 2011; **230**(12):4403 – 4418.
13. Ritter D, Stürmer M, Rüde U. A fast-adaptive composite grid algorithm for solving the free-space Poisson problem on the cell broadband engine. *Numerical Linear Algebra with Applications* 2010; **17**(2-3):291–305.
14. Bell N, Garland M. Implementing sparse matrix-vector multiplication on throughput-oriented processors. *SC '09: Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis*, ACM: New York, NY, USA, 2009; 1–11.
15. Minden V, Smith B, Knepley MG. Preliminary implementation of PETSc using GPUs. *Proceedings of the 2010 International Workshop of GPU Solutions to Multiscale Problems in Science and Engineering*, 2010.
16. Alcouffe RE, Brandt A, Dendy JE, Painter JW. The multigrid method for the diffusion equation with strongly discontinuous coefficients. *SIAM J. Sci. Stat. Comput.* 1981; **2**:430–454.
17. Dendy JE. Black box multigrid. *J. Comput. Phys.* 1982; **48**:366–386.
18. Ruge JW, Stüben K. Algebraic multigrid (AMG). *Multigrid Methods*, *Frontiers in Applied Mathematics*, vol. 3, McCormick SF (ed.). SIAM: Philadelphia, PA, 1987; 73–130.
19. Brandt A, McCormick SF, Ruge JW. Algebraic multigrid (AMG) for automatic multigrid solution with application to geodetic computations. *Technical Report*, Institute for Computational Studies, Colorado State University 1982.
20. Brandt A, McCormick SF, Ruge JW. Algebraic multigrid (AMG) for sparse matrix equations. *Sparsity and Its Applications*, Evans DJ (ed.). Cambridge University Press: Cambridge, 1984.
21. Dendy JE. Black box multigrid for nonsymmetric problems. *Appl. Math. Comput.* 1983; **13**:261–283.
22. Dendy JE. Black box multigrid for systems. *Appl. Math. Comput.* 1986; **19**:57–74.
23. Dendy, Jr JE. Two multigrid methods for three-dimensional problems with discontinuous and anisotropic coefficients. *SIAM J. Sci. Statist. Comput.* 1987; **8**(5):673–685.
24. Dendy JE. Black box multigrid for periodic and singular problems. *Appl. Math. Comput.* 1988; **25**:1–10.
25. Alber DM, Olson LN. Parallel coarse-grid selection. *Numerical Linear Algebra with Applications* 2007; **14**(8):611–643.
26. De Sterck H, Yang UM, Heys JJ. Reducing complexity in parallel algebraic multigrid preconditioners. *SIAM J. Matrix Anal. Appl.* 2006; **27**(4):1019–1039.
27. De Sterck H, Falgout RD, Nolting JW, Yang UM. Distance-two interpolation for parallel algebraic multigrid. *Numer. Linear Algebra Appl.* 2008; **15**(2-3):115–139.
28. Livne O. Coarsening by compatible relaxation. *Numer. Linear Alg. Appl.* 2004; **11**(2):205–227.
29. Brannick JJ, Falgout RD. Compatible relaxation and coarsening in algebraic multigrid. *SIAM Journal on Scientific Computing* 2010; **32**(3):1393–1416.
30. MacLachlan S, Saad Y. A greedy strategy for coarse-grid selection. *SIAM J. Sci. Comput.* 2007; **29**(5):1826–1853.
31. Austin TM, Trew ML, Pullan AJ. Solving the cardiac bidomain equations for discontinuous conductivities. *IEEE Trans Biomed Eng* July 2006; **53**(7):1265–1272.
32. Jiang Y, Pjesivac J, Cantrell C, Freyer J. A multiscale model for avascular tumor growth 2005. Preprint: LA-UR-04-5498.
33. Aruliah DA, Ascher UM. Multigrid preconditioning for Krylov methods for time-harmonic Maxwell's equations in three dimensions. *SIAM Journal on Scientific Computing* 2002; **24**(2):702–718.
34. MacLachlan SP, Moulton JD. Multilevel upscaling through variational coarsening. *Water Resour. Res.* 2006; **42**.
35. Moulton J, Knapek S, Dendy J. Multilevel upscaling in heterogeneous porous media. *CNLS report*, Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, New Mexico 1999.
36. Moulton JD, Dendy JE, Hyman JM. The black box multigrid numerical homogenization algorithm. *J. Comput. Phys.* 1998; **141**:1–29.
37. Adams M, Brezina M, Hu J, Tuminaro R. Parallel multigrid smoothing: polynomial versus Gauss-Seidel. *J. Comput. Phys.* 2003; **188**:593–610.

DOI: 10.1002/nla

38. Brezina M, Tong C, Becker R. Parallel algebraic multigrids for structural mechanics. *SIAM J. Sci. Comput.* 2006; **27**(5):1534–1554.
39. Henson V, Yang U. BoomerAMG: a parallel algebraic multigrid solver and preconditioner. *Applied Numerical Mathematics* 2002; **41**:155–177.
40. Tuminaro RS, Tong C. Parallel smoothed aggregation multigrid: Aggregation strategies on massively parallel machines. *SuperComputing 2000 Proceedings*, Donnelley J (ed.), 2000.
41. Yang UM. Parallel algebraic multigrid methods—high performance preconditioners. *Numerical solution of partial differential equations on parallel computers*, *Lect. Notes Comput. Sci. Eng.*, vol. 51. Springer: Berlin, 2006; 209–236.
42. Brannick J, Brezina M, Keyes D, Livne O, Livshits I, MacLachlan S, Manteuffel T, McCormick S, Ruge J, Zikatanov L. Adaptive smoothed aggregation in lattice QCD. *Domain Decomposition Methods in Science and Engineering XVI*, *Lecture Notes in Computational Science and Engineering*, vol. 55. Springer, 2007; 505–512.
43. Babich R, Brannick J, Brower RC, Clark MA, Manteuffel TA, McCormick SF, Osborn JC, Rebbi C. Adaptive multigrid algorithm for the lattice Wilson-Dirac operator. *arXiv* Jan 2010; **hep-lat**.
44. Brannick J, Brower RC, Clark MA, Osborn JC, Rebbi C. Adaptive multigrid algorithm for lattice QCD. *Physical Review Letters* 2008; **100**(4):041601.
45. Dendy, Jr JE, Moulton JD. Black box multigrid with coarsening by a factor of three. *Numerical Linear Algebra with Applications* 2010; **17**(2-3):577–598.
46. P Vaněk, Mandel J, Brezina M. Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems. *Computing* 1996; **56**:179–196.
47. Brannick J, Frommer A, Kahl K, MacLachlan S. Adaptive reduction-based multigrid for nearly singular and highly disordered physical systems. *Electron. Trans. Numer. Anal.* 2010; **37**:276–295.
48. Chartier T, Orr D. Adaptive multigrid via subcycling on complementary grids 2007; In progress.
49. MacLachlan S, Manteuffel T, McCormick S. Adaptive reduction-based AMG. *Numer. Linear Alg. Appl.* 2006; **13**:599–620.
50. Chow E. An aggregation multilevel method using smooth error vectors. *SIAM J. Sci. Comput.* 2006; **27**:1727–1741.
51. Briggs WL, Henson VE, McCormick SF. *A Multigrid Tutorial*. SIAM Books: Philadelphia, 2000. Second edition.
52. Grauschopf T, Griebel M, Regler H. Additive multilevel-preconditioners based on bilinear interpolation, matrix dependent geometric coarsening and algebraic multigrid coarsening for second order elliptic PDEs. *Appl. Numer. Math.* 1997; **23**:63–96.
53. MacLachlan S. Improving robustness in multiscale methods. PhD Thesis, Univerity of Colorado at Boulder, Boulder, Colorado 2004.
54. Deutsch C, Journel A. *GSLIB, geostatistical software library and user's guide*. Second edn., Oxford University Press: Oxford, 1998.

DOI: 10.1002/nla