AN IMPROVED ALGEBRAIC MULTIGRID METHOD FOR SOLVING MAXWELL'S EQUATIONS

PAVEL B. BOCHEV[1,4], CHRISTOPHER J. GARASI[2,4], JONATHAN J. HU[3,4], ALLEN C. ROBINSON[2,4] AND RAYMOND S. TUMINARO[3,4]

Abstract. We propose two improvements to the Reitzinger and Schöberl algebraic multigrid (AMG) method for solving the eddy current approximations to Maxwell's equations. The main focus in the Reitzinger/Schöberl method is to maintain null space properties of the weak $\nabla\times\nabla\times$ operator on coarse grids. While these null space properties are critical, they are not enough to guarantee h-independent convergence of the overall multigrid method. We illustrate how the Reitzinger/Schöberl AMG method loses h-independence due to the somewhat limited approximation property of the grid transfer operators. We present two improvements to these operators that not only maintain the important null space properties on coarse grids but also yield significantly improved multigrid convergence rates. The first improvement is based on smoothing the Reitzinger/Schöberl grid transfer operators. The second improvement is obtained by using higher order nodal interpolation to derive the corresponding AMG interpolation operators. While not completely h-independent, the resulting AMG/CG method demonstrates improved convergence behavior while maintaining low operator complexity.

Key words. Maxwell's equations, eddy currents, AMG, multigrid, smoothed aggregation.

AMS subject classifications. 76D05, 76D07, 65F10, 65F30

1. Introduction.

We consider the solution to the 3D eddy current formulation of Maxwell's equations discretized with edge and face vectorial finite elements on arbitrary, unstructured hexahedral triangulations [3, 17]. The key difficulty is the large null space associated with the curl operator within Maxwell's equations. This large kernel requires special treatment within both the multigrid smoother and the multigrid grid transfers. It is for this reason that most classical algebraic multigrid methods fail on this set of equations. Reitzinger and Schöberl have proposed an algebraic multigrid (AMG) method for Maxwell's equations [11]. We take this method as our starting point and discuss two improvements to the grid transfer operators. We then show how these modifications lead to a significant reduction in the number of multigrid iterations while maintaining low operator complexity. Although the new multigrid algorithm is still mildly dependent on h (where h is the mesh spacing), the growth in iterations is quite small as the mesh is refined.

In §2 we give a brief description of the governing equations and the corresponding discrete system. In §3.1 we discuss the central difficulty of solving the discrete system. This is followed by a discussion of the Reitzinger/Schöberl AMG method in §3.2. Some initial numerical results highlight the lack of h-independence. Two new ideas are then presented in §4 for improving the grid transfers. One of these ideas is to smooth the Reitzinger/Schöberl grid transfer operators, motivated by the smoothed aggregation AMG method [15]. A second idea is based on using higher order nodal interpolation

[1]Sandia National Laboratories, Computational Math/Algorithms, Albuquerque, NM 87185-1110 (`pbboche@sandia.gov`).

[2]Sandia National Laboratories, Computational Physics R&D, Albuquerque, NM 87185-0819 (`cjgaras@sandia.gov, acrobin@sandia.gov`).

[3]Sandia National Laboratories, Computational Math/Algorithms, PO Box 969, MS 9217, Livermore, CA 94551 (`jhu@sandia.gov, rstumin@sandia.gov`)

[4]Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under contract DE-AC04-94-AL85000.

in the Reitzinger/Schöberl framework. Finally, numerical results and conclusions are given in §5 to demonstrate the effectiveness of the new procedures.

2. Governing Equations and Discretization. The physical model of interest is eddy currents in a single conduction region Ω in $\mathbf{R}^3$. The eddy current equations are obtained by neglecting the displacement current in the full Maxwell equations. This amounts to neglecting high frequency speed-of-light time scale electromagnetic waves in a conducting media. The resulting set of governing equations for the electromagnetic field is

$$\nabla \times \mathbf{H} = \mathbf{J} \quad \text{in } \Omega \tag{2.1}$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad \text{in } \Omega \tag{2.2}$$

$$\nabla \cdot \mathbf{B} = 0 \quad \text{in } \Omega \tag{2.3}$$

$$\nabla \cdot \mathbf{J} = 0 \quad \text{in } \Omega \tag{2.4}$$

where $\mathbf{H}$ is the magnetic field, $\mathbf{J}$ is the current density, $\mathbf{E}$ is the electric field, and $\mathbf{B}$ is the magnetic flux density. These fields are connected by the constitutive relations

$$\mathbf{B} = \mu \mathbf{H} \tag{2.5}$$

$$\mathbf{J} = \sigma \mathbf{E}, \tag{2.6}$$

where σ is the nonconstant electrical conductivity and μ is the permeability. Equation (2.1) is Ampere's theorem, (2.2) is Faraday's law, and (2.6) is Ohm's law. System (2.1)-(2.4) is closed by boundary conditions

$$\mathbf{n} \times \mathbf{E} = \mathbf{n} \times \mathbf{E}_{\mathrm{b}} \quad \text{and} \quad \mathbf{n} \cdot \mathbf{B} = \mathbf{n} \cdot \mathbf{B}_{\mathrm{b}} \quad \text{on } \Gamma^* \tag{2.7}$$

and

$$\mathbf{n} \times \mathbf{H} = \mathbf{n} \times \mathbf{H}_{\mathrm{b}} \quad \text{and} \quad \mathbf{n} \cdot \mathbf{J} = \mathbf{n} \cdot \mathbf{J}_{\mathrm{b}} \quad \text{on } \Gamma, \tag{2.8}$$

where Γ^* and Γ denote disjoint parts of the boundary $\partial\Omega$. We now give an overview of the discretization of the eddy current equations. For a more complete discussion, we refer the reader to [3]. First $\mathbf{H}$ and $\mathbf{J}$ are eliminated from the Ampere law (2.1) and the boundary condition (2.8) by (2.5)-(2.6). To discretize the resulting system we use an edge based finite element model for $\mathbf{E}_h$ and a face based finite element model for $\mathbf{B}_h$. The edge and face element spaces, denoted here by $\mathcal{W}^1$ and $\mathcal{W}^2$, respectively, are conforming subspaces of

$$H(\Omega, \mathbf{curl}) = \{\mathbf{u} \in \mathbf{L}^2(\Omega) | \nabla \times \mathbf{u} \in \mathbf{L}^2(\Omega)\}$$

and

$$H(\Omega, \mathrm{div}) = \{\mathbf{u} \in \mathbf{L}^2(\Omega) | \nabla \cdot \mathbf{u} \in \mathbf{L}^2(\Omega)\}.$$

The two spaces $H(\Omega, \mathbf{curl})$ and $H(\Omega, \mathrm{div})$ are part of the De Rham differential complex and as such they form an exact sequence [4, 5]. This means that curl maps $H(\Omega, \mathbf{curl})$ into the kernel of the divergence operator in $H(\Omega, \mathrm{div})$. Symbolically, this relation is denoted by the diagram

$$H(\Omega, \mathbf{curl}) \overset{\nabla\times}{\longmapsto} H(\Omega, \mathrm{div}) \overset{\nabla\cdot}{\longmapsto} \cdots . \tag{2.9}$$

The edge and face element spaces $\mathcal{W}^1$ and $\mathcal{W}^2$ employed here are constructed so that they obey the same exactness relation:

$$\mathcal{W}^1 \quad \overset{\nabla\times}{\longmapsto} \quad \mathcal{W}^2 \quad \overset{\nabla\cdot}{\longmapsto} \cdots . \tag{2.10}$$

The details about these spaces and their definition can be found in [3]. To derive the discrete eddy current model we replace the time derivative in (2.2) by a finite difference and use the resulting discrete Faraday law

$$\mathbf{B}_h^{n+1} = \mathbf{B}_h^n - \Delta t \nabla \times \mathbf{E}_h^{n+1}$$

to eliminate $\mathbf{B}_h^{n+1}$ from Ampere's law (2.1). After writing (2.1) in weak form we obtain an equation in terms of $\mathbf{E}_h^{n+1}$ only:

$$\begin{aligned} &\int_\Omega \sigma \mathbf{E}_h^{n+1} \cdot \hat{\mathbf{E}}_h + \frac{\Delta t}{\mu}\Big(\nabla \times \mathbf{E}_h^{n+1}\Big) \cdot \Big(\nabla \times \hat{\mathbf{E}}_h\Big) d\Omega \\ &= \int_\Omega \frac{1}{\mu} \mathbf{B}_h^n \cdot \Big(\nabla \times \hat{\mathbf{E}}_h\Big) d\Omega + \int_\Gamma \Big(\mathbf{n} \times \mathbf{H}_{\mathrm{b}}\Big) \cdot \hat{\mathbf{E}}_h d\Gamma \qquad \forall \hat{\mathbf{E}}_h \in \mathcal{W}^1. \end{aligned} \tag{2.11}$$

We write (2.11) compactly as

$$K^{(e)} u = f \tag{2.12}$$

and

$$K^{(e)} = K^{(e)}_{mass} + K^{(e)}_{curl}.$$

The superscript (e) refers to edge elements, and $K^{(e)}_{mass}$ and $K^{(e)}_{curl}$ are the first and second terms on the left hand side of (2.11).

For the proper discussion of algebraic multigrid we need another component of the aforementioned DeRham complex. It involves the already defined space $H(\Omega, \mathbf{curl})$ and the domain of the gradient operator:

$$H(\Omega, \mathrm{grad}) = \{\phi \in L^2(\Omega) | \nabla\phi \in \mathbf{L}^2(\Omega)\}. \tag{2.13}$$

These two spaces form an exact sequence

$$H(\Omega, \mathrm{grad}) \overset{\nabla}{\longmapsto} H(\Omega, \mathbf{curl}) \overset{\nabla\times}{\longmapsto} \cdots \tag{2.14}$$

meaning that the gradient operator maps $H(\Omega, \mathrm{grad})$ into the kernel of the curl operator in $H(\Omega, \mathbf{curl})$. A fundamental property of the finite element spaces considered in [3] is that the nodal space, which is a conforming subspace of $H(\Omega, \mathrm{grad})$ and is denoted here by $\mathcal{W}^0$, forms an exact sequence with the edge element space $\mathcal{W}^1$ introduced earlier:

$$\mathcal{W}^0 \quad \overset{\nabla}{\longmapsto} \quad \mathcal{W}^1 \quad \overset{\nabla\times}{\longmapsto} \cdots . \tag{2.15}$$

3. Multigrid Overview. Multigrid methods (e.g, [9], [14], [7]) are among the most efficient iterative algorithms for solving $Ax = b$, the linear systems associated with elliptic partial differential equations. The basic idea is to capture errors by utilizing multiple resolutions in the iterative scheme. High energy (or oscillatory) components are effectively reduced through a simple smoothing procedure, while the

```
// Solve A_k u_k = b_k
procedure multilevel(A_k, b_k, u_k, k)
    u_k = S_k(A_k, b_k, u_k);
    if ( k ≠ Nlevel)
        r_k = b_k − A_k u_k ;
        A_{k+1} = P_k^T A_k P_k;
        u_{k+1} = 0;
        multilevel(A_{k+1}, P_k^T r_k, u_{k+1}, k + 1);
        u_k = u_k + P_k u_{k+1};
        u_k = S_k(A_k, b_k, u_k);
```

FIG. 3.1. *Multigrid V cycle consisting of 'Nlevel' grids to solve $A_1 u = b$.*

low energy (or smooth) components are tackled using an auxiliary lower resolution version of the problem (coarse grid). The idea is applied recursively on the next coarser level. An example multilevel iteration is given in Figure 3.1 to solve

$$A_1 u_1 = b_1. \tag{3.1}$$

The two operators needed to specify the multigrid method fully are the smoothers, S_k's, and the grid transfers, P_k's. Note that P_k is an interpolation operator that transfers grid information from level k to level $k-1$. The key to fast convergence is the complementary nature of these two operators. That is, errors not reduced by S_k must be well interpolated by P_k. In algebraic multigrid, only A_1 and b_1 are given and so the S_k's and P_k's must be deduced from purely algebraic principles. The focus of this paper is the determination of the P_k's.

3.1. AMG and Maxwell's Equations. To understand the difficulties in solving Maxwell's equations, consider the sequence (2.14) and the exactness relation

$$\nabla \times (\nabla \phi) = 0 \tag{3.2}$$

which states that the gradient of any differentiable scalar function lies within the kernel of the curl operator. The discrete exact sequence (2.15) reproduces the same property for finite element functions. As a result, the discrete null space of $K^{(e)}_{curl}$ can be described by

$$\mathcal{N}_{curl} = \{\nabla\phi | \phi \in \mathcal{W}^0\}.$$

It is important to notice several items about $\mathcal{N}_{curl}$. The first item is that an explicit basis for $\mathcal{N}_{curl}$ is obtained by taking the gradient of each nodal basis function defining $\mathcal{W}^0$. The second is that the dimension of $\mathcal{N}_{curl}$ is the same as the dimension of $\mathcal{W}^0$ (which is approximately the number of mesh nodes). The third is that $\mathcal{N}_{curl}$ contains both high and low frequency components (i.e. the gradients of smooth functions in $\mathcal{W}^0$ are smooth functions, the gradients of oscillatory functions in $\mathcal{W}^0$ are oscillatory). Thus, $K^{(e)}_{mass}$ is small when σ is small, and so $K^{(e)}$ effectively has a large near null space that contains both high and low frequency components.

Geometric multigrid methods have been successfully applied to (2.12) [10]. The key idea is to construct the smoother carefully so that high frequency errors within both $\mathcal{N}_{curl}$ and $\mathcal{N}^{\perp}_{curl}$ are damped. Specifically, a standard smoother is first applied

$$\begin{tikzcd}
\mathcal{W}_k^0 \arrow[r, "\nabla"] & \mathcal{N}_{curl,k} \\
\mathcal{W}_{k+1}^0 \arrow[u, "P_k^{(n)}"] \arrow[r, "\nabla"] & \mathcal{N}_{curl,k+1} \arrow[u, "P_k^{(e)}"']
\end{tikzcd}$$

FIG. 3.2. *Commuting diagram for two levels.*

to $K^{(e)}$. Then, a correction equation is formed and projected onto $\mathcal{N}_{curl}$. Standard smoothing is applied to the projected equation, and the correction is added into the previous solution estimate. Alternative smoothers have also been proposed by [2]. There are two key points to note. First, this smoother requires the construction of a basis for $\mathcal{N}_{curl}$ on all of the grid levels. Second, standard geometric grid transfers properly approximate smooth components (in $\mathcal{N}_{curl}$ and $\mathcal{N}_{curl}^{\perp}$). In contrast, special care is required to construct the AMG coarse grid basis functions.

3.2. The Reitzinger Schöberl AMG method. The key aspect of the Reitzinger and Schöberl algebraic multigrid method is to ensure that the commuting diagram given in Figure 3.2 holds. The subscripts k and $k+1$ indicate fine and coarse grid spaces within a multigrid method while the superscripts (e) and (n) indicate interpolation in the edge and nodal spaces. We refer the interested reader to [11] for the details. In this paper we state a few important implications of Figure 3.2. First, the commuting property guarantees that functions in the coarse grid null space interpolate to the fine grid null space. It also guarantees that the coarse grid kernel corresponds to gradients of coarse grid nodal basis functions and that the coarse grid operator contains the proper null space dimension. Finally, the commuting property between fine and coarse grid operators mirrors a similar commuting property between interpolation from (2.14) to (2.15), and the gradient operator.

The commuting diagram can be written algebraically as

$$P_k^{(e)} T_{k+1} = T_k P_k^{(n)} \tag{3.3}$$

where T_k is a discrete gradient operator on level k whose columns span $\mathcal{N}_{curl,k}$. Each row contains at most two nonzeros (± 1) and corresponds to an edge between two nodes in the associated mesh (or nodal graph) on level k.

Reitzinger and Schöberl describe how to build an algebraic multigrid method using the above commuting diagram and piecewise constant nodal interpolation, $P_k^{(n)}$. We give only a brief description here and refer the reader to [11] for more details. First, a related PDE problem is discretized by a nodal finite element method (i.e., nodal basis functions in $\mathcal{W}^0$)

$$\int_\Omega \sigma u \cdot v + \int_\Omega \frac{\Delta t}{\mu} \nabla u \cdot \nabla v \tag{3.4}$$

to yield a matrix $K_1^{(n)}$. Note that the coefficients of (3.4) are the same as those in (2.11). This matrix is given to the algebraic multigrid procedure and an entire algebraic multigrid hierarchy is constructed for $K_1^{(n)}$. Specifically, the matrix graph associated with $K_1^{(n)}$ is coarsened by aggregating unknowns together. Piecewise constant interpolation (constant over each aggregate) is used to define $P_1^{(n)}$. A coarse

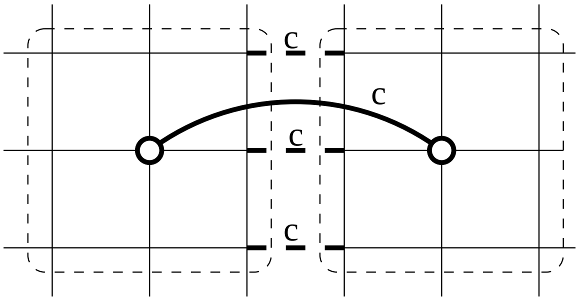

FIG. 3.3. *Interpolation on a uniform mesh within a single edge of $P_k^{(e)}$. The coarse grid edge (heavy solid) with value 'c' is interpolated only to fine edges (heavy dashed) passing between nodal aggregates (in dashed boxes).*

grid discretization matrix is defined via $(P_1^{(n)})^{\mathrm{T}} K_1^{(n)} P_1^{(n)}$ and the process is continued recursively. Effectively, this process of coarsening $K_1^{(n)}$ can be viewed as a way to coarsen the null space associated with $K_1^{(e)}$.

After the nodal mesh hierarchy is created, the coarse grid gradients (T_k) are defined as described above using the graph of $K_k^{(n)}$. Finally, the $P_k^{(e)}$'s are defined so that the commuting diagram holds. Specifically,

$$P_k^{(e)}(e_h, e_H) = \begin{cases} 1 & \text{if } (i_H, j_H) = (\mathbf{agg}(i_h), \mathbf{agg}(j_h)) \\ -1 & \text{if } (i_H, j_H) = (\mathbf{agg}(j_h), \mathbf{agg}(i_h)) \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

where $P_k^{(e)}(e_h, e_H)$ is the matrix entry giving the contribution from a coarse edge $e_H = (i_H, j_H)$ to a fine edge $e_h = (i_m, j_m)$, and $\mathbf{agg}(i) = j$ if i belongs to aggregate j and is zero otherwise. The hierarchy of edge based matrices is the one that is actually used in the multigrid iterations. Hence, the $P_k^{(n)}$'s and $K_k^{(n)}$'s can be discarded. The T_k's are still needed within the smoother to project the $K_k^{(e)}$'s into the null space.

The resulting algebraic multigrid method is fairly effective on intermediate size problems. We have in fact used this type of algebraic multigrid scheme to solve nontrivial 3D problems with over four million elements in approximately 50 iterations. Despite the promising results on intermediate problems, there is an issue of scalability. Upon examination, it is clear that $P_k^{(e)}$ is a strange interpolant. That is, while maintaining the nullspace properties of the fine grid operator, $P_k^{(e)}$ does not accurately approximate smooth error. For example, when interpolating a coarse grid edge, there are no contributions to fine grid edges that are contained completely within an aggregate. Contributions are made only to fine grid edges that connect the two aggregates defining the coarse grid edge (see Figure 3.3). In some sense, the $P_k^{(e)}$ are an approximation to a weak type of gradient of piecewise constant functions (defining $P_k^{(n)}$). Thus, it is not surprising that the resulting method is far from h-independent. Table 3.1 illustrates the growth in iterations as the mesh size increases for a model two dimensional problem using a V(1,1)[1] cycle. The smoother consists of one symmetric Gauss-Seidel iteration followed by one symmetric Gauss-Seidel iteration on the

[1] V(1,1) indicates a V cycle multigrid algorithm is used with 1 pre-smoothing sweep and 1 post-smoothing sweep on each level.

Grid Size	CG/AMG iterations
15×15	16
25×25	24
50×50	42
100×100	76
150×150	93

TABLE 3.1
Number of iterations to reduce initial residual by 10^{10} with the Reitzinger and Schöberl AMG method applied to a model problem on the unit square with $\sigma = 10^{-3}$.

nullspace projected equations followed by a final symmetric Gauss-Seidel iteration on the entire system. It is clear for very large problems that this lack of scalability will be prohibitive. Therefore the remainder of this paper discusses improvements to $P_k^{(e)}$ that improve scalability. $P_k^{(e)}$ to yield a multigrid

4. Improving the AMG Coarse Grid Correction. AMG algorithm is We now propose two new alternatives to the Reitzinger/Schöberl edge prolongator. §4.1 introduces the idea of using a smoothing iteration to improve $P^{(e)}$. §4.2 introduces a method to calculate entries of $P^{(e)}$analytically for an arbitrary $P^{(n)}$. The new prolongation operators exhibit better interpolation properties than the original grid transfers while maintaining low operator complexity. Additionally, the proposed prolongators continue to satisfy (3.3). That is, the diagram in Figure 3.2 still commutes and so coarse grid gradient functions prolongate to fine grid gradient functions.

4.1. Smoothed Prolongation. The notion of minimizing energy associated with grid transfers appears throughout the multigrid literature, perhaps most prominently in [6]. The general idea is that the coarse grid system should correct smooth error components e. These smooth components are typically characterized by small energy. That is, $e^T Ae \ll e^T e$ where A is scaled so $||A|| = 1$. This implies that the energy associated with the interpolated coarse grid correction, $P_k u_{k+1}$, should also be small. Perhaps the most well-known and explicit use of energy minimizing concepts occurs within the development of the smoothed aggregation multigrid method [16, 15]. First, an original (or tentative) interpolant is developed. Then, a new interpolant is produced by lowering the energy of the tentative prolongator. This is usually done via a simple damped Jacobi iteration

$$P_k = (I - \alpha D_k^{-1} A_k)\hat{P}_k \tag{4.1}$$

where $\hat{P}_k$ is the tentative prolongator, A_k is the system being solved, D_k is the diagonal of A_k, and α is a damping parameter. Typically, α is taken as $\frac{4}{3\rho(D_k^{-1}A_k)}$ where $\rho(\cdot)$ denotes the spectral radius. This smoothing step is critical to obtaining h-independent multigrid convergence [6, 15]. Figures 4.1 and 4.2 illustrate the effect of smoothing one dimensional piecewise constant basis functions. After applying Jacobi's method, the basis functions are now piecewise linear with overlapping support. Intuitively, smoothing the prolongator ensures that the coarse grid correction remains low energy in nature and does not pollute high energy modes on the fine grid. We refer the reader to [16] for the details. Using the smoothed aggregation idea, a new edge element AMG

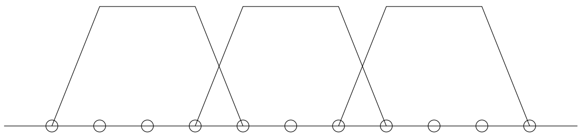

FIG. 4.1. *Piecewise constant basis functions.*

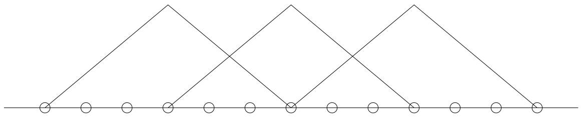

FIG. 4.2. *Smoothed basis functions.*

prolongation operator can now be defined using (4.1) with

$$\begin{aligned} \hat{P}_k &= \text{original Reitzinger/Schöberl grid transfers,} \\ A_k &= K^{(e)}_{curl,k} \end{aligned} \tag{4.2}$$

where $K^{(e)}_{curl,k}$ is the curl term in (2.11) on grid level k.

LEMMA 1. *Assume that an unsmoothed edge interpolation operator, $\hat{P}^{(e)}_k$, satisfies the commutative relation (3.3) and that (4.1) and (4.2) are used to produce a smoothed interpolation operator, $P^{(e)}_k$. Then, $P^{(e)}_k$ also satisfies (3.3).*

Proof
We have

$$\begin{aligned} P^{(e)}_k T_{k+1} &= \left(I - \alpha D_k^{-1} K^{(e)}_{curl,k}\right) \hat{P}^{(e)}_k T_{k+1} \\ &= \left(I - \alpha D_k^{-1} K^{(e)}_{curl,k}\right) T_k P^{(n)}_k \\ &= T_k P^{(n)}_k \end{aligned}$$

where we use the fact that

$$K^{(e)}_{curl,k} T_k = \Theta$$

and Θ denotes the zero matrix.

■

The implication of *Lemma* 1 is that we can maintain the important commutativity relation 3.3 while lowering the energy of the interpolant. If $K^{(e)}_k$ is used instead of $K^{(e)}_{curl,k}$ in the proof, strict commutativity no longer holds because $K^{(e)}_k$ contains the mass term. For $\sigma \ll \Delta t/\mu$, however, the additional term should have little impact. In our practical experiments, we have used $K^{(e)}_k$ instead of $K^{(e)}_{curl,k}$ without any loss in numerical convergence.

4.1.1. AMG complexity. We need to verify that prolongator smoothing does not significantly increase operator complexity (i.e. cost per multigrid iteration). One measure of complexity is

$$\text{AMG complexity} = \frac{\sum_1^{N_{levels}} nnz(K^{(e)}_k)}{nnz(K^{(e)}_k)} \tag{4.3}$$

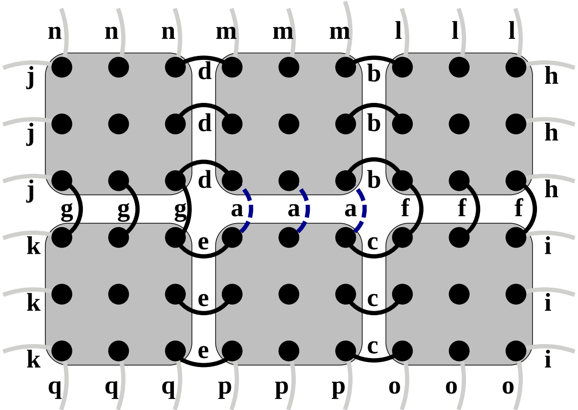

FIG. 4.3. *Uniform aggregate example of basis function distances from face 'a'. Each letter corresponds to the support within a different column of $P^{(e)}$. All distance '4' faces are marked in light gray. Distance '1' faces are marked with black solid lines.*

where $nnz(A)$ is the number of nonzeros in the matrix A. Ideally, $nnz(K_k^{(e)}) \ll nnz(K_1^{(e)})$ for $k > 1$. This occurs when coarse grid matrices have significantly fewer rows and only a modest increase in the number of nonzeros per row relative to $K_1^{(e)}$. To analyze nonzero behavior, we use the notion of matrix column distance. Define

$$dist_k(s_1, s_2) = \{p \;\; | \;\; s_1^T (K_k^{(e)})^p s_2 \neq 0, \;\; s_1^T (K_k^{(e)})^j s_2 = 0, \;\; j = 1, \dots, p-1\} \quad (4.4)$$

where $K_k^{(e)}$ is the $n \times n$ discretization matrix and s_i is a vector of length n.

Remark As a direct consequence of Galerkin coarsening, $K_{k+1}^{(e)} = (P_k^{(e)})^T K_k^{(e)} P_k^{(e)}$, we have

- $K_{k+1}^{(e)}(i,j) \neq 0 \Leftrightarrow dist_k(\phi_i, \phi_j) \leq 1$ where ϕ_r is the r^{th} column of $P_k^{(e)}$.
- $\hat{K}_{k+1}^{(e)}(i,j) \neq 0 \Leftrightarrow dist_k(\phi_i, \phi_j) \leq 3$ where $\hat{K}_{k+1}^{(e)}$ denotes the coarse matrix obtained using a *smoothed* prolongator and ϕ_r is the r^{th} column of an *unsmoothed* prolongator.[2]

Thus, comparing nonzeros in row i of $K_{k+1}^{(e)}$ produced by an unsmoothed prolongator to row i $K_{k+1}^{(e)}$ produced by a smoothed prolongator is equivalent to comparing the number of ϕ_j's with $dist_k(\phi_i, \phi_j) \leq 1$ to the number of ϕ_j's with $dist_k(\phi_i, \phi_j) \leq 3$.

Figure 4.3 illustrates a two dimensional example where each aggregate is 3×3 and perfectly aligned. Ideally, the aggregation algorithm discussed by Reitzinger and Schöberl [11] would produce such 3×3 aggregates on a uniform grid so long as the discretization of (3.4) on a uniform grid yields a 9-point stencil. Each letter corresponds to a different column in the unsmoothed prolongator and marks the only nonzero values within that column. If we look at distances from the column or face marked 'a', we have the following[3]

[2]Smoothing $P_k^{(e)}$ extends the sparsity pattern of interpolation (and also restriction) so that it corresponds to the sparsity pattern of $(K_k^{(e)} P_k^{(e)})$. This leads to the distance 3 cut-off.

[3]Graphically, (4.4) corresponds to counting the fewest number of elements (i.e. boxes in our example) between two faces.

- $dist_k(\phi_a, \phi_a) = 0$,
- $dist_k(\phi_a, \phi_\alpha) = 1, \quad \alpha = b, \ldots, g$,
- $dist_k(\phi_a, \phi_\alpha) = 4, \quad \alpha = h, \ldots, q$.

Since there are no distance '2' or distance '3' faces, the number of nonzeros in row 'a' would be identical with both the smoothed and unsmoothed prolongators. Thus, smoothing the prolongator incurs no growth in the number of nonzeros within row 'a'. Additionally, each non-boundary row of the original fine grid edge element matrix has seven nonzeros per row. This is identical to the number of faces with $dist_k(\phi_a, \phi_\alpha) \leq 3$ and hence to the number of nonzeros in the coarse grid discretization matrix using either the Reitzinger/Schöberl prolongator or the smoothed variant.

In three dimensions, the situation is identical when each aggregate corresponds to a $3 \times 3 \times 3$ brick. That is, there are no distance '2' or distance '3' faces and so smoothing does not increase AMG complexity. Of course, aggregates are not always perfect 3×3 bricks especially when solving over a complex geometry. Nonetheless, Figure 4.3 is representative. The critical feature in controlling nonzero growth is the distance between faces on opposite sides of an aggregate. This distance must be greater than '3'. Aggregation algorithms are designed so that most aggregates have a nodal diameter of '3'. This implies that faces on opposite sides of such an aggregate are a distance 4 apart. Thus, while there is some growth in complexity due to non-perfect aggregates, this growth is very small.

4.2. Local Least Squares Prolongation. method is smoothed. A second improvement is based on the observation that $P^{(e)}$ is implicitly defined by $P^{(n)}$via (3.3):

$$P_k^{(e)} T_{k+1} = T_k P_k^{(n)}.$$

Recall that (3.3) ensures that Figure 3.2 commutes, (i.e., coarse grid gradient functions are interpolated to fine grid gradient functions.) Reitzinger/Schöberl give a method for defining $P^{(e)}$ when $P^{(n)}$ corresponds to piecewise constant interpolation. In this section, we present an algorithm for computing $P^{(e)}$from an arbitrary $P^{(n)}$. This allows us to define $P^{(e)}$based on improved nodal prolongators. The work in this section was motivated in part by an informal presentation of Schöberl at a multigrid workshop[13, 12] on generating $P^{(e)}$from $P^{(n)}$using Clement interpolation [8]. [4] We refer to the improved edge prolongation operator as a *least squares* prolongator and denote it by ${}^{ls}P^{(e)}$. We stress the fact that the small local least-squares problems used to determine the prolongator coefficients have simple analytic solutions and so no numerical least squares procedure is required.

4.2.1. Interpolation to Interior Edges. When defining coefficients of $P^{(e)}$, two types of fine grid edges must be considered: *interior* and *crossing*. An *interior* edge has both endpoints within the same aggregate while a *crossing* edge has endpoints in different aggregates. We first consider interior edges. For convenience, the subscripts on $P_k^{(e)}$ and $P_k^{(n)}$ are dropped in the following discussion. Assume that we have an arbitrary nodal prolongator and want to determine the edge interpolation weights for a single interior edge. Figure 4.4 depicts a fine grid interior edge with endpoints (A, B). The prolongated nodal value at A is given by $\Sigma_{i=0}^{r} \alpha_i c_i$ where c_i

[4] Clement interpolation is defined by first projecting a function onto a constant defined on the support of each nodal shape function and then taking the resulting constants as nodal coefficients in the interpolant. The operator used in Clement interpolation is the local L^2 projection.

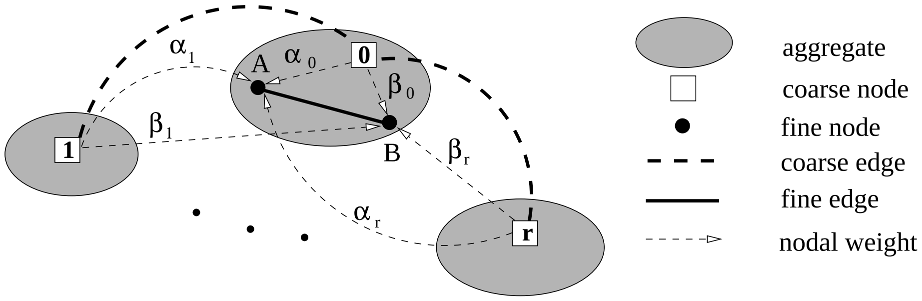

FIG. 4.4. *Arbitrary nodal interpolation for an interior edge.*

is the value at coarse node i. Similarly, the prolongated value at node B is given by $\Sigma_{i=0}^{r}\beta_i c_i$. Hence, we can rewrite the right hand side of (3.3) as

$$\begin{aligned}[T_k P^{(n)}]_{(A,B)}\ [c_0\ c_1\ \cdots\ c_r]^{\mathrm{T}} &= \Sigma_{i=0}^{r}(\beta_i-\alpha_i)c_i \\ &= [(\beta_0-\alpha_0)\cdots(\beta_r-\alpha_r)][c_0\ c_1\ \cdots\ c_r]^{\mathrm{T}} \end{aligned} \tag{4.5}$$

where $[H]_{(A,B)}$ is the matrix row of H corresponding to fine grid edge (A,B). The left hand side of (3.3) can be expressed as

$$[P^{(e)}\ ]_{(A,B)}\ T_{k+1}[c_0\ c_1\ \cdots\ c_r]^{\mathrm{T}}. \tag{4.6}$$

Equating coefficients of (4.5) and (4.6) and transposing, we have

$$T_{k+1}^{\mathrm{T}}\ [P^{(e)}]_{(A,B)}^{\mathrm{T}} = [\beta_0-\alpha_0,\ldots,\beta_r-\alpha_r]^{\mathrm{T}}. \tag{4.7}$$

We need only consider coarse edges in T_{k+1} that interact with $[P^{(e)}]_{(A,B)}$. This local submatrix of T_{k+1} is denoted by $\bar{T}_{k+1}$ and contains $(\frac{r+1}{2})r$ coarse edges (all possible edges connecting the $r+1$ aggregates).

Equation (4.7) can now be written as

$$\bar{T}_{k+1}^{T}\begin{pmatrix} p_{0,1}\\ \vdots \\ p_{0,r}\\ p_{1,2}\\ \vdots\\ p_{1,r}\\ \vdots \end{pmatrix} = \begin{pmatrix} \beta_0-\alpha_0\\ \beta_1-\alpha_1\\ \\ \vdots\\ \\ \beta_r-\alpha_r \end{pmatrix} \tag{4.8}$$

where $p_{i,j}$ is the interpolation weight from the coarse edge between the i^{th} and j^{th} aggregates to fine edge (A,B) and $\bar{T}_{k+1}^{T}$ is given by

$$\bar{T}_{k+1}^{T} = \begin{pmatrix} S_r & & & \\ & S_{r-1} & & \\ & & \cdots & \\ & & & S_1 \end{pmatrix} \tag{4.9}$$

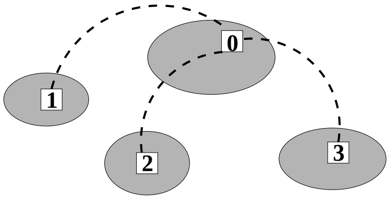

FIG. 4.5. *Coarse grid edge configuration which excludes coarse edges (1,2), (2,3), and (1,3).*

where S_i is an $i{+}1 \times i$ matrix of the form

$$S_i = \begin{pmatrix} -1 & -1 & -1 & \cdots & -1 \\ 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & \vdots & \\ & & & & 1 \end{pmatrix}. \tag{4.10}$$

In general, $\bar{T}^T_{k+1}$ has more columns than rows and so (4.8) is under-determined. Several possibilities exist to limit the number of solutions. For example, if interpolation coefficients, $\hat{p}_{i,j}$, are first computed by some standard AMG process, the commuting property will most likely not hold. However, nearby coefficients can be found that satisfy (3.3) by considering the minimization problem

$$\min_{p_{i,j}} \sum_{0\le i\le r} \sum_{i<j\le r} (p_{i,j} - \hat{p}_{i,j})^2 \tag{4.11}$$

subject to the constraint that (4.8) holds. We do not pursue this further but instead opt for an approach that simplifies the determination of interpolation coefficients. Specifically, we assume that the only nonzero weights for a fine edge interior to aggregate zero come from coarse edges that have aggregate zero as one endpoint (see Figure 4.5). That is, $p_{i,j} = 0$ for $i \ne 0$. This effectively replaces $\bar{T}^T_{k+1}$ with S_r in (4.8). S_r has one more row than column and so the system now appears over determined. It is straight forward to show that a unique solution exists and is given by

$$p_{0,j} = \beta_j - \alpha_j,\ \ j = 1, \ldots, r \tag{4.12}$$

when we assume

$$\sum_{0\le i\le r} \alpha_i = 1 \qquad \text{and} \qquad \sum_{0\le i\le r} \beta_i = 1. \tag{4.13}$$

This assumption implies that the nodal prolongator exactly interpolates constants.

4.2.2. Interpolation to Crossing Edges. We now determine interpolation coefficients for crossing edges. Figure 4.6 depicts a fine grid crossing edge with endpoints (A, B). For notational convenience, the node/aggregate indices now begin at -1. The prolongated nodal value at A is given by $\Sigma^r_{i=-1}\alpha_i c_i$, and the prolongated nodal value at B is given by $\Sigma^r_{i=-1}\beta_i c_i$ where c_i is the value at coarse node i.

Proceeding as in §4.2.1, the right hand side of (3.3) is expressed as

$$\begin{aligned} &[T_k P^{(n)}]_{(A,B)}\ [c_{-1}\ c_0\ \cdots\ c_r]^{\mathrm{T}} = \\ &\qquad [(\beta_{-1} - \alpha_{-1})(\beta_0 - \alpha_0)\cdots(\beta_r - \alpha_r)][c_{-1}\ c_0\ \cdots\ c_r]^{\mathrm{T}}. \end{aligned} \tag{4.14}$$

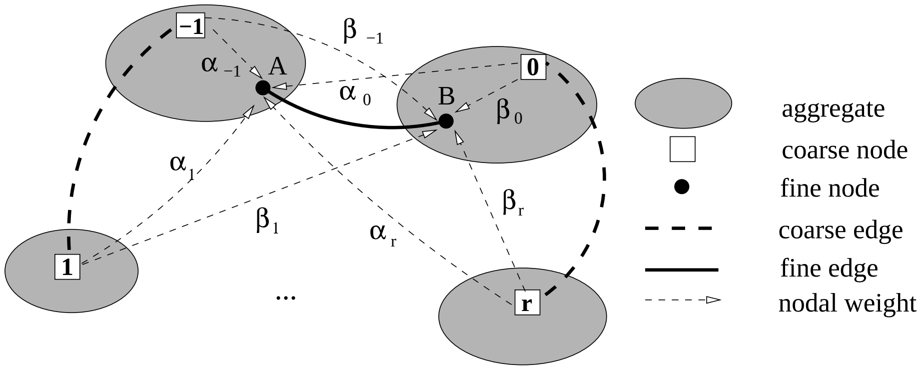

FIG. 4.6. *Arbitrary nodal interpolation for crossing edge.*

and the left hand side of (3.3) is written as

$$[P^{(e)}]_{(A,B)} \;\; T_{k+1}[c_{-1} \; c_0 \; \cdots \; c_r]^{\mathrm{T}}. \tag{4.15}$$

Equating coefficients of (4.14) and (4.15) and transposing yields

$$T^{\mathrm{T}}_{k+1} \;\; [P^{(e)}]^{\mathrm{T}}_{(A,B)} = [\beta_{-1} - \alpha_{-1}, \ldots, \beta_r - \alpha_r]^{\mathrm{T}}. \tag{4.16}$$

The local part of T_{k+1} denoted by $\bar{T}_{k+1}$ is again of the form given in (4.9). Since the system is under-determined, we exclude edges that connect aggregates in the set $\{1, 2, \ldots, r\}$ and only consider edges connecting:

1. the i^{th} and j^{th} aggregates where $i \in \{-1, 0\}$ and $j \in \{1, \ldots, r\}$
2. the 0^{th} and the -1^{th} aggregate.

Equation (4.16) is now rewritten as

$$\begin{pmatrix} \boxed{S_{r+1}} & \boxed{S_r} \end{pmatrix} \begin{pmatrix} p_{-1,0} \\ \vdots \\ p_{-1,r} \\ p_{0,1} \\ \vdots \\ p_{0,r} \end{pmatrix} = \begin{pmatrix} \beta_{-1} - \alpha_{-1} \\ \beta_0 - \alpha_0 \\ \\ \vdots \\ \\ \beta_r - \alpha_r \end{pmatrix}. \tag{4.17}$$

It is not difficult to show that (4.17) has the unique solution

$$\begin{aligned} p_{-1,j} &= -\alpha_j, \text{ for } 1 \le j \le r, \\ p_{0,j} &= \beta_j, \text{ for } 1 \le j \le r, \\ p_{-1,0} &= 1 - \beta_{-1} - \alpha_0 \end{aligned} \tag{4.18}$$

when $P^{(n)}$exactly interpolates constant functions. Using (4.12) and (4.18), it is now possible to obtain edge prolongators that commute with any $P^{(n)}$that exactly interpolates constant functions. The hope is that an improved edge prolongator results when $P^{(n)}$is better than piecewise constant.

4.2.3. Nodal Prolongators. ${}^{ls}P^{(e)}$is identical to the Reitzinger/Schöberl edge prolongator when $P^{(n)}$is piecewise constant interpolation over each aggregate. Specifically, interior edges have

$$\alpha_0 = 1, \beta_0 = 1, \text{ and } \alpha_j = \beta_j = 0 \text{ for } j \neq 0 \tag{4.19}$$

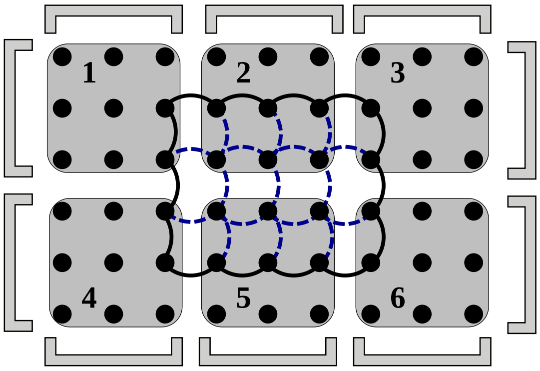

FIG. 4.7. *Uniform aggregates where dashed arcs marks support for a column of* ${}^{ls}P^{(e)}$ *and solid arcs mark distance 1 edges.*

and crossing edges have

$$\alpha_{-1} = 1,\ \beta_0 = 1,\ \alpha_j = 0 \text{ for } j \neq -1, \text{ and } \beta_j = 0 \text{ for } j \neq 0. \tag{4.20}$$

Combining (4.19) and (4.20) with (4.12) and (4.18) gives interpolation weights that are '0' for interior edges and '-1' for crossing edges.

The real interest in ${}^{ls}P^{(e)}$is to use higher order nodal interpolation. In principle, any standard algebraic multigrid procedure for generating prolongators can be considered for $P^{(n)}$. In this paper, we use smoothed aggregation to generate nodal prolongators. Specifically, a new nodal prolongator is defined using (4.1) with

$$\begin{aligned} \hat{P}_k &= \text{piecewise constant interpolation,} \\ A_k &= K_k^{(n)}. \end{aligned} \tag{4.21}$$

Notice that $\hat{P}^k$ are the original Reitzinger/Schöberl piecewise constant nodal transfers. The new nodal interpolant is used in all least squares examples presented in the remainder of this paper.

4.2.4. Operator Complexity. As with edge prolongator smoothing, the cost per iteration does not grow significantly when ${}^{ls}P^{(e)}$is derived from the smoothed nodal interpolation operator. The situation is more complex, however, than in the case of the smoothed edge prolongator. Therefore we do not digress too much on this subject as computed operator complexities of the form (4.3) and run times are given in the numerical results. Figure 4.7 illustrates a two dimensional example where each aggregate is 3×3 and perfectly aligned. The dashed arcs correspond to the only nonzero values within a column of ${}^{ls}P^{(e)}$. The solid arcs are distance 1 edges from the face defining the dashed arcs. Only columns of ${}^{ls}P^{(e)}$containing an arc depicted in the figure will correspond to a distance 1 face. The key point is that columns corresponding to faces highlighted by gray square brackets do not contain any of these edges and so these faces (faces across aggregates from the central face) do not add additional nonzeros. We omit a detailed analysis of this subject and generalization to higher dimensions as numerical results are given to measure the AMG complexity.

4.2.5. Smoothing ${}^{ls}P^{(e)}$. One additional algorithm is presented. This corresponds to combining the smoothing and the least squares ideas of §§4.1-4.2. In particular, the edge interpolator produced by the least squares algorithm can be smoothed.

This corresponds to using (4.1) with

$$\begin{aligned} \hat{P}_k &= {}^{ls}P^{(e)}, \\ A_k &= K^{(e)}_{curl,k}. \end{aligned} \tag{4.22}$$

At first glance this smoothing step would appear to introduce a high AMG operator complexity. In fact, the gray square bracket faces in Figure 4.7 correspond to distance 3 faces. However, beneficial cancellation occurs to prevent large scale growth. This cancellation relies on properties of the curl operator and the fact that some nodal interpolation weights are the same (when smoothed aggregation is used to define the nodal interpolant). These weights are the same in regions where σ is constant or where σ is ignored when smoothing the nodal prolongator. Unfortunately, the beneficial cancellation only occurs when $K^{(e)}_{curl,k}$ is used in (4.22). In our experiments, however, we use $K^{(e)}_k$. To recover the benefits of cancellation, a drop tolerance must be employed. Specifically, using $K^{(e)}_k$ (instead of $K^{(e)}_{curl,k}$) introduces a number of small terms in the prolongator. By dropping interpolation weights that are less than 10^{-5}, we can maintain the cancellation benefits without degrading numerical convergence. We avoid digressing into the details of this cancellation as they are somewhat tedious. Numerical results will be given with the corresponding operator complexities.

5. Numerical Experiments. To assess the interpolation improvements proposed in this paper, we compare the following AMG methods:

R/S AMG	The original Reitzinger and Schöberl (R/S) AMG.
R/S-LS AMG	$P^{(e)}_k$ derived via the Section 4.2 least-squares algorithm using a smoothed R/S nodal interpolant: $P^{(n)}_k = (I-\gamma K^{(n)}_k)\hat{P}^{(n)}_k$ where $\hat{P}^{(n)}_k$ is piecewise constant interpolation and $\gamma = \frac{3}{2}\rho(K^{(n)}_k)$.
R/S-S AMG	Smoothed R/S edge interpolant: $P^{(e)}_k = (I-\gamma K^{(e)}_k)\hat{P}^{(e)}_k$ where $\hat{P}^{(e)}_k$ is the R/S edge interpolant and $\gamma = \frac{4}{3}\rho(K^{(e)}_k)$ and entries in $P^{(e)}_k$ are dropped as above.
R/S-SLS AMG	Smoothed R/S-LS edge interpolant: $P^{(e)}_k = (I-\gamma K^{(e)}_k)\hat{P}^{(e)}_k$ where $\hat{P}^{(e)}_k$ is the R/S-LS edge interpolant and $\gamma = \frac{4}{3}\rho(K^{(e)}_k)$ and entries in $P^{(e)}_k$ are dropped as above.

In each case a multigrid V cycle is used as a preconditioner for a conjugate gradient iteration. The pre-smoother consists of three steps:

1. A first order Chebyshev semi-iterative scheme accelerating a Jacobi iteration is applied to $K^{(e)}_k u = f$.
2. A new residual, r, is computed. A first order Chebyshev semi-iterative scheme accelerating a Jacobi iteration is applied to $(T_k^T K^{(e)}_k T_k)\hat{u} = T_k^T r$.
3. The *step 2* correction is added to the *step 1* estimate, i.e. $u \leftarrow u + T_k\hat{u}$.

The details of this Chebyshev semi-iterative method can be found in [1]. It is oriented toward damping high frequency errors and uses only an estimate of the largest eigenvalue of the matrix. The post-smoother performs *step 1* and *step 2* in reverse order and guarantees the symmetry of the preconditioner. For the uniform grid examples,

σ	Grid Size	R/S	R/S-LS	R/S-S	R/S-SLS
10^1	30×30	29/.05	27/.05	19/.04	14/.03
	90×90	65/1.4	66/1.7	26/.6	18/.4
	270×270	136/27.1	135/31.7	39/8.2	23/5.3
	810×810	288/524.6	289/593.1	56/105.1	29/63.1
10^{-1}	30×30	34/.06	43/.09	29/.06	23/.05
	90×90	75/1.6	76/1.9	44/.9	31/.7
	270×270	170/34.9	170/41.2	69/14.8	41/9.3
	810×810	256/460.9	256/528.5	79/145.9	42/90.4
10^{-3}	30×30	40/.08	38/.09	27/.05	22/.05
	90×90	50/1.1	50/1.3	40/.8	18/.4
	270×270	114/22.9	114/27.0	31/6.6	23/5.3
	810×810	249/446.1	249/504.2	42/77.6	30/65.5

TABLE 5.1

Iteration counts/run times (in seconds) for the CG/AMG V(1,1) schemes and various values of σ on a model problem.

method	AMG complexity
R/S	1.1
R/S-LS	1.3
R/S-S	1.1
R/S-SLS	1.1

TABLE 5.2

AMG Complexities for 810×810 grid. Note: R/S-SLS complexities are lower than R/S-LS complexities due to the dropping of small prolongator entries in the R/S-SLS variant.

a right hand side is generated by taking a random vector and multiplying by the discretization matrix. A zero initial guess is used for the uniform grid examples. Finally, the iterations terminate when the 2-norm of the residual is reduced below 10^{-6}.

We first consider the two dimensional version of (2.11) on the unit cube with Dirichlet boundary conditions, a constant value of σ, and $\Delta t/\mu = 1.0$. Table 5.1 gives iteration counts and run time while Table 5.2 gives AMG complexity results. There are several things to notice about the data. The most important observation is the dramatic improvement in both iterations and run time associated with using the R/S-S and R/S-SLS algorithms. On the 810×810 problem the run time was reduced from about 500 seconds to about 65 seconds (almost a factor of 8). Further, algorithm convergence does not degrade as σ becomes smaller. In fact, convergence of the smoothed variants is poorest for intermediate values of σ. It is also interesting to note that the R/S-LS algorithm offers no improvement over the R/S algorithm. That is, using the improved nodal interpolation to derive the edge interpolant does not necessarily imply better convergence. It should be recalled that the $P_k^{(e)}$'s are effectively a kind of weak gradient of the $P_k^{(n)}$'s. Thus, for example, when $P_k^{(n)}$

σ	Grid Size	R/S	R/S-LS	R/S-S	R/S-SLS
10^1	$15 \times 15 \times 15$	17/.8	15/1.0	12/.6	10/1.3
	$45 \times 45 \times 45$	42/96.6	35/100.8	21/49.1	17/65.0
	$135 \times 135 \times 135$	10/7040.3	90/7310.5	37/2473.1	29/2671.5
10^{-1}	$15 \times 15 \times 15$	19/.9	17/1.1	12/.6	11/1.2
	$45 \times 45 \times 45$	47/108.3	42/121.0	22/51.0	20/72.0
	$135 \times 135 \times 135$	126/8054.3	105/8512.7	41/2731.5	36/3312.1
10^{-3}	$15 \times 15 \times 15$	19/.9	17/1.1	12/.6	11/1.3
	$45 \times 45 \times 45$	48/108.9	42/122.6	21/50.0	20/73.8
	$135 \times 135 \times 135$	126/8063.1	108/8754.6	37/2477.8	37/3408.8

TABLE 5.3

Iteration counts/run times (in seconds) for the CG/AMG V(1,1) schemes and various values of σ on a model problem.

method	AMG complexity
R/S	1.04
R/S-LS	1.29
R/S-S	1.04
R/S-SLS	1.46

TABLE 5.4

AMG Complexities for $135 \times 135 \times 135$ *grid.*

corresponds to linear interpolation, the $P_k^{(e)}$'s are still something less than linear interpolation. As discussed in §4.2, there are other possible simplifications of the local coarse grid edge matrix $\bar{T}_{k+1}^T$ that may lead to better convergence properties. Finally, it is important to notice that the best algorithm RS-SLS is still not h-independent. However, the growth in iterations is quite modest. Similar (though somewhat less dramatic) results are observed for the three dimensional model problem depicted in Table 5.3 and Table 5.4. Overall the smoothed variants performs best. The RS-SLS generally requires the fewest iterations though it does have a higher AMG complexity. The run time improvements associated with the new interpolants are somewhat less pronounced than in the two dimensional case due to the smaller mesh sizes. However, there is still about a factor of three improvement in run time when using the new methods. As mesh sizes grow, the advantages of the two smoothing variants should become even more significant.

We now describe a problem that is representative of three dimensional problems associated with the Z-pinch apparatus described in [18]. This reduced scale problem is a conductive cylinder 4 mm high and 6 mm in diameter in which a cylindrical slot has been cut out from one end as shown in Figure 5.1. The cylinder is given a conductivity of 6.33×10^7 while the slot is represented by a "void" region with conductivity 1.0. Also $\Delta t = 5 \times 10^{-6}$ and $\mu = 4\pi \times 10^{-7}$. Homogeneous tangential electric field Dirichlet boundary conditions are applied on the center and outside surface of the top of the cylinder. Tangential magnetic field boundary conditions are applied weakly to the remaining surfaces. Homogeneous magnetic field conditions are applied on

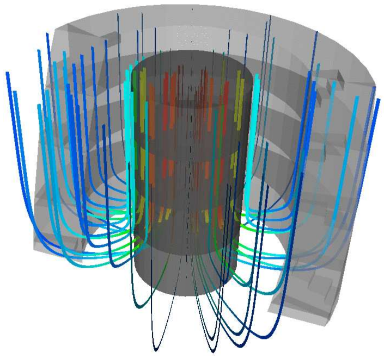

FIG. 5.1. *Cutaway of axial slot configuration showing material discontinuities and current density streamlines colored by current density magnitude.*

Number of edges	R/S	R/S-LS	R/S-S	R/S-SLS
130,008	44/98.1/1.15	39/94.8/1.42	36/75.9/1.06	34/83.6/1.45
451,000	81/23.7/1.13	–	66/19.0/1.08	–

TABLE 5.5
Slot problem iteration counts, run times, and multilevel complexities for various CG/AMG V(1,1) schemes. The stopping criterion was reduction of the relative residual $||r||_2/||b||_2$ *by* 10^8.

the outer and bottom surfaces and an inhomogeneous azimuthal tangential magnetic field condition is given on the middle ring surface at the top of the cylinder. The field fills the slot immediately for the time step chosen. Table 5.5 gives the results for this solve for each problem size. A zero initial guess is given and the iterative solution is terminated when $||r||_2/||b||_2 \leq 10^{-8}$. The first problem is meshed with $44,544$ hexahedral elements resulting in $46,761$ nodes and $130,008$ edges and the second contains 150528 elements, 154505 nodes, and approximately $451,000$ edges. Table 5.5 illustrates the results for this solve for each problem size. The R/S-LS and R/S-SLS methods have not been adapted to parallel, so results for these two methods are shown only for the smallest grid size. The R/S-S and R/S-SLS methods again show roughly the same performance for the smaller slot problem. For the R/S-S method, there is modest growth in the number of iterations. The R/S-S variant still outperforms the original R/S method, although the gains are more modest than in the model problem tests. The R/S-S method requires approximately 20% fewer iterations than the original method for both problem sizes.

6. Conclusions. We have proposed two improvements to the Reitzinger and Schöberl AMG grid transfers for Maxwell's equations. One improvement is based on smoothing the grid transfer proposed in the original method using ideas from smoothed aggregation. A second idea involves using a higher order nodal interpolation operator and then deriving an edge interpolation operator while maintaining the commuting diagram property advocated by Reitzinger and Schöberl. Numerical results have been presented showing the improved iteration counts and run times associated with the new grid transfers. While not completely h-independent, the new AMG/CG method demonstrates improved convergence behavior and has only a slight growth in the number of iterations as the mesh size increases.

REFERENCES

[1] M. F. Adams, M. Brezina, J. J. Hu, and R. S. Tuminaro, *Parallel multigrid smoothing: polynomial versus Gauss-Seidel*. Submitted for publication.

[2] D. N. Arnold, R. S. Falk, and R. Winther, *Multigrid in H(div) and H(curl)*, Numer. Math., 85 (2000), pp. 197–217.

[3] P. B. Bochev, J. J. Hu, A. C. Robinson, and R. S. Tuminaro, *Parallel algebraic multigrid for Maxwell's equations*, Electronic Transactions on Numerical Analysis, (2002).

[4] A. Bossavit, *A rationale for "edge-elements" in 3-D fields computations*, IEEE Transactions on Magnetics, 24 (1988), pp. 74–79.

[5] ———, *Computational electromagnetism*, Academic Press, 1998.

[6] J. Bramble, J. Pasciak, J. Wang, and J. Xu, *Convergence estimates for multigrid algorithms without regularity assumptions*, Math. Comp., 57 (1991), pp. 23–45.

[7] W. L. Briggs, V. E. Henson, and S. McCormick, *A multigrid tutorial, Second Edition*, SIAM, Philadelphia, 2000.

[8] P. Clement, *Approximation by finite element functions using local regularization*, RAIRO, Anal. Numer., 9, R-2 (1975), pp. 77–84.

[9] W. Hackbusch, *Multigrid Methods and Applications*, vol. 4 of Computational Mathematics, Springer–Verlag, Berlin, 1985.

[10] R. Hiptmair, *Multigrid method for Maxwell's equations*, SIAM J. Numer. Anal., 36 (1998), pp. 204–225.

[11] S. Reitzinger and J. Schöberl, *An algebraic multigrid method for finite element discretizations with edge elements*, Numerical Linear Algebra with Applications, 9 (2002), pp. 223–238.

[12] J. Schöberl, *Commuting quasi-interpolation operators for mixed finite elements*. http://www.sfb013.uni-linz.ac.at/~joachim/papers/publications.html, 2001.

[13] ———, *Informal presentation*, 2001. Multigrid workshop held in Livermore, CA.

[14] U. Trottenberg, C. Oosterlee, and A. Schüller, *Multigrid*, Academic Press, London, 2001.

[15] P. Vaněk, M. Brezina, and J. Mandel, *Convergence of algebraic multigrid based on smoothed aggregation*, Tech. Report report 126, UCD/CCM, Denver, CO, 1998.

[16] P. Vaněk, J. Mandel, and M. Brezina, *Algebraic multigrid based on smoothed aggregation for second and fourth order problems*, Computing, 56 (1996), pp. 179–196.

[17] J. S. V. Welij, *Calculation of eddy currents in terms of H on hexahedra*, IEEE Transactions on Magnetics, MAG-21 (1985), pp. 2239–2241.

[18] G. Yonas, *Fusion and the Z pinch*, Scientific American, 279 (1998), pp. 22–27.