

AN AGGREGATION MULTILEVEL METHOD USING SMOOTH ERROR VECTORS*

EDMOND CHOW†

Abstract. This paper presents an aggregation multilevel method for problems where the near-nullspace of the operator is not known, in particular, the method does not assume that the slow-to-converge error is locally constant. The method uses samples of slow-to-converge error to construct its interpolation operator. The basis vectors for an aggregate are computed via a singular value decomposition of the sample vectors over that aggregate. Compared to other methods, this method does not require that the stiffness matrices for each aggregate (which sum to the global matrix) are available.

Key words. preconditioning, multilevel and multigrid methods, smoothed aggregation

AMS subject classifications. 65F10, 65F35, 65F50

1. Introduction. Algebraic multilevel methods use complementary smoothing and coarse grid correction for solving linear systems from discretized partial differential equations. In these methods, *algebraically smooth error* is the error that remains after the smoother has been applied and that must be reduced at the next level. The *interpolation* or *prolongation* operator P must be able to represent this smooth error on the coarser level, i.e., P must be constructed such that the algebraically smooth error is in the range of P . To accomplish this, multilevel methods traditionally must make assumptions about the nature of the smooth error. In algebraic multigrid (AMG), smooth error is assumed to be locally constant or slowly varying along strong couplings [14].

In aggregation multilevel methods for elliptic PDEs, the near-nullspace of the discrete operator is assumed to locally form an approximate basis for the algebraically smooth error. For second and fourth order PDEs, these near-nullspace vectors are constants and linear functions [16]; for linear elasticity, these vectors are rigid body modes [8]. If it is known that the solution is physically smooth, then the geometric coordinate vectors of the grid points, x , y , and z , and monomial functions of these, may locally represent the algebraically smooth error. Similarly, for p -version finite elements, appropriate basis vectors are also known [13].

Algebraic multilevel methods often fail because the above assumptions do not hold. For example, a simple scaling of the matrix will change the near-nullspace. In other cases, the near-nullspace and the nature of the algebraically smooth error are simply not known. Further, aggregation algebraic multilevel methods are often not robust because the near-nullspace vectors are not sensitive to the PDE coefficients, which may be important, for example, for anisotropic problems. (This problem is alleviated somewhat by using an aggregation technique that is sensitive to the strength of connection between the variables.) Finally, for aggregation methods, it may be desirable for some problems to use basis vectors in addition to the near-nullspace vectors. A procedure is then needed to compute and incorporate these additional vectors.

*Lawrence Livermore National Laboratory Technical Report UCRL-JRNL-204139. This work was performed under the auspices of the U.S. Department of Energy by University of California Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48.

†Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, L-560, Box 808, Livermore, California, 94551 (echow@llnl.gov).

Recently, methods have been developed that are designed to be more robust and more general by not making assumptions about the algebraically smooth error for a given problem. Instead, these methods attempt to reduce some measure of the interpolation error. Some of these methods can also add additional basis vectors if necessary, leading to better convergence [6, 9]. We briefly review some of these methods here.

In the energy minimization technique [17], P is constructed by directly optimizing a constant in the subspace correction framework [18] and thus improving the convergence rate. The stability and approximation inequalities are satisfied, guaranteeing mesh-independent convergence. For finite element discretizations, the AMGe method [4] constructs P by minimizing a measure of the interpolation error [2, 1],

$$(1.1) \quad \min_{P,v} \frac{\|u - Pv\|_2}{\|u\|_A}, \quad \forall u \neq 0$$

where it is assumed that A is scaled to have a unit diagonal. A better minimum leads to a better convergence rate for a two-level method. However, to be practical, the minimization (1.1) must be localized over an aggregate of elements or grid points. The solution turns out to involve the eigenvectors of the aggregate matrices [11, 6, 9].

For aggregation multilevel methods, a local basis for the algebraically smooth error for each aggregate is formed by the low-energy eigenvectors of the aggregate submatrix. These submatrices are defined differently depending on whether grid points or elements are aggregated. For non-overlapping aggregates of the grid points, the aggregate submatrix is a stiffness matrix restricted to the grid points of an aggregate [11]. P constructed this way, however, will generally not contain the near-nullspace of the operator. For an element aggregation, the aggregate submatrix is the assembled superelement corresponding to an aggregate of elements before application of the boundary conditions [6, 9]. This approach, however, requires access to the element stiffness matrices.

In this paper, we present a new approach for the case where no assumptions about the algebraically smooth error can be made. We construct P based on grid point aggregation, but instead of relying on aggregate submatrices and their low-energy eigenvectors, our approach is to directly construct interpolation operators that will interpolate samples of algebraically smooth error or other low-energy vectors.

For a coefficient matrix A , samples of algebraically smooth error can be generated by applying the smoother to $Ae = 0$, starting with a random initial guess. Alternatively, low-energy vectors may be used, constructed cheaply by a few steps of a Lanczos or Arnoldi procedure. Building interpolation that fits algebraically smooth error in the AMG context is also being investigated [3].

Algebraically smooth error is also used explicitly in recently developed *adaptive* multilevel methods. These methods update themselves as the algebraically smooth error becomes revealed in the solution process. In these strategies, a few steps of the “current” method is applied to the homogeneous problem $Ax = 0$ to reveal the components of the error that are slow to converge. Versions based on both smoothed aggregation [5] and AMG [12] have been developed. We also note that samples of algebraically smooth error may be used to define a coarsening procedure [10].

In Section 2, the new interpolation operator is described. The interpolation is matrix-dependent; anisotropies and physical jumps in the smoothed error are reflected in the interpolation operator. In Section 3, results of numerical tests are shown and discussed.

2. Interpolation using smooth error vectors.

2.1. New interpolation operator. In this section, we describe an interpolation operator based on aggregating grid points that is constructed from samples of low-energy vectors, rather than eigenvectors of aggregate submatrices. Here, we assume that the aggregates are given. We also suppose that the coefficient matrix A at each grid level has been reordered such that the rows and columns of A for an aggregate are ordered together and consecutively by aggregates. This will simplify the notation in this paper.

As in smoothed aggregation [15, 16], we will smooth the tentative interpolation operator in order to reduce the energy of the basis functions. Thus, we begin by seeking a rectangular tentative interpolation operator of the form

$$\tilde{P} = \begin{bmatrix} \tilde{P}_1 & & & & \\ & \ddots & & & \\ & & \tilde{P}_i & & \\ & & & \ddots & \\ & & & & \tilde{P}_J \end{bmatrix}$$

for J aggregates, where \tilde{P}_i forms the tentative basis vectors for aggregate i . The \tilde{P}_i matrices are $n_i \times k_i$ with $n_i \geq k_i$, and \tilde{P} is block diagonal because the grid point aggregates are non-overlapping.

Let $S = [s_1, \dots, s_m]$ be a block of m algebraically smooth error vectors. These vectors may be generated by applying the smoother (to be used in the multigrid solution process) to the homogeneous equations

$$Ae = 0$$

with a random initial guess for e . When the near-nullspace of A is known, then S should be composed of these near-nullspace vectors and the method will be essentially the same as the smoothed aggregation method [16]. The samples S may be partitioned as

$$S = \begin{bmatrix} S_1 \\ \vdots \\ S_i \\ \vdots \\ S_J \end{bmatrix}$$

corresponding to the partitioning of \tilde{P} .

Given S_i , the portion of the sample vectors corresponding to aggregate i , we seek a low-rank approximate basis \tilde{P}_i for S_i . Formally, we seek

$$\min_{\tilde{P}_i, W} \|S_i - \tilde{P}_i W\|_2$$

where \tilde{P}_i has rank k_i , where $k_i \leq m$. The minimum is achieved when

$$\tilde{P}_i W = U_{k_i} \Sigma_{k_i} V_{k_i}^T$$

where $U_{k_i} \Sigma_{k_i} V_{k_i}^T$ is the rank k_i truncated singular value decomposition of S_i . By matching variables, $\tilde{P}_i = U_{k_i}$, the first k_i left singular vectors. We note that the computations are small, dense SVD computations.

This technique exploits the fact that a local portion (over an aggregate) of an algebraically smooth sample vector may have larger or smaller local energy than the same portion of other algebraically smooth sample vectors. The truncated SVD reduces the effect of the higher-energy portions, while capturing the desirable low-energy components which are more typical. Using more sample vectors than k_i , the minimum required, generally improves the bases constructed for the aggregates.

Recalling (1.1), the approximate basis should be better for samples with lower energy, i.e., we wish to have a smaller residual corresponding to samples with smaller energy. This can be accomplished, although imprecisely, by scaling each sample vector s_j by $(s_j^T A s_j)^{-1}$ before computing the singular value decomposition. (Empirically this scaling was more effective than $(s_j^T A s_j)^{-1/2}$.) This is particularly important if the vectors have very different energy norms. Ideally, we would like to scale the local portion of the sample vectors by their *local* energy norms. However, this would require the construction of aggregate submatrices.

Finally, to reduce the energy of the basis functions, \tilde{P} is smoothed one step by a Jacobi smoother to construct the final P . The Jacobi smoother is

$$(2.1) \quad I - \frac{4}{3\rho} D^{-1} A$$

where D is the diagonal of A , and ρ is the spectral radius of $D^{-1}A$ estimated by a few steps of a Lanczos or Arnoldi method. This smoothing preserves constant functions.

2.2. Number of sample vectors. The number of sample vectors is a parameter of this method. A larger number of sample vectors will improve the convergence rate, but this must be balanced with the cost of generating these sample vectors and the cost of larger SVD computations.

Figure 2.1 plots the local energy of each of 20 sample vectors for a 9-node aggregate of a 1-D isotropic diffusion operator. The sample vectors were generated by 3 steps of symmetric Gauss-Seidel (SGS) applied to the homogeneous error equation with random initial guesses. The figure also plots the local energies of the first, second, and third singular vectors as the number of sample vectors increases. It is evident that the first singular vector has the least energy compared to the second and third, and that the energy of the first vector generally decreases when more sample vectors are used. Interestingly, the energy of this singular vector appears to decrease when a low-energy sample vector is generated. High-energy sample vectors do not dramatically affect the energy of the first few singular vectors. However, after a moderate number of sample vectors, the energy of the first singular vector then decreases very slowly. Thus, using a very large number of sample vectors is not effective, and should be balanced with the number of smoothing steps applied to each sample vector. This will also be reflected in the numerical tests of the convergence rate. We note that in these examples, the local energy of each sample is plotted. In practice, these local energies cannot be computed if the local aggregate matrices are not known.

2.3. Example. Figure 2.2 shows a sample of algebraically smooth error and three basis vectors produced for an anisotropic diffusion problem with Dirichlet boundary conditions on a 32×32 grid. The direction of anisotropy is 45° (bottom left to top right). The aggregates, shown by the lighter boundary lines, are 4×4 grid points.

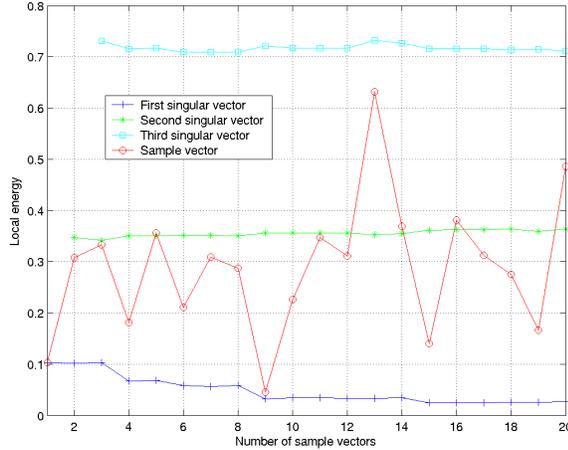


FIG. 2.1. Local energy of samples and singular vectors vs. number of sample vectors. The energy of the first singular vector generally decreases when more samples are used, but decreases very slowly after a moderate number of samples.

The value at each grid point is indicated by the gray-level of the grid point. For the illustration to be clear, 20 sample vectors were used, with 20 SGS smoothing steps for each vector.

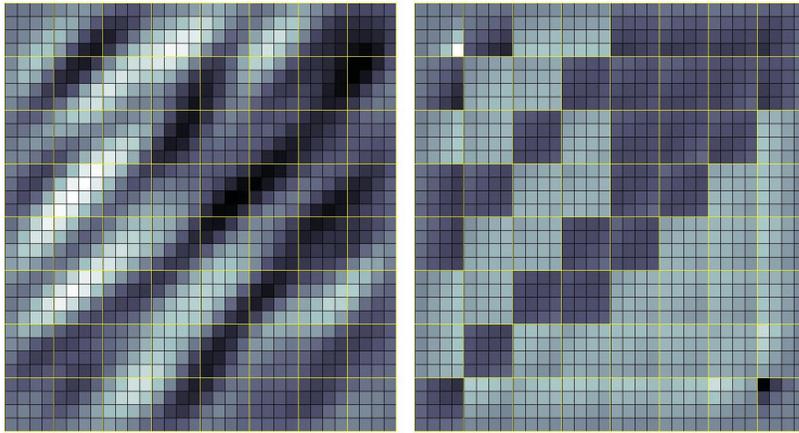
As expected, in Figure 2.2(b), the first basis vector contains nearly locally constant values over each aggregate. Some aggregates have positive values and others negative, which explains the two different gray-levels predominant for this basis vector. It is also noticeable in the figure that at the Dirichlet boundaries there is a decay of the basis vectors toward zero. This problem-dependent behavior is a feature of this method.

Figure 2.2(c) shows that the second basis vector over each aggregate varies slowly in the direction of anisotropy, like the algebraically smooth error, and varies sharply in the cross direction, again like the smooth error. Figure 2.2(d) shows that the third basis vector is oscillatory in the cross direction, which is very helpful in representing the smooth error in this example. These oscillatory basis functions cannot be constructed from simple smooth functions such as polynomials of the coordinate vectors. Thus, these are good basis functions for representing the algebraically smooth error in Figure 2.2(a).

2.4. Number of basis vectors per aggregate. In the smoothed aggregation method [16], the number of basis vectors is chosen beforehand, with each vector being a near-nullspace vector that is known for the problem. In spectral AMGe [9] and the method in [6], the number of basis vectors may be different for each aggregate, and may be chosen such that (1.1) is bounded. In the method presented here, the number of basis vectors may also be different for each aggregate, but no bound on (1.1) is possible.

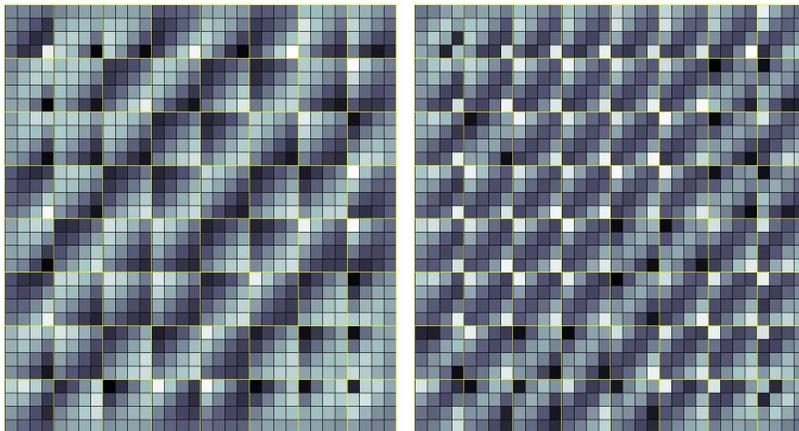
For aggregate i , the number of basis vectors may be chosen based on the singular values from the singular value decomposition of S_i . More basis vectors will improve interpolation for the given aggregate, but increases the cost of the method.

For certain problems, a different number of basis vectors may be suitable for each



(a) Sample of algebraically smooth error

(b) First basis vector



(c) Second basis vector

(d) Third basis vector

FIG. 2.2. An anisotropic diffusion problem with Dirichlet boundary conditions. The first basis vectors (b) are nearly constant over each aggregate. The second (c) and third (d) basis vectors vary slowly in the direction of anisotropy, and sharply in the cross direction, like the sample of algebraically smooth error (a).

aggregate. One strategy is to use the singular vectors that correspond to singular values that are larger than δ times the largest singular value, with $0 \leq \delta < 1$. A sharp decay of the singular values indicates that only a few singular vectors suffice to accurately represent the sample vectors.

To illustrate this, consider two anisotropic diffusion problems on a 32×32 grid using 4×4 aggregates. The two problems have different angles of anisotropy, 0° and 45° . Figures 2.3(a) and 2.3(b) plot the first seven singular values for every aggregate for these two problems. Each curve represents an aggregate, with the singular values

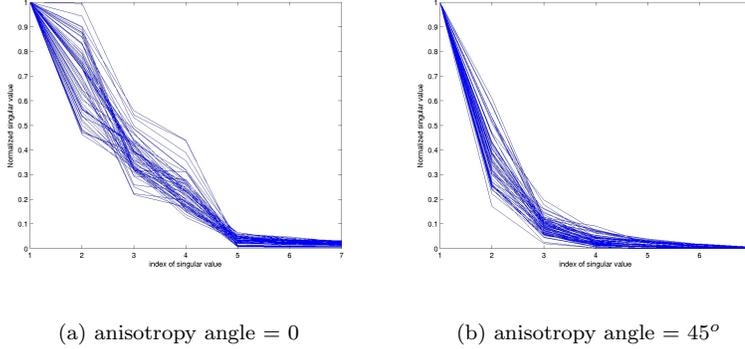


FIG. 2.3. First seven singular values for each aggregate (each curve represents an aggregate), scaled such that the first singular value equals unity. The plot suggests using four and three basis vectors for (a) and (b), respectively.

scaled such that the largest one for each aggregate is unity. For an angle of anisotropy of 0, it is natural to use four basis vectors since there are four lines of grid points in each aggregate, and this is reflected in the singular values. For an angle of anisotropy of 45°, a smaller number of basis vectors for many aggregates appears adequate.

2.5. Extension to multiple levels. Once P has been defined for an operator A , the operator at the next coarser level may be defined by $A_c = P^T A P$. The method can be extended to multiple levels by applying the method recursively to solve the coarse grid correction equations involving A_c .

To apply the method to A_c , a set of algebraically smooth error vectors for A_c is needed. These vectors may be generated from scratch, but the following procedure is more effective.

Recall that S denotes m low-energy vectors for A . We seek T , which denotes m low-energy vectors for A_c . If T is constructed such that

$$S = PT$$

then $A_c T = P^T A S \approx 0$ and the vectors in T are low-energy. Assuming that $P \approx \tilde{P}$, then aggregate-wise we have

$$(2.2) \quad S_i \approx \tilde{P}_i T_i$$

where T_i is the portion of T corresponding to aggregate i . From the SVD of S_i already computed,

$$(2.3) \quad S_i = U_i \Sigma_i V_i^T$$

$$(2.4) \quad \approx \tilde{P}_i \Sigma_{k_i} V_{k_i}^T$$

and matching variables in (2.2) and (2.4), we have

$$T_i = \Sigma_{k_i} V_{k_i}^T.$$

Once T is formed, it generally is not smooth enough due to the approximations made and the fact that this is a representation of a smooth vector on a coarser grid. Thus

T must also be smoothed a few steps before it can be used. For small problems, a small number of steps are needed, and more steps do not significantly improve the convergence rate. For large problems, more steps can be advantageous. In our tests, we used 2 steps.

For efficiency, it is possible to perform these smoothing steps along with the spectral radius estimation required by the prolongation smoother (2.1) by performing matrix-vector products with a block of vectors simultaneously.

3. Numerical Tests. In this section, we test smooth-vector interpolation in a preconditioned multigrid method. For the multigrid method, restriction is defined to be the transpose of interpolation, the coarse grid operator is the Galerkin coarse grid operator, and V(1,1) cycles are used, with symmetric Gauss-Seidel smoothing. Unstructured aggregation of the grid points used the algorithm of the smoothed aggregation method [16], with a strength threshold of 0.08. For coarser levels and non-scalar problems, the degrees of freedom corresponding to a grid point are always aggregated together.

The test problems are discretizations of 2-D unstructured isotropic and anisotropic diffusion equations,

$$\begin{aligned} au_{xx} + bu_{yy} &= f & \text{in } \Omega = (0,1)^2 \\ u &= 0 & \text{on } \partial\Omega \end{aligned}$$

where $a = 1$ and $b = 1000$ for the anisotropic problems. Random right-hand sides were used. The discretization was linear triangular finite elements. Most of the tests were performed with the largest test problems, UNI7 and ANI7 (isotropic and anisotropic, respectively), which have 205,761 equations and 1,436,481 nonzeros. Smaller test problems were also used. In addition, one test with a plane strain problem was performed and will be described later.

Conjugate gradient acceleration was used with a zero initial guess. The iterations were stopped when the preconditioned residual norm was decreased by 8 orders of magnitude. The experiments were run on a Linux 1.5 GHz Intel Xeon computer with 256 kbytes of cache memory and 512 Mbytes of main memory. Due to the use of randomness in the interpolation, the iteration counts vary by a few steps from test to test.

3.1. Test results for isotropic and anisotropic problems. Tables 3.1 and 3.2 show test results with UNI7 and ANI7, respectively. The algebraically smooth error vectors that were used were generated by SGS relaxation, and the tables show results for various numbers of smooth sample vectors and various numbers of smoothing steps per vector. The top portion of each table shows results when a budget of 36 smoothing steps is used to generate the smooth vectors, either using fewer vectors and more smoothing steps per vector, or vice-versa. Three basis vectors were used for each aggregate. Five levels were used in the method, and the grid and operator complexities [7] were approximately 1.4 and 3.6, respectively. Operator complexity is an indication of the work per V-cycle of the multigrid process, relative to a matrix-vector multiply with the fine-grid matrix.

Tables 3.1 and 3.2 show the main point of this paper: given a fixed budget of smoothing steps, it can be worthwhile to use more vectors that are less smooth, than to apply all the smoothing steps to a small number of vectors. When CG acceleration is not used, the effect is much more pronounced. In any code, the improvement depends on the portion of the preconditioner setup cost that is due to generating the

Smooth vectors	Smoothing steps	Iterations	Time (s)		
			Setup	Solve	Total
3	12	41	10.29	10.56	20.85
4	9	31	10.60	7.98	18.58
6	6	26	11.43	6.76	18.19
9	4	28	12.65	7.26	19.91
12	3	31	13.53	8.00	21.53
12	6	23	14.59	6.02	20.61
12	12	16	16.03	4.25	20.28
12	24	14	19.53	3.74	23.27
12	36	12	23.04	3.25	26.29

TABLE 3.1
Test results for UNI7.

Smooth vectors	Smoothing steps	Iterations	Time (s)		
			Setup	Solve	Total
3	12	221	13.77	58.91	72.68
4	9	187	14.15	49.92	64.07
6	6	164	14.88	43.72	58.60
9	4	158	16.25	42.25	58.50
12	3	156	17.61	41.89	59.50
12	6	126	17.81	33.71	51.52
12	12	109	19.55	29.07	48.62
12	24	97	22.82	25.96	48.78
12	36	86	26.31	23.01	49.32
12	48	80	29.80	21.41	51.21
12	60	76	33.23	20.35	53.58
12	72	68	37.02	18.27	55.29

TABLE 3.2
Test results for ANI7.

smooth vectors. The disadvantage of using too many smooth vectors is the increased cost of the SVD calculations, and this is reflected in the setup timings.

Table 3.3 shows test results for increasing problem sizes, for both the isotropic and anisotropic problems. Again, 3 basis vectors were used for each aggregate, constructed from 12 smooth vectors, with each smooth vector generated using 36 SGS relaxation steps. The results show that grid-independent convergence is not present, particularly for the anisotropic problems. It is expected that for larger problems, a larger number of smoothing steps for the sample vectors is required to achieve a convergence rate comparable to that for smaller problems.

Table 3.4 shows test results when a variable number of basis vectors is used for each aggregate. These tests were performed using a MATLAB code which had this functionality, but no timings are available. The test problem was ANI5 (see Table 3.3 for matrix information), and four levels were used in the method. We used a block of 12 sample vectors, each constructed from 3 SGS steps. The number of basis vectors was chosen based on δ (defined in Section 2.4); the singular vectors that were used were those corresponding to singular values larger than δ times the largest singular value for each aggregate, but no more than 3 basis vectors were used. Table 3.4 shows that a savings in storage can be achieved with a small impact on convergence rate, and in some cases a savings in time may be possible due to a lower operator complexity ($\delta = 0.3$ case).

3.2. Using low-energy vectors from the Lanczos method. The Lanczos method can generate approximations to the extremal eigenvalues and eigenvectors

	Equations	Nonzeros	Levels	Iterations	Time (s)		
					Setup	Solve	Total
UNI4	3321	22761	2	7	0.25	0.03	0.28
UNI5	13041	90321	3	7	1.25	0.11	1.36
UNI6	51681	359841	4	7	5.57	0.49	6.06
UNI7	205761	1436481	5	13	23.12	3.49	26.61
ANI4	3321	22761	2	13	0.24	0.05	0.29
ANI5	13041	90321	3	22	1.30	0.32	1.62
ANI6	51681	359841	4	44	6.24	2.86	9.10
ANI7	205761	1436481	5	89	25.97	23.77	49.74

TABLE 3.3

Test for increasing problem sizes. Grid-independent convergence is not present, particularly for the anisotropic problems. It is expected that for larger problems, a larger number of smoothing steps are required to achieve a convergence rate comparable to that for smaller problems.

δ	Grid complexity	Operator complexity	Iterations
0.0	1.40	3.03	46
0.3	1.35	2.59	49
0.4	1.31	2.28	59
0.5	1.27	1.95	75

TABLE 3.4

Test results for ANI5 using a variable number of basis vectors per aggregate. As the grid and operator complexities improve by adaptively using fewer basis vectors, the iteration counts increase.

of the fine grid matrix. For SPD problems, it is natural to use the Ritz vectors corresponding to the smallest Ritz values to construct the smooth-vector interpolation operator. If a large number of Lanczos steps are desired, however, the method can be costly both in terms of storage (of the Lanczos vectors) and computation (forming the Ritz vectors).

We note also that SGS relaxation can be applied to a *block* of vectors, which is computationally very efficient. Block versions of the Lanczos method, on the other hand, generate poorer low-energy Ritz vectors, depending on the block size.

Tables 3.5 and 3.6 show test results using low-energy Ritz vectors for UNI7 and ANI7, respectively. Three basis vectors were used for each aggregate and 5 levels were used in the multigrid method.

The tables show budgets of either 36 or 72 matrix-vector multiplies. (A budget of 72 corresponds to 36 SGS smoothing steps, but the computational cost is greater.) A different number of low-energy Ritz vectors were constructed. The results show that using more Ritz vectors will improve the convergence rate, but the total time may not be improved, due to the increased setup cost.

3.3. Plane strain problem. We briefly consider a problem that strictly requires multiple basis vectors per aggregate, a plane strain problem on a square. The 3 rigid body modes are known from the geometry and grid for this problem and span the near-nullspace of the PDE operator.

The problem was discretized with 217×217 linear quadrilateral elements (93,312 equations). The nodes were aggregated regularly, using 3×3 node aggregates. Ten smooth vectors were used, with 10 SGS relaxation steps to generate each vector. Three basis vectors were used per aggregate. The multigrid method used 4 levels.

The original problem matrix A was generated, as well as a scaled matrix, DAD , with $D = \text{diag}(10^{d_1}, 10^{d_2}, \dots)$ and the real numbers d_i were selected randomly from

Matvec budget	Smooth vectors	Iterations	Time (s)		
			Setup	Solve	Total
36	3	50	10.57	12.77	23.34
	4	45	11.24	11.53	22.77
	6	41	12.19	10.50	22.69
	9	36	13.83	9.24	23.07
	12	33	14.73	8.53	23.26
72	3	26	12.11	6.74	18.85
	4	24	12.75	6.24	18.99
	6	21	14.02	5.52	19.54
	9	21	15.63	5.48	21.11
	12	20	17.19	5.26	22.45

TABLE 3.5

Test results for UNI7, using low-energy Ritz vectors.

Matvec budget	Smooth vectors	Iterations	Time (s)		
			Setup	Solve	Total
36	3	300	15.30	80.69	95.99
	4	289	15.99	77.72	93.71
	6	282	17.30	75.89	93.19
	9	280	18.27	75.72	93.99
	12	281	19.44	75.61	95.05
72	3	204	15.49	54.33	69.82
	4	184	16.51	49.28	65.79
	6	176	17.75	47.12	64.87
	9	173	19.68	46.40	66.08
	12	170	21.22	45.45	66.67

TABLE 3.6

Test results for ANI7, using low-energy Ritz vectors.

(0, 6).

Table 3.7 shows the results for the original problem matrix and the scaled matrix, using both the rigid body modes to construct the interpolation operator, and the algebraically smooth vectors. Interpolation using the rigid body modes is ideal for the original matrix, but is entirely inappropriate for the scaled matrix. On the other hand, by using the algebraically smooth vectors, both problems can be solved. As expected, however, for the original matrix, the smooth-vector interpolation does not perform as well as interpolating using the rigid body modes.

4. Concluding Remarks. The smooth-vector interpolation operator presented in this paper is constructed to interpolate samples of algebraically smooth error. The interpolation operator is thus matrix-dependent but particular entries in the matrix are not needed once the aggregates have been chosen. When the budget for generating the sample vectors is fixed, it can be beneficial to use more samples than the number of basis vectors, even if each sample vector has more energy. This technique is particularly advantageous when many basis vectors are required, since additional vectors are available (from the singular value decomposition) at no additional cost. Compared to other methods such as spectral AMG [9], this method does not require that the stiffness matrices for each aggregate be available. Generating the sample vectors may also be very efficient, if blocks of vectors are relaxed simultaneously, but this was not tested in this paper.

We note that the performance of the method cannot be better than using the exact near-nullspace vectors if they are known and available. Another disadvantage

Poisson ratio	Interpolation using rigid body modes		Smooth-vector interpolation	
	Original	Scaled	Original	Scaled
.30	13	200+	34	64
.40	18	200+	49	97
.45	24	200+	87	161

TABLE 3.7

PCG iteration counts for plane strain problem, original and scaled matrices. A maximum of 200 iterations was used.

of the method is that for larger problems, smoother sample vectors are required to maintain the same rate of convergence. Thus it may be necessary to use a multigrid method to help generate the sample vectors themselves, as in adaptive multilevel methods [5, 12].

We note that it is also possible to perform the node aggregation using samples of algebraically smooth error, instead of using matrix entries. As in [10], nodes may be aggregated if the samples of smooth error show a strong coupling between the nodes. Incorporating such a strategy may help develop multigrid methods that are matrix-free.

Acknowledgments. The author wishes to thank Marian Brezina, Rob Falgout, Oren Livne, Scott MacLachlan, Tom Manteuffel, Steve McCormick, Charles Tong, and Panayot Vassilevski.

REFERENCES

- [1] J. H. Bramble, J. E. Pasciak, J. Wang, and J. Xu. Convergence estimates for multigrid algorithms without regularity assumptions. *Math. Comp.*, 57:23–45, 1991.
- [2] A. Brandt. Algebraic multigrid theory: The symmetric case. *Appl. Math. Comput.*, 19:23–56, 1986.
- [3] A. Brandt. Multiscale scientific computation: Review 2000. *Electronic Transactions on Numerical Analysis*, 10:1–20, 2000.
- [4] M. Brezina, A. J. Cleary, R. D. Falgout, V. E. Henson, J. E. Jones, T. A. Manteuffel, S. F. McCormick, and J. W. Ruge. Algebraic multigrid based on element interpolation (AMGe). *SIAM J. Sci. Comput.*, 22:1570–1592, 2000.
- [5] M. Brezina, R. Falgout, S. MacLachlan, T. Manteuffel, S. McCormick, and J. Ruge. Adaptive smoothed aggregation (α SA). *SIAM J. Sci. Comput.*, to appear.
- [6] M. Brezina, C. Heberton, J. Mandel, , and P. Vanek. An iterative method with convergence rate chosen a priori. Technical Report UCD/CCM 140, University of Colorado at Denver, April 1999.
- [7] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A Multigrid Tutorial*. SIAM Books, Philadelphia, 2000. Second edition.
- [8] V. E. Bulgakov. Iterative aggregation technique for large-scale finite element analysis of mechanical systems. *Computers & Structures*, 52:829–840, 1994.
- [9] T. Chartier, R. D. Falgout, V. E. Henson, J. Jones, T. Manteuffel, S. McCormick, J. Ruge, and P. S. Vassilevski. Spectral AMGe (ρ AMGe). *SIAM J. Sci. Comput.*, 25:1–26, 2003.
- [10] E. Chow. An unstructured multigrid method based on geometric smoothness. *Num. Lin. Alg. Appl.*, 10:401–421, 2003.
- [11] J. Fish and V. Belsky. Generalized aggregation multilevel solver. *International Journal for Numerical Methods in Engineering*, 40:4341–4361, 1997.
- [12] Scott MacLachlan. *Improving Robustness of Multiscale Methods (draft)*. PhD thesis, University of Colorado at Boulder, 2004.
- [13] J. Mandel. Iterative methods for p-version finite elements: Preconditioning thin solids. *J. Comput. Meth. Appl. Mech. Engrg.*, 133:247–257, 1996.
- [14] J. W. Ruge and K. Stüben. Algebraic multigrid (AMG). In S. F. McCormick, editor, *Multigrid Methods*, volume 3 of *Frontiers in Applied Mathematics*, pages 73–130. SIAM, Philadelphia, PA, 1987.

- [15] P. Vanek. Acceleration of convergence of a two-level algorithm by smoothing transfer operators. *Applications of Mathematics*, 37:265–274, 1992.
- [16] P. Vanek, J. Mandel, and M. Brezina. Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems. *Computing*, 56:179–196, 1996.
- [17] W. L. Wan, T. F. Chan, and B. Smith. An energy-minimizing interpolation for robust multigrid methods. *SIAM J. Sci. Comput.*, 21:1632–1649, 2000.
- [18] J. Xu. Iterative methods by space decomposition and subspace correction: A unifying approach. *SIAM Review*, 34:581–613, 1992.