JOURNAL OF COMPUTATIONAL PHYSICS **125,** 279–292 (1996)
ARTICLE NO. 0094

On the Algebraic Multigrid Method

QIANSHUN CHANG,*† YAU SHU WONG,‡ AND HANQING FU†

Institute of Applied Mathematics, The Chinese Academy of Sciences, Beijing, 100080, China; †Laboratory of Computational Physics, Institute of Applied Physics and Computational Mathematics, Beijing, 100088, China; ‡Department of Mathematical Sciences, University of Alberta, Edmonton, Alberta, Canada T6G 2G1

Received January 9, 1995; revised August 7, 1995

New formulations for the algebraic multigrid (AMG) method are presented. A new interpolation operator is developed, in which the weighting could be negative. Numerical experiments demonstrate that the use of negative interpolation weights is necessary in some applications. New approaches to construct the restriction operator and the coarse-grid equations are discussed. Two new AMG methods are proposed. Theoretical study and convergence analysis of the AMG methods are presented. The main contributions of this paper are to improve the convergence rate and to extend the range of applications of an AMG method. Numerical experiments are reported for matrix computations that resulted from partial differential equations, signal processing, and queueing network problems. The success of the proposed new AMG algorithms is clearly demonstrated by applications to non-diagonally dominant matrix problems for which the standard AMG method fails to converge.

1. INTRODUCTION

The multigrid method has been applied widely in many fields. The main advantage of this method is its asymptotically optimal convergence, i.e., the computational work required to achieve a fixed accuracy is proportional to the number of discrete unknowns [1]. However, the standard multigird solver assumes some underlying geometrical structures, such as grids etc. Consequently, users need to be familiar with the multigrid principles and special code has to be composed for solving different problems. Furthermore, the application of the standard multigrid method is difficult or impossible for many kinds of problems, for example, problems with complex domain, problems using non-uniform coarsening procedure, and purely discrete problems.

The algebraic multigrid (AMG) method is designed to utilize the principle of the geometrically oriented multigrid (GMG) method to obtain a fast and automatic solution procedure for matrix computations. The basic idea of an AMG method was first introduced by Brandt, McCormick, and Ruge [2], and the method was subsequently developed by Ruge and Stüben. An efficient AMG algorithm for M-matrices is described in [3]. In [4–6], we improve the interpolation operator and present different algorithms to construct the coarse grid equations. The comparison between the GMG and AMG methods is given in Table I.

In this paper, we present another approach to construct the coarse grid equations and compare the performance of various AMG algorithms. The new AMG methods improve the convergence rate and extend the range of applications of an AMG method. In Section 2, the basic AMG algorithm is described. Three AMG methods are described in Section 3. In Section 4, a theoretical analysis of convergence for the AMG methods is presented. Computational results are then reported in Section 5. Finally, summary and concluding remarks are given in Section 6.

2. THE BASIC AMG ALGORITHM

Consider the system of linear equations

$$AU = F, \tag{2.1}$$

where $A = (a_{ij})_{n\times n}$, $U = (u_1, u_2, ..., u_n)^{\mathrm{T}}$, $F = (f_1, f_2, ... f_n)^{\mathrm{T}}$. A sequence of systems of equations is generated as

$$A^m U^m = F^m, \tag{2.2}$$

where $A^m = (a^m_{ij})_{n_m\times n_m}$, $U^m = (u^m_1, u^m_2, ..., u^m_{n_m})^{\mathrm{T}}$, $F^m = (f^m_1, f^m_2, ..., f^m_{n_m})^{\mathrm{T}}$, $m = 1, 2, ..., M$, $n = n_1 > n_2 > \cdots > n_m$, $A^1 = A$, $U^1 = U$, $F^1 = F$. These equations formally play the same role as the coarse grid equations defined in the GMG method. A grid Ω^m can be regarded as a set of unknowns u^m_j $(1 \le j \le n_{\mathrm{m}})$.

The coarse grid Ω^{m+1} is chosen as a subset in Ω^m, which is denoted by C^m. The remainder subset $\Omega^m - C^m$ is denoted by F^m. A point i is said to be strongly connected to j, if

$$(-a^m_{ij}) \ge \theta_0 \cdot \max_{k\neq i}(-a^m_{ik}), \quad 0 < \theta_0 \le 1. \tag{2.3}$$

Let S^m_i denote the set of all strongly connection points

0021-9991/96 $18.00

TABLE I

Comparison between the GMG and the AMG

	GMG method	AMG method
Solved problem	Continuous problem	Linear system of algebraic equations
Used information	Geometrical structure of the problem	Only entries of the matrix
Smoothing operator	vary for each problem	Fixed
Program	Compose program for each problem	Only one program for different problems
Efficiency	Very good	Good

of the point i and let $C_i^m = C^m \cap S_i^m$. In general, C^m and F^m are chosen so that the following criteria are satisfied:

(C1) For each $i \in F^m$, each point $j \in S_i^m$ should be either in C_i^m or strongly connected to at least one point in C_i^m

(C2) C^m should be a maximal subset of all points with the property that no two C-points are strongly connected to each other.

In practice, it is impossible to strictly satisfy both criteria (C1) and (C2) for all systems of equations. However, (C2) is generally used as a guideline to construct C^m such that condition (C1) is held. Now define the set of points which are strongly connected to i by $S_i^T = \{j : i \in S_j^m\}$, and for a set P, let $|P|$ denote the number of elements in P. The following two-part process is suggested by Ruge and Stüben [3]. First, a basic choice for the C-point is performed as follows:

(1) Set $C^m = \emptyset$, $F^m = \emptyset$, $U = \Omega^m$, and $\lambda = |S_i^T|$ for all i,

(2) Pick an $i \in U$ with maximal λ_i, and set $C^m = C^m \cup \{i\}$, $U = U - \{i\}$,

(3) For all $j \in S_i^T \cap U$, perform (4) and (5),

(4) Set $F^m = F^m \cup \{j\}$ and $U = U - \{j\}$,

(5) For all $l \in S_j^m \cap U$, set $\lambda_l = \lambda_l + 1$,

(6) For all $j \in S_i^m \cap U$, set $\lambda_j = \lambda_j - 1$,

(7) If $U = \emptyset$, stop. Otherwise, go to (2).

The first part attempts to enforce the criterion (C2) by distributing the C-points uniformly over the grid. The second part is combined with the computation of interpolation weights, in which the tentative F-points resulting from the first part are tested to ensure that the criterion (C1) holds. The new C-points will be added as necessary. It should be noted that the steps (1)–(7) need only $O(n)$ operations when an efficient implementation is used.

After the coarse grid Ω^{m+1} is chosen, the interpolation operators I_{m+1}^m, restriction operator I_m^{m+1}, and coarse grid equations A^{m+1} can be constructed in the preparation phase of an AMG method. The smoothing operator G^m is chosen as a fixed iterative procedure, for example, Gauss–Seidel or Jacobi iteration. Once the above five components Ω^m, I_{m+1}^m, I_m^{m+1}, A^m, G^m are known, a multigrid cycling procedure can be set up in the usual manner [1].

In general, there are two phases required in an AMG method: (1) the preparation phase or the setup phase, in which the five components Ω^m, I_{m+1}^m, I_m^{m+1}, A^m, and G^m are constructed; (2) the solver phase, i.e., the multigrid cycling procedure, in which the system of equations is solved.

3. AMG METHODS

In this section we describe three AMG methods. The methods differ from each other depending on the choice of the interpolation operators and different algorithms to construct the coarse grid equations and the restriction operators.

3.1. *The Interpolation Operators*

Let $N_i^m = \{j \in \Omega^m : j \neq i, a_{ij}^m \neq 0\}$ denote the neighborhood of a point $i \in \Omega^m$, and $D_i = N_i^m - C_i$, $D_i^s = D_i \cap S_i$, $D_i^w = D_i - D_i^s$.

Each variable in C^m interpolates directly from the corresponding variable in Ω^{m+1} with a weighting of unity, and each variable $i \in F^m$ interpolates from the smaller set C_i^m.

In [3], Ruge and Stüben present the following interpolation formula for the variable $i \in F^m$:

$$e_i^m = \sum_{j \in C_i^m} w_{ij}^m e_j^{m+1}, \quad \forall i \in F^m, \tag{3.1}$$

and

$$w_{ij}^m = -\frac{1}{a_{ii}^m + \sum_{k \in D_i^w} a_{ik}^m}\left[a_{ij}^m + \sum_{k \in D_i^s} a_{ik}^m a_{kj}^m \Big/ \sum_{l \in C_i^m} a_{kl}^m\right]. \tag{3.2}$$

The formulae are efficient for the M-matrices. However, when (3.1)–(3.2) is applied to general matrix problems with positive and negative off-diagonal entries, the denominator $\sum_{l \in C_i^m} a_{kl}^m$ may be small or zero. Consequently, the AMG method could fail during the setup phase.

In [4, 6], we present a new interpolation formula. Now, instead of using the inequality (2.3), we define a point i which is considered to be strongly connected to j, if

$$|a_{ij}^m| \geq \theta_0 \cdot \max_{k \neq i} |a_{ik}^m|, \quad 0 < \theta_0 \leq 1. \tag{3.3}$$

We then introduce the following geometric assumptions:

(G1) In the neighborhood N_i^m of a point $i \in \Omega^m$, the larger the quantity $|a_{ij}^m|$ is, the closer point j is to the point i.

(G2) An algebraically smooth error is also geometrically smooth between points i and j if $a_{ij}^m < 0$ or $|a_{ij}^m|$ is small, and it is geometrically oscillating if $a_{ij}^m > 0$ is large.

Because the error e_i^m to be interpolated in an AMG method is obtained after a smoothing process, it gives

$$a_{ii}^m e_i^m + \sum_{j \in N_i^m} a_{ij}^m e_j^m = d_i^m \approx 0, \quad \forall i \in \Omega^m,$$

which can be rewritten as

$$a_{ii}^m e_i^m + \sum_{k \in C_i^m} a_{ik}^m e_k^m + \sum_{j \in D_i^s} a_{ij}^m e_j^m + \sum_{j \in D_i^w} a_{ij}^m e_j^m \approx 0, \quad \forall i \in \Omega^m. \tag{3.4}$$

Let

$$\xi_{ij}^m = \frac{-\sum_{k \in C_i^m} a_{jk}^m}{\sum_{k \in C_i^m} |a_{jk}^m|}, \quad \eta_{ij}^m = \frac{|a_{ji}^m|\, l_{ij}^m}{\sum_{k \in C_i^m} |a_{jk}^m|},$$

where l_{ij}^m denotes the number of elements in a set $S_{ij}^m = \{k : k \in C_i^m,\ a_{jk}^m \neq 0\}$. The quantity ξ_{ij}^m indicates whether there is a large positive entry a_{jk}^m for $k \in S_{ij}^m$. By means of the geometrical assumption (G2), it can be shown that error between point i and j is geometrically smooth and the extrapolation formula can be applied if $\xi_{ij}^m \geq 0.5$ and $a_{ij}^m < 0$. The quantity η_{ij}^m is the ratio of $|a_{ji}^m|$ to the average value $(1/l_{ij}^m) \sum_{k \in C_i^m} |a_{jk}^m|$. It then follows from the first geometrical assumption (G1) that η_{ij}^m approximately gives the ratio of the distance between j and i to the average distance between j and the elements of the set S_{ij}^m. Let

$$g_{jk}^m = \frac{|a_{jk}^m|}{\sum_{k \in C_i^m} |a_{jk}^m|}, \quad j \in D_i^m, k \in C_i^m.$$

Consequently, the following approximations are used in (3.4):

(1) For points $j \in D_i^w$, we have

$$e_j^m = \begin{cases} e_i^m, & \text{if } l_{ij}^m = 0, a_{ij}^m < 0, \\ -\mathrm{e}_i^m, & \text{if } l_{ij}^m = 0, a_{ij}^m > 0, \\ 2\sum_{k \in C_i^m} g_{jk}^m e_k^m - e_i^m, & \text{if } l_{ij}^m > 0, \xi_{ij}^m \geq 0.5, a_{ij}^m < 0, \\ \sum_{k \in C_i^m} g_{jk}^m e_k^m, & \text{otherwise.} \end{cases} \tag{3.5}$$

(2) For points $j \in D_i^s$, more accurate approximations are presented,

$$e_j^m = \begin{cases} 2\sum_{k \in C_i^m} g_{jk}^m e_k^m - e_i^m, & \text{if } \eta_{ij}^m < 0.75, \xi_{ij} \geq 0.5, a_{ij}^m < 0, \\ \frac{1}{2}\left(\sum_{k \in C_i^m} g_{jk}^m e_k^m + e_i^m\right), & \text{if } \eta_{ij}^m > 2, \xi_{ij}^m \geq 0.5, a_{ij}^m < 0, \\ \sum_{k \in C_i^m} g_{jk}^m e_k^m, & \text{otherwise.} \end{cases} \tag{3.6}$$

Remark. Substituting (3.5)–(3.6) into (3.4) is equivalent to modifying the coefficients in (3.4) by combining the following steps:

step (1) add $-|a_{ij}^m|$ to a_{ii}^m, $\forall j \in D_i^{(1)}$, which is equivalent to e_j^m being replaced by e_i^m or $-e_i^m$;

step (2) add $a_{ij}^m g_{jk}^m$ to a_{ik}^m, $\forall k \in C_i^m$, $\forall j \in D_i^{(2)}$, which is equivalent to e_j^m being approximated by $\sum_{k \in C_i^m} g_{jk}^m e_k^m$;

step (3) add $2a_{ij}^m g_{jk}^m$ to a_{ik}^m, $\forall k \in C_i^m$, and subtract a_{ij}^m from a_{ii}^m, $\forall j \in D_i^{(3)}$, which is equivalent to e_j^m being approximated by $2\sum_{k \in C_i^m} g_{jk}^m e_k^m - e_i^m$;

step (4) add $0.5a_{ij}^m g_{jk}^m$ to a_{ik}^m, $\forall k \in C_i^m$, and add $0.5a_{ij}^m$ to a_{ii}^m, $\forall j \in D_i^{(4)}$, which is equivalent to e_j^m being approximated by $0.5\left(\sum_{k \in C_i^m} g_{jk}^m e_k^m + e_i^m\right)$;

where $D_i^{(l)} \equiv \{j : j \in D_i, e_j \text{ is eliminated by the corresponding step } (l)\}$ $(l = 1, 2, 3, 4)$.

Thus, a new interpolation formula derived from (3.4) is given by

$$e_i^m = \sum_{j \in C_i^m} w_{ij}^m e_j^{m+1}, \quad \forall i \in F^m, \tag{3.7}$$

where

$$w_{ik}^m = \frac{-\bar{a}_{ik}^m}{\bar{a}_{ii}^m}, \quad k \in C_i^m, \tag{3.8}$$

$$\bar{a}_{ii}^m = a_{ii}^m - \sum_{j \in D_i^{(1)}} |a_{ij}^m| - \sum_{j \in D_i^{(3)}} a_{ij}^m + 0.5 \sum_{j \in D_i^{(4)}} a_{ij}^m,$$

$$\bar{a}_{ik}^m = a_{ik}^m + \sum_{j \in D_i^{(2)}} a_{ij}^m g_{jk}^m + 2\sum_{j \in D_i^{(3)}} a_{ij}^m g_{jk}^m + 0.5 \sum_{j \in D_i^{(4)}} a_{ij}^m g_{jk}^m.$$

Remark. The proposed interpolation formula (3.7)–(3.8) should be more accurate than (3.1)–(3.2) used in the standard AMG method because of the following reasons. First, we consider that off-diagonal elements with large absolute values are more important than the others in the interpolation process, whereas only negative elements with large absolute values are being regarded as the strongly connected points by Ruge and Stüben. Second, two geometrical assumptions are introduced in which extrapolation and averaging formulae are taken into account in the interpolation process. This remark will be verified by numerical experiments reported in Section 5. It can also be shown that the new interpolation formula preserves linear functions for M-matrix systems with zero row-sums.

3.2. *The Coarse Grid and Restriction Operators*

A simple approach to define the coarse grid operator A^{m+1} and the restriction operator I_m^{m+1} is by the Galerkin type algorithm [3], in which

$$I_m^{m+1} = (I_{m+1}^m)^{\mathrm{T}} \tag{3.9}$$

and

$$A^{m+1} = I_m^{m+1} A^m I_{m+1}^m. \tag{3.10}$$

The Galerkin type method suggested by Ruge and Stüben will be referred as the first algorithm. The second algorithm discussed in [5, 6] is to use direct approximations based on the fine grid operator A^m to construct A^{m+1} and I_m^{m+1}. Assuming the operator A^m is known, we start from the following equations:

$$a_{ii}^m u_i^m + \sum_{j\in C^m} a_{ij}^m u_j^m + \sum_{j\in F^m} a_{ij}^m u_j^m = f_j^m, \quad i \in C^m. \tag{3.11}$$

In order to derive the coarse grid operator A^{m+1}, the terms associated with u_j^m, $j \in F^m$, in the ith equation, $i \in C^m$, should be approximated. The simplest way is to use the interpolation formula to eliminate all $u_j^m, j \in F^m$. However, the resulting coarse grid operator could not provide a sufficiently accurate correction to an approximate solution in the fine grid. As a consequence of this, it leads to poor convergence for the multigrid method. Alternatively, the terms associated with u_j^m, $j \in F^m$, in the ith equation can be replaced by the jth equation without introducing any error. Even though this operation can not eliminate all $u_j^m, j \in F^m$, it can reduce the magnitude of the coefficient for $u_j^m, j \in F^m$. This particularly works well if the diagonal element is larger than the off-diagonal elements. The new approach for constructing A^{m+1} is thus obtained by the following switching algorithm. The u_j^m, $j \in F^m$, are eliminated by the interpolation formula (3.7)–(3.8) if $|a_{ij}^m/a_{ii}^m| \le \theta_1$ and the u_j^m, $j \in F^m$, are replaced by means of the jth operator when $|a_{ij}^m/a_{ii}^m| > \theta_1$, i.e.,

$$u_j^m = \begin{cases} \sum_{k\in C_j} w_{jk} u_k^m, & \text{if } |a_{ij}^m/a_{ii}^m| \le \theta_1, \\ \dfrac{f_j^m}{a_{jj}^m} - \dfrac{1}{a_{jj}^m} \sum_{k\in N_j} a_{jk}^m u_k^m, & \text{if } |a_{ij}^m/a_{ii}^m| > \theta_1. \end{cases} \tag{3.12}$$

Hence the following new equations are obtained

$$a_{ii}^{(2)} u_i^m + \sum_{j\in C^m} a_{ij}^{(2)} u_j^m + \sum_{j\in F^m} a_{ij}^{(2)} u_j^m = f_i^m - \sum_{j\in F_1^m} \frac{a_{ij}^m}{a_{jj}^m} f_j^m, \; i \in C^m, \tag{3.13}$$

where $F_1^m = \{j : |a_{ij}^m/a_{ii}^m| > \theta_1, j \in F^m\}$. The procedure is repeated until u_j^m, $j \in F^m$, no longer appear in the ith equation, $i \in C^m$. Therefore,

$$a_{ii}^{(L)} u_i^m + \sum_{j\in C^m} a_{ij}^{(L)} u_j^m = f_i^m - \sum_{l=1}^{L-1} \sum_{j\in F_l^m} \frac{a_{ij}^{(l)}}{a_{jj}^m} f_j^m, \quad i \in C^m, \tag{3.14}$$

where $a_{ij}^{(1)} = a_{ij}^m$, $F_l^m = \{j\colon |a_{ij}^{(l)}/a_{ii}^{(l)}| > \theta_1, j \in F^m\}$. The coarse grid operator A^{m+1} is now defined as

$$A^{m+1} = (a_{ij}^{(L)}),$$

and the restriction operator I_m^{m+1} is resulted directly from (3.14), i.e.,

$$f_i^{m+1} = f_i^m - \sum_{l=1}^{L-1} \sum_{j\in F_l^m} \frac{a_{ij}^{(l)}}{a_{jj}^m} f_j^m, \quad i \in \Omega^{m+1}. \tag{3.15}$$

Notice that $I_m^{m+1} \ne (I_{m+1}^m)^{\mathrm{T}}$ for general matrices. Since small elements $a_{ij}^{(L)}$ are introduced during the process of constructing the coarse grid operator, a parameter θ_2 is used so that small elements $a_{ij}^{(L)}$ are ignored in A^{m+1} if $|a_{ij}^{(L)}/a_{ii}^{(L)}| < \theta_2$. In practice, the operator A^m is normalized so that $a_{ii}^m = 1$, $\forall i \in \Omega^m$. Thus division operations are not required in computing the operators I_m^{m+1} and A^{m+1}.

Now, we consider the third approach for the coarse grid operator A^{m+1}. Let an auxiliary matrix $B_m^{m+1} = I_m^{m+1} A^m$ and $B_m^{m+1} = (b_1^{m+1}, b_2^{m+1}, \ldots, b_{n_{m+1}}^{m+1})^{\mathrm{T}}$. Then multiplying the matrix (2.2) by the restriction operator I_m^{m+1}, we get

$$I_m^{m+1} A^m U^m = I_m^{m+1} F^m,$$

which is equivalent to

$$B_m^{m+1} U^m = I_m^{m+1} F^m, \quad B_m^{m+1} \equiv I_m^{m+1} A^m. \tag{3.16}$$

In order to obtain the coarse grid operator A^{m+1}, u_j^m, $j \in F^m$, in the equality (3.16) need to be approximated. It follows directly from (3.16) that

$$(B_m^{m+1} U^m)_i = b_i^{m+1} U^m = (I_m^{m+1} F^m)_i \tag{3.17}$$

and

$$b_i^{m+1} U^m = \sum_j b_{ij}^{m+1} u_j^m = \sum_{j\in C^m} b_{ij}^{m+1} u_j^m + \sum_{j\in F^m} b_{ij}^{m+1} u_j^m.$$

The u_j^m, $j \in F^m$, are then eliminated using the interpolation formula (3.7)–(3.8) if $|b_{ij}^{m+1}| > \theta_1 \cdot a_{ii}^m$, and u_j^m, $j \in F^m$, are replaced by u_i^m if $|b_{ij}^{m+1}| \le \theta_1 \cdot a_{ii}^m$; i.e.,

$$u_j^m = \begin{cases} \sum_{k \in C_j} w_{jk} u_k^m, & \text{if } |b_{ij}^{m+1}| > \theta_1 \cdot a_{ii}^m, \\ u_i^m, & \text{if } |b_{ij}^{m+1}| \le \theta_1 \cdot a_{ii}^m. \end{cases} \quad (3.18)$$

Substituting (3.18) into (3.17), we get

$$A^{m+1} U^{m+1} = I_m^{m+1} F^m \equiv F^{m+1}. \quad (3.19)$$

3.3. *Three AMG Methods*

Two formulae for the interpolation operator I_{m+1}^m were presented in Section 3.1, and different algorithms to construct the coarse grid operator A^{m+1} and the restriction operator I_m^{m+1} were discussed in Section 3.2. Depending upon the choice of I_{m+1}^m, I_m^{m+1}, and A^{m+1}, it leads to a particular version of an AMG method. To avoid confusion between various algorithms, we shall use the following defintions to signify each method:

Method	Interpolation	Restriction	Coarse grid equation
I	(3.1)–(3.2)	(3.9)	(3.10)
II	(3.7)–(3.8)	(3.9)	(3.16)–(3.19)
III	(3.7)–(3.8)	(3.11)–(3.15)	

Notice that, Method I is the standard AMG algorithm proposed by Ruge and Stüben [3], in which the interpolation formula is given by (3.1)–(3.2) and a Galerkin-type algorithm is used to define I_m^{m+1} and A^{m+1}. Methods II and III are the two new AMG methods presented in this paper where a more accurate interpolation (3.7)–(3.8) is applied. In Method II I_{m+1}^m and A^{m+1} are obtained from (3.16)–(3.19) and (3.9); (3.11)–(3.15) are used in Method III.

4. CONVERGENCE ANALYSIS

In this section we consider convergence analysis for the AMG methods. Theorems 1–4 are due to Huang [11], Ruge and Stüben [3], and Theorems 5–8 give a theoretical analysis for the AMG method used in conjunction with the new interpolation formula (3.7)–(3.8) proposed in the previous section. We shall prove that the two-level AMG method is convergent, and the result can be extended to multi-level AMG when certain conditions are satisfied. The bound on the convergence factor is shown in Theorem 7.

Let

$$G^m, A^{m+1} = I_m^{m+1} A^m I_{m+1}^m, \qquad T^m = I^m - I_{m+1}^m (A^{m+1})^{-1} I_m^{m+1} A^m, \quad (4.1)$$

denote the smoothing operator, coarse grid operator, and $(m, m+1)$ the coarse grid correction operator, respectively. In addition to the Euclidean inner product $(\cdot, \cdot)$, three different inner products

$$(u, v)_0 = (Du, v), \quad (u, v)_1 = (Au, v), \quad (u, v)_2 = (D^{-1} Au, Av),$$

are defined, together with the corresponding norms $\| \cdot \|_i$ $(i = 0, 1, 2)$.

First, we describe the following theorems, which are given by Ruge and Stüben in [3].

THEOREM 1. *Let $A^m > 0$ and define, with any positive vector $W^m = (w_i^m)$,*

$$\gamma_-^{(m)} = \max_i \left\{ \frac{1}{w_i^m a_{ii}^m} \sum_{j<i} w_j^m |a_{ij}^m| \right\},$$

$$\gamma_+^{(m)} = \max_i \left\{ \frac{1}{w_i^m a_{ii}^m} \sum_{j>i} w_j^m |a_{ij}^m| \right\}.$$

Then the Gauss–Seidel relaxation satisfies

$$\|G^m e^m\|_1^2 \le \|e^m\|_1^2 - \alpha_m \|e^m\|_2^2, \quad \alpha_m > 0. \quad (4.2)$$

THEOREM 2. *Let $A^m > 0$ and $\gamma_0^m \ge \rho((D^m)^{-1} A)$. Then Jacobi relaxation with parameter $0 < \omega^m \le 2/\gamma_0^m$ satisfies (4.2) if $\alpha_m \le \omega^m(2 - \omega^m \gamma_0^m)$.*

THEOREM 3. *Let $A^m > 0$ and let $G^m > 0$ satisfy (4.2). Suppose that the interpolation I_{m+1}^m has a full rank and that, for each e^h,*

$$\min \|e^m - I_{m+1}^m e^{m+1}\|_0^2 \le \beta_m \|e^m\|_1^2, \quad (4.3)$$

with $\beta_m > 0$ independent of e^m. Then $\beta_m \ge \alpha_m$, and the $(m, m+1)$ two-level convergence factor satisfies:

$$\|G^m T^m\|_1 \le \sqrt{1 - \alpha_m/\beta_m}.$$

In [11], Huang extends the results of Ruge and Stüben and the following theorem is presented.

THEOREM 4. *Let $A^m > 0$, and assume for any given set C^m of the C-points, that I_{m+1}^m is of the form (3.1) with*

$S_i^m \le 1$, $S_i^m = \sum_{k \in C_i^m} |w_{ik}^m|$. Then (4.3) is satisfied if the following two inequalities hold with $\beta_m > 0$ independent of e^m:

$$\sum_{i \in F^m} \sum_{k \in C^m} a_{ii}^m |w_{ik}^m| \left(e_i^m - \frac{w_{ik}^m}{|w_{ik}^m|} e_k^m \right)^2 \le \frac{\beta_m}{2} \sum_i \sum_{j \ne i} |a_{ij}^m| \left(e_i^m + \frac{a_{ij}^m}{|a_{ij}^m|} e_j^m \right)^2, \tag{4.4}$$

$$\sum_{i \in F^m} a_{ii}^m (1 - S_i^m)(e_i^m)^2 \le \beta_m \sum_i \left(a_{ii}^m - \sum_{j \ne i} |a_{ij}^m| \right) (e_i^m)^2. \tag{4.5}$$

Remark. It is easy to verify that Theorem 4 holds if the set C^m is replaced by C_i^m. We will use the form of the Theorem 4 with C_i^m.

THEOREM 5. *Let $A^m > 0$ and assume A^m is a weakly diagonally dominant matrix, then*

$$S_i^m \le 1 \tag{4.6}$$

for the interpolation formulae (3.7)–(3.8), where S_i^m are defined in Theorem 4.

Proof. Observe that

$$S_i = \sum_{k \in C_i} |w_{ik}| = \sum_{k \in C_i} \frac{|\bar{a}_{ik}|}{\bar{a}_{ii}}.$$

Using (3.8) and $\sum_{k \in C_i} g_{ik} = 1$, we obtain

$$S_i \le \frac{1}{a_{ii} - \sum_{j \in D_i^{(1)}} |a_{ij}| - \sum_{j \in D_i^{(3)}} a_{ij} + 0.5 \sum_{j \in D_i^{(4)}} a_{ij}} \times \left[\sum_{j \in C_i} |a_{ij}| + \sum_{j \in D_i^{(2)}} |a_{ij}| + 2 \sum_{j \in D_i^{(3)}} |a_{ij}| + 0.5 \sum_{j \in D_i^{(4)}} |a_{ij}| \right]. \tag{4.7}$$

The weakly diagonally dominant matrix A^m and $a_{ij} < 0$, $\forall j \in D_i^{(3)} \cup D_i^{(4)}$ implies that

$$a_{ii} \ge \sum_{j \in N_i} |a_{ij}| = \sum_{j \in C_i} |a_{ij}| + \sum_{j \in D_i^{(1)}} |a_{ij}| + \sum_{j \in D_i^{(2)}} |a_{ij}| + \sum_{j \in D_i^{(3)}} |a_{ij}| + \sum_{j \in D_i^{(4)}} |a_{ij}|;$$

i.e.,

$$\sum_{j \in C_i} |a_{ij}| + \sum_{j \in D_i^{(2)}} |a_{ij}| + 2 \sum_{j \in D_i^{(3)}} |a_{ij}| + 0.5 \sum_{j \in D_i^{(4)}} |a_{ij}| \le a_{ii} - \sum_{j \in D_i^{(1)}} |a_{ij}| - \sum_{j \in D_i^{(3)}} a_{ij} + 0.5 \sum_{j \in D_i^{(4)}} a_{ij}.$$

Thus, $S_i \le 1$ is proved.

THEOREM 6. *Let $\xi^m > 0$ and $0 \le \gamma^m < 1$ be fixed constants. Assuming the C-points are picked in such a way that, for each $i \in F^m$ and $j \in C_i^m$, there are*

$$\sum_{j \in D_i^m} |a_{ij}^m| \le \gamma^m a_{ii}^m, \quad \sum_{j \in D_i^m} |a_{ij}^m| \le \xi^m \max_{j \in C_i^m} a_{ij}^m, \quad \frac{w_{ij}^m}{|w_{ij}^m|} = -\frac{a_{ij}^m}{|a_{ij}^m|}. \tag{4.8}$$

Then, if A^m is a symmetric positive definite matrix with weakly diagonal dominance, the interpolation formula (3.7)–(3.8) *satisfy the inequalities* (4.4) *and* (4.5) *with* $\beta = (2/(1 - \gamma^m))(1 + 2\xi^m/\theta_0)$.

Proof. Observe the inequality (4.5)

$$\begin{aligned}
a_{ii}(1 - S_i) &\le a_{ii} - \frac{a_{ii}}{\bar{a}_{ii}} \left(\sum_{j \in C_i} |a_{ij}| + \sum_{j \in D_i^{(2)}} |a_{ij}| + 2 \sum_{j \in D_i^{(3)}} |a_{ij}| + 0.5 \sum_{j \in D_i^{(4)}} |a_{ij}| \right) \\
&= \frac{a_{ii}}{\bar{a}_{ii}} \left(a_{ii} - \sum_{j \in D_i^{(1)}} |a_{ij}| - \sum_{j \in D_i^{(3)}} a_{ij} + 0.5 \sum_{j \in D_i^{(4)}} a_{ij} \right. \\
&\quad \left. - \sum_{j \in C_i} |a_{ij}| - \sum_{j \in D_i^{(2)}} |a_{ij}| - 2 \sum_{j \in D_i^{(3)}} |a_{ij}| - 0.5 \sum_{j \in D_i^{(4)}} |a_{ij}| \right) \\
&= \frac{a_{ii}}{\bar{a}_{ii}} \left(a_{ii} - \sum_{j \ne i} |a_{ij}| \right) \\
&= \frac{a_{ii}}{a_{ii} - \sum_{j \in D_i^{(1)}} |a_{ij}| - \sum_{j \in D_i^{(3)}} a_{ij} + 0.5 \sum_{j \in D_i^{(4)}} a_{ij}} \left(a_{ii} - \sum_{j \ne i} |a_{ij}| \right) \\
&\le \frac{a_{ii}}{a_{ii} - \gamma a_{ii}} \left(a_{ii} - \sum_{j \ne i} |a_{ij}| \right) \\
&= \frac{1}{1 - \gamma} \left(a_{ii} - \sum_{j \ne i} |a_{ij}| \right).
\end{aligned} \tag{4.9}$$

The inequality (4.5) holds for any $\beta \ge 1/(1 - \gamma)$. For the inequality (4.4), we have

$$\sum_{i\in F}\sum_{k\in C_i} a_{ii}|w_{ik}|\left(e_i-\frac{w_{ik}}{|w_{ik}|}e_k\right)^2$$

$$=\sum_{i\in F}\sum_{k\in C_i}\frac{a_{ii}}{\bar{a}_{ii}}\cdot\left|\left(a_{ik}+\sum_{j\in D_i^{(2)}}a_{ij}g_{jk}-2\sum_{j\in D_i^{(3)}}|a_{ij}|g_{jk}\right.\right.$$

$$\left.\left.-0.5\sum_{j\in D_i^{(4)}}|a_{ij}|g_{jk}\right)\right|\cdot\left(e_i-\frac{w_{ik}}{|w_{ik}|}e_k\right)^2$$

$$\le\sum_{i\in F}\sum_{k\in C_i}\frac{a_{ii}}{\bar{a}_{ii}}\cdot\left(|a_{ik}|+2\sum_{j\in D_i}|a_{ij}|\right)\cdot\left(e_i-\frac{w_{ik}}{|w_{ik}|}e_k\right)^2$$

$$\le\sum_{i\in F}\sum_{k\in C_i}\frac{a_{ii}}{\bar{a}_{ii}}\cdot\left(|a_{ik}|+2\xi\max_{j\in C_i}|a_{ij}|\right)\cdot\left(e_i-\frac{w_{ik}}{|w_{ik}|}e_k\right)^2.$$

The inequality (3.3) implies that $(1/\theta_0)\ |a_{ik}| \ge \max_{k\neq i}|a_{ik}|$, $\forall k \in C_i$. Thus, we obtain

$$\sum_{i\in F}\sum_{k\in C_i} a_{ii}|w_{ik}|\left(e_i-\frac{w_{ik}}{|w_{ik}|}e_k\right)^2$$

$$\le\sum_{i\in F}\sum_{k\in C_i}\frac{a_{ii}}{\bar{a}_{ii}}\left(|a_{ik}|+2\xi\cdot\frac{1}{\theta_0}|a_{ik}|\right)\left(e_i-\frac{w_{ik}}{|w_{ik}|}e_k\right)^2$$

$$\le\sum_{i\in F}\sum_{k\in C_i}\frac{a_{ii}}{\bar{a}_{ii}}\left(1+\frac{2\xi}{\theta_0}\right)|a_{ik}|\left(e_i-\frac{w_{ik}}{|w_{ik}|}e_k\right)^2$$

$$\le\sum_{i\in F}\sum_{k\in C_i}\frac{a_{ii}}{a_{ii}-\gamma a_{ii}}\left(1+\frac{2\xi}{\theta_0}\right)|a_{ik}|\left(e_i-\frac{w_{ik}}{|w_{ik}|}e_k\right)^2$$

$$=\frac{1}{1-\gamma}\left(1+\frac{2\xi}{\theta_0}\right)\sum_{i\in F}\sum_{k\in C_i}|a_{ik}|\left(e_i-\frac{w_{ik}}{|w_{ik}|}e_k\right)^2$$

$$\le\frac{1}{1-\gamma}\left(1+\frac{2\xi}{\theta_0}\right)\sum_{i}\sum_{k\neq i}|a_{ik}|\left(e_i-\frac{w_{ik}}{|w_{ik}|}e_k\right)^2.$$

The inequality (4.4) is satisfied for any $\beta \ge (1/(1-\gamma))\,(1+2\xi/\theta_0)$. Hence the conclusion of the theorem holds for $\beta = (2/(1-\gamma))\,(1+2\xi/\theta_0)$.

THEOREM 7. *Assuming A^m is a symmetric positive definite matrix with weakly diagonal dominance and the C-points are picked in such a way that, for each $i \in F^m$, $j \in C_i^m$, then the conditions (4.8) are satisfied. Suppose that the interpolation formulae (3.7)–(3.8) and the Gauss–Seidel relaxation (or the Jacobi relaxation with parameter $0 < w^m < 2/\gamma_0^m$, $\gamma_0^m \ge \rho((D^m)-1\cdot A^m)$) are used in the AMG method. Then the $(m, m+1)$-two-level AMG algorithm is convergent with factor $\|G^mT^m\|_1 \le \sqrt{1-\alpha_m/\beta_m}$, where $\beta_m = (2/(1-\gamma^m))\,(1+2\xi^m/\theta_0) > \alpha_m$, α_m is given by Theorem 1 or Theorem 2.*

Proof. The theorem follows from Theorem 1–Theorem 6.

THEOREM 8. *Suppose the matrix A^m is symmetric positive definie and weakly diagonally dominant. If the condition (4.8) is satisfied and the interpolation weights of (3.7)–(3.8) satisfy*

$$|a_{ik}^m| \ge \gamma^m|\omega_{ik}^m|a_{ii}^m,\quad \forall i\in F^m, k\in C_i^m, \tag{4.10}$$

then the coarse grid operator A^{m+1} is also symmetric positive definite and weakly diagonally dominant.

Proof. First, it follows from (4.1) that A^{m+1} is symmetric positive definite. Because $w_{ik} = \delta_{ik}$ (if $i, k \in C$) and for the reasons of symmetry, we can rewrite the entries of the coarse grid operator A^{m+1} in the following form:

$$a_{kl}^{m+1}=\sum_{ij}w_{ik}^m a_{ij}^m w_{jl}^m$$

$$=a_{kl}^m+\sum_{i\in F^m}(w_{ik}^m a_{il}^m+w_{il}^m a_{ik}^m)$$

$$+\sum_{i\in F^m}\sum_{j\in F^m}w_{ik}^m a_{ij}^m w_{jl}^m$$

$$=a_{kl}^m+\sum_{i\in F^m}\left[w_{ik}^m\left(a_{il}^m+\frac{1}{2}a_{ii}^m w_{il}^m\right)\right.$$

$$\left.+w_{il}^m\left(a_{ik}^m+\frac{1}{2}a_{ii}^m w_{ik}^m\right)\right]$$

$$+\sum_{i\in F^m}\sum_{\substack{j\in F^m\\ j\neq i}}w_{ik}^m a_{ij}^m w_{jl}^m,$$

where $k, l \in C^m$. By using $w_{ij}^m/|w_{ij}^m| = -a_{ij}^m/|a_{ij}^m|$ in (4.8), we have

$$a_{kk}^{m+1}\ge a_{kk}^m-2\sum_{i\in F^m}|w_{ik}^m|\left(|a_{ik}^m|-\frac{1}{2}a_{ii}^m|w_{ik}^m|\right)$$

$$-\sum_{i\in F^m}\sum_{\substack{j\in F^m\\ j\neq i}}|w_{ik}^m|\,|a_{ij}^m|\,|w_{jk}^m|,$$

and

$$|a_{kl}^{m+1}|\le|a_{kl}^m|+\sum_{i\in F^m}\left[|w_{ik}^m|\left(|a_{il}^m|-\frac{1}{2}a_{ii}^m|w_{il}^m|\right)\right.$$

$$\left.+|w_{il}^m|\left(|a_{ik}^m|-\frac{1}{2}a_{ii}^m|w_{ik}^m|\right)\right]$$

$$+\sum_{i\in F^m}\sum_{\substack{j\in F^m\\ j\neq i}}|w_{ik}^m|\,|a_{ij}^m|\,|w_{jl}^m|.$$

In view of the definition of S_i^m and the symmetry of A^m, we obtain

$$
\begin{aligned}
a_{kk}^{m+1} &- \sum_{\substack{l\in C\\ l\neq k}} |a_{kl}^{m+1}| \\
&\geq a_{kk}^m - \sum_{\substack{l\in C^m\\ l\neq k}} |a_{kl}^m| - \sum_{i\in F^m} |w_{ik}^m| \sum_{l\in C^m} |a_{il}^m| \\
&\quad - \sum_{i\in F^m} |a_{ik}^m| S_i^m + \sum_{i\in F^m} |w_{ik}^m| S_i^m a_{ii}^m \\
&\quad - |w_{ik}^m| \sum_{\substack{j\in F^m\\ j\neq i}} S_j^m |a_{ij}^m| \qquad (4.11)\\
&= a_{kk}^m - \sum_{l\neq k} |a_{kl}^m| + \sum_{i\in F^m} (1 - S_i^m)|a_{ik}^m| \\
&\quad - \sum_{i\in F^m} |w_{ik}^m| \left[\sum_{l\in C^m} |a_{il}^m| - S_i^m a_{ii}^m + \sum_{\substack{j\in F^m\\ j\neq i}} S_j^m |a_{ij}^m| \right] \\
&\geq a_{kk}^m - \sum_{l\neq k} |a_{kl}^m| + \sum_{i\in F^m} (1 - S_i^m)|a_{ik}^m| \\
&\quad - \sum_{i\in F^m} |w_{ik}^m| \left(\sum_{l\neq i} |a_{il}^m| - S_i^m a_{ii}^m \right),
\end{aligned}
$$

where $S_l^m \leq 1$ is used. Using the inequality (4.9), we get

$$
\begin{aligned}
(1 - S_i^m)|a_{ik}^m| &- |w_{ik}^m| \left(\sum_{l\neq i} |a_{il}^m| - S_i^m a_{ii}^m \right) \\
&= (1 - S_i^m)|a_{ik}^m| - |w_{ik}^m| \left[-a_{ii}^m \right. \\
&\quad \left. + \sum_{l\neq i} |a_{il}^m| + (1 - S_i^m) a_{ii}^m \right] \qquad (4.12)\\
&\geq (1 - S_i^m)|a_{ik}^m| - |w_{ik}^m|[-(1-\gamma^m) a_{ii}^m (1 - S_i^m) \\
&\quad + (1 - S_i^m) a_{ii}^m] \\
&= (1 - S_i^m)(|a_{ik}^m| - \gamma^m |w_{ik}^m| a_{ii}^m) \geq 0,
\end{aligned}
$$

where the inequality (4.10) is used. Hence,

$$
a_{kk}^{m+1} - \sum_{\substack{l\in C\\ l\neq k}} |a_{kl}^{m+1}| \geq a_{kk}^m - \sum_{\substack{l\in C\\ l\neq k}} |a_{kl}^m| \geq 0.
$$

Remark. In this section, a theoretical analysis of convergence is presented. It has been proven that for symmetric, positive definite, and weakly diagonally dominant matrices, a uniform convergence is achieved for a two-level AMG method. An important result presented in Theorem 7 is that the convergence factor of the AMG method used in conjunction with the new interpolation formulae (3.7)–(3.8) is shown to be less than one. In Theorem 8, it is proved that if A^m is symmetric, positive definite, and weakly diagonally dominant, then these properties are preserved in the coarse grid operator A^{m+1}. Consequently, the convergence analysis for a two-level method can be extended to a multi-level AMG method. Our result provides the bound on the convergence factor, but it does not guarantee that it is independent of the number of grid levels. However, in practical computations, only a finite number of grid levels is applied in an AMG method. For symmetric, positive definite and weakly diagonally dominant matrix problems, our numerical results given in the next section indicate that the convergence factor is indeed h-independent when conditions (4.8) and (4.10) are satisfied.

5. NUMERICAL RESULTS

A series of numerical experiments were tested on an INDIGO2 Silicon Graphics workstation to evaluate the performance of the new AMG methods (Methods II and III) proposed in this paper. Numerical results were compared with those obtained using the standard AMG algorithm (Method I) of Ruge and Stüben. Particular attentions are focused on the convergence factor and the range of applications.

The following notations are used for the results reported in all tables:

ρ: asymptotic convergence factor,

t_I: computing time in seconds for one V-cycle,

t_P: computing time for the setup phase,

N: number of iterations for convergence defined by $\|r^N\|/\|r^0\| \leq 10^{-6}$, where r^N is the residual vector at the Nth iteration,

EQ: total number of matrix equations,

σ^A: ratio of the space occupied by all operators to the space at the finest grid,

σ^Ω: ratio of the total number of points on all grids to that on the finest grid.

In all computations, the initial iteration u^0 is taken to be random numbers uniformly distributed in $[0, 1]$, and the Gauss–Seidel relaxation is used as the smoothing operator and $\theta_0 = 0.25$. Notice that, when $\theta_1 = 0$ in Method II, the coarse grid equation is essentially constructed by a Galerkin-type algorithm. Thus the main difference between Method I and Method II with $\theta_1 = 0$ is in the interpolation formula.

PROBLEM 1. Poisson problems on a unit square/cube with Dirichlet boundary conditions. For two-dimensional

TABLE II

Numerical Results for 2D–Poisson Problem with 5-Point Stencil

Method	θ_1	EQ	ρ	t_I	t_P	σ^A	σ^Ω
I		64×64	0.0339	0.07	0.42	2.21	1.68
		128×128	0.0334	0.34	1.93	2.21	1.68
II	0	64×64	0.0211	0.07	0.41	2.16	1.66
		128×128	0.0215	0.33	1.65	2.18	1.67
	0.01	64×64	0.0202	0.07	0.39	2.16	1.66
		128×128	0.0204	0.33	1.57	2.18	1.67
III	1/17	64×64	0.0597	0.07	0.38	2.20	1.68
		128×128	0.0611	0.33	1.66	2.20	1.68
	0.1	64×64	0.0576	0.06	0.38	2.20	1.68
		128×128	0.0582	0.32	1.65	2.20	1.68

TABLE III

Numerical Results for 2D–Poisson Problem with 9-Point Stencil

Method	θ_1	EQ	ρ	t_I	t_P	σ^A	σ^Ω
I		64×64	0.0904	0.07	0.46	1.36	1.36
		128×128	0.0933	0.29	2.07	1.35	1.35
II	0	64×64	0.0704	0.06	0.43	1.32	1.33
		128×128	0.0756	0.29	1.89	1.33	1.33
	0.01	64×64	0.0725	0.06	0.43	1.32	1.33
		128×128	0.0709	0.29	1.85	1.33	1.33
III	1/17	64×64	0.128	0.06	0.44	1.33	1.33
		128×128	0.129	0.29	2.01	1.33	1.33
	0.1	64×64	0.238	0.06	0.42	1.32	1.33
		128×128	0.327	0.29	1.88	1.33	1.33

Poisson problems, we consider the following 5-point and 9-point stencils

$$\frac{1}{h^2}\begin{bmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix}, \quad \frac{1}{20h^2}\begin{bmatrix} -1 & -4 & -1 \\ -4 & 20 & -4 \\ -1 & -4 & -1 \end{bmatrix}.$$

For three-dimensional problems, the 7-point difference approximation is applied.

$$\frac{1}{h^2}(6u_{i,j,k} - u_{i+1,j,k} - u_{i-1,j,k} - u_{i,j+1,k} - u_{i,j-1,k} - u_{i,j,k+1} - u_{i,j,k-1}) = f_{i,j,k}.$$

The computational results for these problems are given in Tables II, III, and IV.

It follows from Tables II and III that the 9-point stencils requires more time for relaxation on the finest grid than the 5-point stencils, the overall computing times per cycle, however, are actually somewhat reduced because σ^A and σ^Ω are smaller for the 9-point stencils. It is generally true that larger stencils result in a faster coarsening step.

PROBLEM 2. Anisotropic problems on a unit square with Dirichlet boundary conditions. The first problem is

$$-\varepsilon u_{xx} - u_{yy} = f,$$

where $\varepsilon = 0.01$ is used. This example demonstrates the ability of the AMG method to tailor the coarsening step for a given problem. Another problem with variable coefficient is given as

$$-(100^{x+y-1}u_x)_x - u_{yy} = f.$$

Here, the direction and the strength of an anisotropy problem varies over the domain. The two problems are discretized on a uniform grid using the 5-point stencils. The computation results are given in Tables V and VI.

In a geometric multigrid, a line relaxation must be used for the anisotropic problems in order to ensure sufficient smoothing when a standard coarsening is used. However, an AMG method with a fixed Gauss–Seidel relaxation can be used to solve these problems, since the coarsening in

TABLE IV

Numerical Results for 3D–Poisson Problem with 7-Point Scheme

Method	θ_1	EQ	ρ	t_I	t_P	σ^A	σ^Ω
I		$16 \times 16 \times 16$	0.0263	0.11	1.36	3.16	1.67
		$24 \times 24 \times 24$	0.0619	0.52	5.72	3.33	1.66
II	0	$16 \times 16 \times 16$	0.0160	0.11	1.01	2.63	1.60
		$24 \times 24 \times 24$	0.0186	0.44	3.50	2.72	1.60
	0.01	$16 \times 16 \times 16$	0.0502	0.11	0.99	2.62	1.60
		$24 \times 24 \times 24$	0.1310	0.42	3.31	2.71	1.60
	0.005	$16 \times 16 \times 16$	0.0201	0.10	0.95	2.62	1.60
		$24 \times 24 \times 24$	0.0493	0.42	3.31	2.71	1.60
III	1/37	$16 \times 16 \times 16$	0.0583	0.11	0.99	2.84	1.65
		$24 \times 24 \times 24$	0.0794	0.43	3.83	2.88	1.64
	0.05	$16 \times 16 \times 16$	0.0772	0.11	0.90	2.79	1.65
		$24 \times 24 \times 24$	0.1090	0.42	2.85	2.85	1.64

TABLE V

Numerical Results for Anisotropic Problem with Constant Coefficient

Method	θ_1	EQ	ρ	t_I	t_P	σ^A	σ^Ω
I		64 × 64	0.0043	0.13	0.83	4.28	2.03
		128 × 128	0.0076	0.67	4.14	4.65	2.04
II	0	64 × 64	0.0040	0.13	0.84	4.45	2.06
		128 × 128	0.0057	0.67	4.03	4.87	2.09
	0.01	64 × 64	0.0081	0.13	0.68	3.99	2.03
		128 × 128	0.0237	0.67	3.12	4.51	2.08
III	1/17	64 × 64	0.0188	0.09	0.44	2.32	1.99
		128 × 128	0.0187	0.42	1.78	2.74	2.04
	0.2	64 × 64	0.0162	0.09	0.36	2.53	2.02
		128 × 128	0.0181	0.40	1.78	2.63	2.03

each part of the domain is automatically adapted to the direction of stronger connections there.

For the anisotropic problems, the complexity parameters σ^A and σ^Ω of Method III are less than for the other methods. Hence the computing times t_I and t_P are faster, even though the convergence factor is slower.

PROBLEM 3. Nonsymmetric problems. Consider the convection–diffusion equation in a unit square with Dirichlet boundary conditions

$$-\varepsilon\Delta u + a(x, y)u_x + b(x, y)u_y = f(x, y).$$

TABLE VI

Numerical Results for Anisotropic Problem with Variable Coefficient

Method	θ_1	EQ	ρ	t_I	t_P	σ^A	σ^Ω
I		64 × 64	0.1330	0.11	0.65	3.54	1.98
		128 × 128			[a]		
II	0	64 × 64	0.1662	0.14	0.80	4.32	2.12
		128 × 128	0.2031	0.59	3.51	4.57	2.11
	0.01	64 × 64	0.5983	0.13	0.70	4.04	2.11
		128 × 128			[b]		
III	0.01	64 × 64	0.6814	0.11	0.66	3.41	1.86
		128 × 128	0.8637	0.57	3.35	4.30	1.95
	0.1	64 × 64	0.7910	0.08	0.34	2.58	1.85
		128 × 128	0.8792	0.43	1.64	3.02	1.94

[a] Method is divergent.
[b] Error cannot be reduced.

TABLE VII

Computation Results for Nonsymmetric Problem

Method	θ_1	EQ	ρ	t_I	t_P	σ^A	σ^Ω
I		64 × 64	0.0083	0.13	0.66	3.79	2.02
		128 × 128	0.0289	0.60	3.78	4.04	2.02
II	0	64 × 64	0.0092	0.12	0.64	3.70	2.00
		128 × 128	0.0144	0.57	3.06	3.96	2.01
	0.01	64 × 64	0.0117	0.12	0.59	3.50	2.00
		128 × 128	0.0167	0.54	2.60	3.71	2.01
III	1/17	64 × 64	0.0753	0.08	0.26	2.58	1.99
		128 × 128	0.1341	0.41	1.36	2.70	2.00
	0.01	64 × 64	0.0753	0.08	0.27	2.61	1.99
		128 × 128	0.1341	0.41	1.44	2.75	2.00

The discrete operator is based on the 5-point finite difference approximation of the form

$$\frac{1}{h^2}\begin{bmatrix} & -\varepsilon + bh\mu_y & \\ -\varepsilon + ah(\mu_x - 1) & -\Sigma & -\varepsilon + ah\mu_x \\ & -\varepsilon + bh(\mu_y - 1) & \end{bmatrix},$$

where

$$\mu_x = \begin{cases} \dfrac{\varepsilon}{2ah} & \text{if } ah > \varepsilon \\ 1 + \dfrac{\varepsilon}{2ah} & \text{if } ah < -\varepsilon; \\ \dfrac{1}{2} & \text{if } |ah| \le \varepsilon \end{cases} \quad \mu_y = \begin{cases} \dfrac{\varepsilon}{2bh} & \text{if } bh > \varepsilon \\ 1 + \dfrac{\varepsilon}{2bh} & \text{if } bh < -\varepsilon \\ \dfrac{1}{2} & \text{if } |bh| \le \varepsilon. \end{cases}$$

Σ denotes the sum of the surrounding coefficients. In computation, the functions a and b are taken as $a(x, y) = \sin(\pi/8)$, $b(x, y) = \cos(\pi/8)$, and $\varepsilon = 10^{-5}$. The numerical reults are given in Table VII.

PROBLEM 4. Biharmonic equation on a unit square. Let

$$\Delta^2 u = 0 \quad \text{on } \Omega,$$

$$u|_{\partial\Omega} = 0, \quad \left.\frac{\partial u}{\partial n}\right|_{\partial\Omega} = 0,$$

with the following 13-point finite difference stencil:

TABLE VIII

Computation Results for Biharmonic Problem

Method	θ_1	EQ	ρ	N	t_I	t_P	σ^A	σ^Ω
I		32 × 32	0.889	118	0.03	0.42	2.25	1.72
		64 × 64	0.866	96	0.15	1.82	2.27	1.70
		128 × 128	0.856	84	0.94	7.19	2.28	1.69
II	0	32 × 32	0.457	18	0.04	0.33	2.83	1.86
		64 × 64	0.464	18	0.24	1.57	2.96	1.87
		128 × 128	0.493	19	1.03	6.85	3.03	1.89
	0.2	32 × 32	0.894	123	0.02	0.12	1.46	1.69
		64 × 64	0.905	138	0.11	0.60	1.47	1.69
		128 × 128	0.896	125	0.51	2.47	1.47	1.69
III	1/17	32 × 32	0.463	18	0.04	0.61	3.72	1.88
		64 × 64	0.687	37	0.29	2.81	3.93	1.88
		128 × 128	0.793	57	1.21	11.82	4.04	1.90
	1/33	32 × 32	0.274	11	0.08	0.85	4.36	1.91
		64 × 64	0.674	35	0.36	4.08	4.65	1.89
		128 × 128	0.806	68	1.41	17.66	4.85	1.92

$$\begin{bmatrix} & & 1 & & \\ & 2 & -8 & 2 & \\ 1 & -8 & 20 & -8 & 1 \\ & 2 & -8 & 2 & \\ & & 1 & & \end{bmatrix}.$$

The resulting matrix equation is symmetric but not diagonally dominant. Furthermore, the linear system is very ill-conditioned with a condition number $O(h^{-4})$, whereas the condition numbers for the previous problems are $O(h^{-2})$. Table VIII compares the performance of various AMG methods.

From the results presented in Table VIII, we observe that Method I gives a slow convergence factor. However, a much faster convergence factor is achieved and this leads to a significant saving in the number of iterations for Method II, using $\theta_1 = 0$. Recall that the main difference between Method I and Method II with $\theta_1 = 0$ is in the interpolation formula. Hence, the results clearly demonstrate the effectiveness of the new interpolation proposed in this paper. Method II with $\theta_1 = 0.2$ reduces the complexity parameters σ^A and σ^Ω, and, hence, it provides improvement in the setup phase but the convergence factor is increased.

Problem 5. Toeplitz matrix. Signal processing problems arise in many applications, e.g., image restoration, seismic tomography, noise reduction, system identifications, neural networking, and data compression. In these applications, the problem can be transformed into problems of solving a linear system $Ax = b$, where the n by n coefficient matrix A is either a Toeplitz matrix or a near-Toeplitz matrix. We now consider a simple Toeplitz problem

$$A = (a_{ij}), \quad a_{ij} = 2^{-(i-j)},$$

where A is a symmetric full matrix with positive entries. In this problem, we only consider three maximal off-diagonal entries for each row, when the interpolation operators are constructed. The numerical results are given in Table IX. Method I cannot be used to solve this problem, since all entries of the matrix are positive. In [9], we modify Method II, and a faster convergence for more complex Toeplitz matrices was obtained.

Problem 6. Queueing network problems (singular problems). Queueing networks are often analyzed to determine the behavior under different traffic situations. The analysis indicates the effect of increasing servers on the waiting times of customers and on the blocking of customers. In this problem, we have to solve a linear homogeneous system. Now, consider an overflow queueing model [12] and the linear system is given by

$$Au = (A_0 + R_0)u = 0,$$
$$\sum_i u_i = 1, \quad u_i \geq 0,$$

where $A_0 = G \otimes I_n + I_n \otimes G$, $R_0 = (e_n e_n^*) \otimes R_1$, and

TABLE IX

Computation Results for Toeplitz Matrix

Method	θ_1	EQ	ρ	t_I	t_P	σ^A	σ^Ω
I		Unable to compute					
II	0	256	0.1320	0.07	0.22	1.15	1.62
		512	0.1330	0.29	0.94	1.15	1.62
	0.2	256	0.1290	0.07	0.19	1.15	1.62
		512	0.1310	0.29	0.86	1.15	1.62
III	1/17	256	0.3850	0.06	0.20	1.14	1.58
		512	0.3871	0.27	0.84	1.14	1.58
	1/33	256	0.3742	0.06	0.27	1.17	1.71
		512	0.3757	0.29	1.13	1.17	1.71

TABLE X

Computation Results for Queueing Network Problem

Method	θ_1	EQ	ρ	t_I	t_P	σ^A	σ^Ω
I		64×64	0.291	0.08	0.42	2.23	1.69
		128×128	[a]				
II	0	64×64	0.127	0.10	0.48	2.40	1.72
		128×128	0.214	0.38	1.94	2.49	1.73
	0.01	64×64	0.164	0.08	0.46	2.34	1.72
		128×128	0.246	0.36	1.80	2.44	1.72
III	1/17	64×64	0.500	0.09	0.51	3.35	1.88
		128×128	0.707	0.44	2.28	3.10	1.81
	1/33	64×64	0.481	0.12	0.63	3.72	1.88
		128×128	0.701	0.47	2.67	3.40	1.82

[a] Error cannot be reduced.

$$G = \begin{bmatrix} \lambda & -\mu & & & & & \\ -\lambda & \lambda+\mu & -2\mu & & & & \\ & -\lambda & \lambda+2\mu & -3\mu & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & -\lambda & \lambda+s\mu & -s\mu & \\ & & & \ddots & \ddots & \ddots & \\ & & & & -\lambda & \lambda+s\mu & -s\mu \\ & & & & & -\lambda & -s\mu \end{bmatrix}_{n\times n},$$

$$R_1 = \lambda \cdot \begin{bmatrix} 1 & & & & \\ -1 & 1 & & 0 & \\ & \ddots & \ddots & & \\ & 0 & & -1 & 1 \\ & & & & -1 & 0 \end{bmatrix}_{n\times n}.$$

Here, I_n is an identity matrix of order n, e_n denotes the jth unit vector in R^n, and $\otimes$ denotes the Kronecker product. The matrix A is of order n^2 by n^2 which is irreducible with zero column sums, strictly positive diagonal, and non-positive off-diagonal entries and it has a one-dimensional null-space. The computational results for the problem with parameters $s = 5$, $\mu = 1$, $\lambda = s\mu - \frac{1}{2}(n-1)^{-1}$ are given in Table X.

This problem demonstrates that the new AMG algorithms can directly apply to solve singular systems of equations. In [8], we modify Method II for other queueing network problems and much faster convergence rates are obtained.

Problem 7. Elasticity problems. Now, consider a system of partial differential equations. The plane-stress elasticity problem can be written in terms of the displacements u and v as

$$u_{xx} + \frac{1-\nu}{2}u_{yy} + \frac{1+\nu}{2}v_{xy} = f,$$

$$\frac{1+\nu}{2}u_{xy} + \frac{1-\nu}{2}v_{xx} + v_{yy} = g, \quad (x, y) \in (0,1) \times (0,1),$$

where ν is Possion's ratio, chosen to be $\frac{1}{3}$. The Dirichlet boundary conditions are used, and the problem is discretised by a central difference:

$$\frac{1}{h^2}(u_{i+1,j} - 2u_{i,j} + u_{i-1,j}) + \frac{1-\nu}{2h^2}(u_{i,j+1} - 2u_{i,j} + u_{i,j-1})$$
$$+\frac{1+\nu}{h^2}(v_{i+1,j+1} - v_{i-1,j+1} - v_{i+1,j-1} + v_{i-1,j-1}) + = f_{i,j};$$
$$\frac{1-\nu}{2h^2}(v_{i+1,j} - 2v_{i,j} + v_{i-1,j}) + \frac{1}{h^2}(v_{i,j+1} - 2v_{i,j} + v_{i,j-1})$$
$$+\frac{1+\nu}{h^2}(u_{i+1,j+1} - u_{i-1,j+1} - u_{i+1,j-1} + u_{i-1,j-1}) = g_{i,j}.$$

The resulting system is solved directly by AMG methods and the numerical results are reported in Table XI.

In this problem, the complexity parameters σ^A and σ^Ω are larger compared to other problems since this is a difficult system. The convergence factors given by the Methods II and III indicate that both methods are working very well.

Remark. From the results presented in this section, we observe the following:

TABLE XI

Computation Results for Elastic Problem

Method	θ_1	EQ	ρ	t_I	t_P	σ^A	σ^Ω
I		$32 \times 32 \times 2$	0.8407	0.12	1.51	5.41	2.02
		$48 \times 48 \times 2$	[a]				
II	0	$32 \times 32 \times 2$	0.1260	0.23	1.64	8.04	2.29
		$48 \times 48 \times 2$	0.2070	1.11	4.09	8.23	2.25
	0.01	$32 \times 32 \times 2$	0.3450	0.13	0.96	5.65	2.30
		$48 \times 48 \times 2$	0.5591	0.34	2.48	5.93	2.30
III	1/17	$32 \times 32 \times 2$	0.1700	0.21	1.92	9.40	2.49
		$48 \times 48 \times 2$	0.2503	0.75	4.48	10.53	2.53
	1/24	$32 \times 32 \times 2$	0.0960	0.24	2.30	10.10	2.49
		$48 \times 48 \times 2$	0.1350	0.77	5.74	11.61	2.56

[a] Method is divergent.

(1) When both Methods I and II are convergent, Method II with $\theta_1 = 0$ gives a faster convergence for all problems tested here. For an ill-conditioned system such as the biharmonic equation (see Table VIII), a significant reduction in the convergence factor and the number of iterations is achieved by using Method II with $\theta_1 = 0$. Notice that the improvement over Method I results are entirely due to a more accurate interpolation formula (3.7)–(3.8), proposed in this paper.

(2) By varying the parameter θ_1 in Method II, we reduce the complexity parameters σ^A and σ^Ω, so that less computing time is required in the setup phase and so is the time needed to perform one V-cycle multigrid. However, this usually leads to a larger convergence factor and, thus, the number of iterations is increased. However, the overall computing time may be reduced in some applications due to the improvement in the complexity parameters.

(3) Method III is robust and can be used to solve all the problems presented here. However, it requires larger complexity parameters and convergence factor.

(4) The new AMG methods II and III are superior compared to the standard method I. It has been clearly demonstrated by the fact that when Method I failed to converge for difficult problems (such as Problems 5–7), Methods II and III converge with good convergence factors.

6. CONCLUDING REMARKS

In this paper, a new formulation for the interpolation formula (3.7)–(3.8) is presented. The proposed formula is compared to the standard formula (3.1)–(3.2) suggested by Ruge and Stüben. The standard formula is of order one and suitable for M-matrix problems. The new formula is more accurate; it is nearly of order two and can be used for more general matrix problems. In choosing the C-points required in the setup phase of an AMG method, the computational results demonstrate that fewer additional C-points are introduced by the interpolation (3.7)–(3.8) compared to those required by using (3.1)–(3.2). Consequently, even though the new interpolation operator is more complicated, the complexity parameters σ^A, σ^Ω and the computing times in using (3.7)–(3.8), however, can be less than those using (3.1)–(3.2). This could lead to a more efficient implementation for an AMG method.

In the standard AMG method, a positive weighting is always used in the interpolation operator. The proposed new interpolation formula includes the choice of negative weights. Indeed, theoretical analysis and our computational experience show that using negative interpolation weights is necessary in some applications, for example, the solutions of Toeplitz matrix and systems of partial differential equations in elasticity problems.

New algorithms are presented for the construction of the restriction operator and the coarse-grid equations. The proposed new AMG methods II and III are compared to the standard AMG method I developed by Ruge and Stüben. In Method II, the parameter θ_1 can be chosen as a small quantity, for example, 0.01, 0.005, 0.1, or 0.2, in order to reduce the complexity parameters and the computing time. In Method III, the parameter θ_1 is generally taken to be 1/17 for 2D problems and 1/37 for 3D problems, and it can also be chosen as other values in order to reduce the computing time or to accelerate the convergence.

Numerous computational experiments are reported including matrix solutions that resulted from partial differential equations, signal processing, and queueing network problems. For systems of partial differential equations, Ruge and Stüben presented the unknown approach and the point approach methods [3]. Here, the new AMG methods II and III are applied directly to the systems of equations such as the elasticity problem. Method III has also been successfully tested for problems in computational fluid dynamics, including the solutions of the system of Euler equations [7]. From our analysis and numerical results, the following conclusions are made. The standard AMG method I is only efficient for the M-matrix problems. The proposed new AMG methods II and III not only can provide a fast convergence rate, but they can also be applied to more general applications, including non-diagonally dominant matrices.

ACKNOWLEDGMENTS

The authors express their appreciation to J. Ruge and K. Stüben for providing the AMG code of Method I. The authors also thank the referees for their comments and suggestions. This research was supported in part by the Natural Science and Engineering Research Council of Canada.

REFERENCES

1. K. Stüben and U. Trottenberg, "Multigrid Methods: Fundamental Algorithms, Model Problem Analysis and Application," Lecture Note in Mathematics, Vol. **962** (Springer-Verlag, New York/Berlin, 1982).
2. A. Brandt, S. F. McCormick, and J. Ruge, Institute for Computational Studies Technical Report, Colorado State University, Fort Collins, CO, 1982 (unpublished).
3. J. Ruge and K. Stüben, "Algebraic Multigrid," in *Multigrid Methods,* Vol. 4 (S. F. McCormick, Ed.) (SIAM, Philadelphia, 1987).
4. Q. Chang, Y. S. Wong, and Z. Li, *Appl. Math. Comput.* **50,** 223 (1992).
5. Q. Chang and Y. S. Wong, "Recent Developments in Algebraic Multigrids," in *Proceedings, Copper Mountain Conference on Iterative Methods, Vol. 1, Colorado, April 9–14, 1992.*
6. Q. Chang and Y. S. Wong, *Int. J. Comput. Math.* **49,** 197 (1993).
7. Q. Chang, Y. S. Wong, and H. Fu, "Algebraic Multigrid and Its

Application to the Euler Equations,'' in *Proceedings, Second International Conference on Computational Physics* (World Scientific, Singapore, 1993).

8. R. H. Chan, Q. Chang, and W. K. Ching, to appear.
9. R. H. Chan, Q. Chang, and H. W. Sun, to appear.
10. J. Ruge, *Appl. Math. Comput.* **19,** 293 (1986).
11. W. Z. Huang, *Appl. Math. Comput.* **46,** 145 (1991).
12. R. H. Chan, *Numer. Math.* **51,** 143 (1987).