

# New Convergence Estimates for Multigrid Algorithms

By James H. Bramble and Joseph E. Pasciak\*

**Abstract.** In this paper, new convergence estimates are proved for both symmetric and nonsymmetric multigrid algorithms applied to symmetric positive definite problems. Our theory relates the convergence of multigrid algorithms to a “regularity and approximation” parameter  $\alpha \in (0, 1]$  and the number of relaxations  $m$ . We show that for the symmetric and nonsymmetric  $\mathcal{V}$  cycles, the multigrid iteration converges for any positive  $m$  at a rate which deteriorates no worse than  $1 - cj^{-(1-\alpha)/\alpha}$ , where  $j$  is the number of grid levels. We then define a generalized  $\mathcal{V}$  cycle algorithm which involves exponentially increasing (for example, doubling) the number of smoothings on successively coarser grids. We show that the resulting symmetric and nonsymmetric multigrid iterations converge for any  $\alpha$  with rates that are independent of the mesh size. The theory is presented in an abstract setting which can be applied to finite element multigrid and finite difference multigrid methods.

**1. Introduction.** In recent years, multigrid methods have been used extensively as tools for obtaining approximations to solutions of partial differential equations (see the references in [5], [9]). In conjunction, there has been intensive research into the theoretical understanding of the convergence properties of these methods (cf. [2], [3], [4], [9], [11], [12]–[16], [18]). This paper will present a number of new results on the convergence of multigrid algorithms.

We shall be concerned with the analysis of many-level multigrid schemes. The first approach to this problem involved Fourier analysis and only applied to rather limited situations, i.e., rectangular domains [8]. More general results can be obtained by variational or finite element like formulations of multigrid. One approach is to obtain results for two-grid schemes and use those results to derive estimates for the many-level schemes [3], [9]. The problem with this technique is that it only leads to results for  $\mathcal{W}$  cycles with sufficiently large  $m$  (the number of smoothing iterations). Another interesting approach was taken in [4] in which a direct analysis was made for the many-level scheme. This technique leads to results for the  $\mathcal{V}$  cycle with any  $m$  but under the assumption of ‘full elliptic regularity’, i.e.,  $\alpha = 1$ . Results for the  $\mathcal{W}$  cycle algorithm for any  $m$  and  $\alpha$  were given in [13], [14]. The theory presented in this paper also provides a direct analysis of many-level schemes.

---

Received January 5, 1987.

1980 *Mathematics Subject Classification* (1985 Revision). Primary 65N30; Secondary 65F10.

\*This manuscript has been authored under contract number DE-AC02-76CH00016 with the U.S. Department of Energy. Accordingly, the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes. This work was also supported in part under the National Science Foundation Grant No. DMS84-05352 and under the Air Force Office of Scientific Research, Contract No. ISSA86-0026 and by the U.S. Army Research Office through the Mathematical Science Institute, Cornell University.

©1987 American Mathematical Society  
0025-5718/87 \$1.00 + \$.25 per page

We shall provide an analysis for symmetric and nonsymmetric multigrid cycling algorithms applied to symmetric positive definite problems. We relate our convergence estimates to a certain discrete regularity and approximation assumption which involves the parameter  $\alpha \in (0, 1]$ . We will show that both the symmetric and nonsymmetric  $\mathcal{V}$  cycle algorithms converge for any  $m$  and any  $\alpha$  at a rate

$$(1.1) \quad \delta_j \leq \frac{M_\alpha(j_0 + j)^{(1-\alpha)/\alpha}}{m^\alpha + M_\alpha(j_0 + j)^{(1-\alpha)/\alpha}}$$

per iteration. Here  $M_\alpha$  and  $j_0$  are constants which are independent of  $j$ , the number of grid levels. In finite element and finite difference applications,  $j$  is proportional to  $\log(1/h)$  where  $h$  is essentially the size of the finest grid. In such applications, our theorems guarantee that the convergence rate can only deteriorate like  $1 - c/\log^{(1-\alpha)/\alpha}(1/h)$  as  $h \rightarrow 0$ . The estimates improve as  $m$  becomes large.

We next consider the symmetric and nonsymmetric  $\mathcal{W}$  cycle algorithms. Although the uniform convergence (independent of the number of levels) of this algorithm for  $\alpha = 1$  and any  $m$  was shown in [13], [14], we derive convergence estimates in the context of our framework. We prove a convergence bound of the form

$$(1.2) \quad \delta \leq (1 + m/M_\alpha)^{-\alpha}.$$

Again, we have convergence for any  $m$  and any  $\alpha$ , but for the  $\mathcal{W}$  cycle the rate is also bounded independently of the number of levels  $j$ .

Finally, we introduce a generalized  $\mathcal{V}$  cycle algorithm and derive the corresponding convergence estimates. In this algorithm, the number of smoothings varies on different grid levels. One example doubles the number of smoothings on each consecutive coarser grid level. Note that this doubling strategy gives rise to the same number of smoothings as the  $\mathcal{W}$  cycle but is simpler to code. We show that the generalized  $\mathcal{V}$  cycle also converges for any  $\alpha$  with rates that are independent of  $j$  (see Theorems 5 and 6).

In our algorithms and analysis, the smoothing process is defined in terms of a general smoothing operator. Consequently, our analysis is applicable to many of the smoothing processes used in actual multigrid applications. We should however note that according to our analysis, little appears to be gained by using smoothing operators which are more complex than a weighted identity, although in practice some improvement may result.

The outline of the remainder of the paper is as follows. In Section 2 we define the multigrid algorithms. These algorithms are described by a simple induction process and lead to linear operators which ‘approximate’ the inverse of the problem to which they are applied. In Section 3 we give the multigrid convergence analysis. In Section 4 we show how the theorems of Section 3 can be used to guarantee rapid convergence of preconditioned iterative schemes using the multigrid operator as a preconditioner. Finite element and finite difference applications are considered in Section 5. In addition, we show that for finite differences, symmetric Gauss-Seidel iteration leads to a smoothing operator which satisfies all of the hypotheses of the theorem. Finally, in Section 6, we give the results of numerically computed convergence factors for multigrid algorithms. We also give numerical evidence which suggests that the ‘regularity and approximation’ assumption does not in general hold for all  $\alpha$ . Hence, the new convergence estimates presented in this paper for  $\alpha < 1$  are of theoretical importance.

**2. The Multigrid Algorithms.** In this section we describe both the symmetric and nonsymmetric multigrid cycling algorithms. Along the way, we derive

basic recursion relations which play major roles in the analysis of the methods. For convenience, the algorithms are developed in an abstract Hilbert space setting. The results most naturally apply to finite element multigrid algorithms but can also be applied to certain formulations of finite difference multigrid algorithms.

Let us assume that we are given a nested sequence of finite-dimensional vector spaces

$$\mathcal{M}_0 \subset \mathcal{M}_1 \subset \cdots \subset \mathcal{M}_j.$$

In addition, let  $A(\cdot, \cdot)$  and  $(\cdot, \cdot)_k$  be symmetric positive definite bilinear forms on  $\mathcal{M}_k$  for  $k = 0, \dots, j$ . We shall develop multigrid algorithms for the solution of the problem: Given  $f \in \mathcal{M}_j$ , find  $v \in \mathcal{M}_j$  satisfying

$$(2.1) \quad A(v, \phi) = (f, \phi)_j \quad \text{for all } \phi \in \mathcal{M}_j.$$

To define the multigrid algorithms, we shall define auxiliary operators. For  $k = 0, \dots, j$ , define the operator  $A_k: \mathcal{M}_k \mapsto \mathcal{M}_k$  by

$$(A_k w, \phi)_k = A(w, \phi) \quad \text{for all } \phi \in \mathcal{M}_k.$$

The operator  $A_k$  is clearly symmetric (in both the  $A(\cdot, \cdot)$  and  $(\cdot, \cdot)_k$ -inner products) and positive definite. Also define the projectors  $P_k: \mathcal{M}_{k+1} \mapsto \mathcal{M}_k$  and  $P_k^0: \mathcal{M}_{k+1} \mapsto \mathcal{M}_k$  by

$$A(P_k w, \phi) = A(w, \phi) \quad \text{for all } \phi \in \mathcal{M}_k,$$

and

$$(P_k^0 w, \phi)_k = (w, \phi)_{k+1} \quad \text{for all } \phi \in \mathcal{M}_k.$$

Note that  $P_k$  is symmetric in the  $A$ -inner product.

To define the smoothing process, we require a linear operator  $R_k: \mathcal{M}_k \mapsto \mathcal{M}_k$  for  $k = 1, \dots, j$ . We assume that  $R_k$  is symmetric in the  $(\cdot, \cdot)_k$ -inner product and set  $K_k = (I - R_k A_k)$ . We further assume that  $K_k$  is nonnegative in the sense that  $A(K_k u, u) \geq 0$  for all  $u \in \mathcal{M}_k$ .

We first define the symmetric multigrid operator  $B_k^s: \mathcal{M}_k \mapsto \mathcal{M}_k$  by induction.

**Algorithm S.**

Set  $B_0^s = A_0^{-1}$ . Assume that  $B_{k-1}^s$  has been defined and define  $B_k^s g$  for  $g \in \mathcal{M}_k$  as follows:

- (1) Set  $x^0 = 0$  and  $q^0 = 0$ .
- (2) Define  $x^l$  for  $l = 1, \dots, m(k)$  by

$$(2.2) \quad x^l = x^{l-1} + R_k(g - A_k x^{l-1}).$$

- (3) Define  $x^{m(k)+1} = x^{m(k)} + q^p$  where  $q^i$  for  $i = 1, \dots, p$  is defined by

$$q^i = q^{i-1} + B_{k-1}^s [P_{k-1}^0 (g - A_k x^{m(k)}) - A_{k-1} q^{i-1}].$$

- (4) Set  $B_k^s g = x^{2m(k)+1}$  where  $x^l$  is defined for  $l = m(k) + 2, \dots, 2m(k) + 1$  by (2.2).

In this algorithm,  $m(k)$  is a positive integer which may vary from level to level and determines the number of smoothing iterations on that level. Because of this variable smoothing, the above algorithm is more general than that usually described [2], [3], [5], [9]. If all of the  $m(k)$  are the same, then this algorithm is the usual symmetric multigrid algorithm described in a notation which is convenient for our analysis. Note that  $B_k^s$  is clearly a linear operator for each  $k$ . In this algorithm,  $p$  is a positive integer. We shall study the cases  $p = 1$  and  $p = 2$  which correspond

respectively to the symmetric  $\mathcal{V}$  and  $\mathcal{W}$  cycles of multigrid. Generalizations to  $p > 2$  are straightforward and will not be considered.

The definition of the nonsymmetric multigrid operator  $B_k^n$  is similar except that the smoothings of Step 4 are excluded. More precisely, we define  $B_k^n: \mathcal{M}_k \mapsto \mathcal{M}_k$  by induction.

**Algorithm N.**

Set  $B_0^n = A_0^{-1}$ . Assume that  $B_{k-1}^n$  has been defined and define  $B_k^n g$  for  $g \in \mathcal{M}_k$  as follows:

- (1) Set  $x^0 = 0$  and  $q^0 = 0$ .
- (2) Define  $x^l$  for  $l = 1, \dots, m(k)$  by (2.2).
- (3) Define  $B_k^n g = x^{m(k)} + q^p$  where  $q^i$  for  $i = 1, \dots, p$  is defined by

$$(2.3) \quad q^i = q^{i-1} + B_{k-1}^n [P_{k-1}^0 (g - A_k x^{m(k)}) - A_{k-1} q^{i-1}].$$

The above algorithm defines a linear operator  $B_k^n$  which is equivalent to the standard nonsymmetric multigrid algorithms described in [3], [9] when  $m(k)$  is constant.

*Remark 2.1.* One computationally effective algorithm with variable  $m$  is the  $\mathcal{V}$  cycle algorithm ( $p = 1$ ) with  $m(k) = m_0 2^{j-k}$ . Note that, for this algorithm, the total number of smoothing iterations on each level is the same as that for the  $\mathcal{W}$  cycle ( $p = 2$ ) with  $m(k) = m_0$  for all  $k$ . We shall prove in Section 3 (see Theorems 5 and 6) that, like the corresponding  $\mathcal{W}$  cycle, this  $\mathcal{V}$  cycle converges for any  $m_0$  and  $\alpha$  with rates that are independent of  $j$ . In addition, this generalized  $\mathcal{V}$  cycle is easier to implement.

Let  $g = A_k x$ . It is straightforward to check that  $q^p$  defined by (2.3) satisfies

$$(2.4) \quad q^p = (I - (I - B_{k-1}^n A_{k-1})^p) A_{k-1}^{-1} P_{k-1}^0 A_k (x - x^{m(k)}).$$

A trivial computation gives that

$$(2.5) \quad x - x^{m(k)} = K_k^{m(k)} x.$$

Noting that  $P_{k-1}^0 A_k = A_{k-1} P_{k-1}$  and combining (2.4) and (2.5) gives

$$(2.6) \quad I - B_k^n A_k = [(I - P_{k-1}) + (I - B_{k-1}^n A_{k-1})^p P_{k-1}] K_k^{m(k)}.$$

Equation (2.6) gives a fundamental recurrence relation for the nonsymmetric multigrid algorithms. The analogous recurrence in the symmetric case is

$$(2.7) \quad I - B_k^s A_k = K_k^{m(k)} [(I - P_{k-1}) + (I - B_{k-1}^s A_{k-1})^p P_{k-1}] K_k^{m(k)},$$

which follows from similar reasonings.

Note that (2.7) implies that

$$(2.8) \quad \begin{aligned} A((I - B_k^s A_k)u, v) &= A((I - P_{k-1})K_k^{m(k)}u, K_k^{m(k)}v) \\ &\quad + A((I - B_{k-1}^s A_{k-1})^p P_{k-1} K_k^{m(k)}u, K_k^{m(k)}v). \end{aligned}$$

*Remark 2.2.* An obvious argument using (2.8) and induction gives that  $I - B_k^s A_k$  is a symmetric operator in the  $A$ -inner product. Consequently,  $B_k^s$  is symmetric in  $(\cdot, \cdot)_k$ .

*Remark 2.3.* A similar argument gives that

$$(2.9) \quad A((I - B_k^s A_k)u, u) \geq 0 \quad \text{for all } u \in \mathcal{M}_k.$$

We finally note that by the definition of  $P_k$ ,

$$\begin{aligned}
 & A((I - B_k^n A_k)u, (I - B_k^n A_k)u) \\
 (2.10) \quad & = A((I - P_{k-1})K_k^{m(k)}u, K_k^{m(k)}u) \\
 & \quad + A((I - B_{k-1}^n A_k)^p P_{k-1} K_k^{m(k)}u, (I - B_{k-1}^n A_k)^p P_{k-1} K_k^{m(k)}u).
 \end{aligned}$$

**3. Multigrid Analysis.** In this section we give an analysis of the multigrid algorithms described in the previous section. The goal of this section is to prove norm inequalities of the form

$$(3.1) \quad A((I - B_k^s A_k)u, u) \leq \delta_k A(u, u) \quad \text{for all } u \in \mathcal{M}_k,$$

and

$$(3.2) \quad A((I - B_k^n A_k)u, (I - B_k^n A_k)u) \leq \delta_k A(u, u) \quad \text{for all } u \in \mathcal{M}_k.$$

We shall relate  $\delta_k$  to two *a priori* assumptions. Let  $0 < \alpha \leq 1$ . The first assumption is a “regularity and approximation” assumption of the form

$$(3.3) \quad A((I - P_{k-1})u, u) \leq C_\alpha^2 \left( \frac{\|A_k u\|_k^2}{\lambda_k} \right)^\alpha A(u, u)^{1-\alpha} \quad \text{for all } u \in \mathcal{M}_k,$$

where  $\lambda_k$  is the largest eigenvalue of  $A_k$ . More precisely, we assume that (3.3) holds with  $C_\alpha$  independent of  $k$  for  $k = 1, \dots, j$ . As will be demonstrated in Section 5, in finite element applications, the derivation of inequalities of the form (3.3) uses regularity estimates for the elliptic operator being approximated and the approximation properties of the subspace. The second assumption is that the smoothing operator  $R_k$  satisfies the inequality

$$(3.4) \quad \frac{\|u\|_k^2}{\lambda_k} \leq C_R (R_k u, u)_k \quad \text{for all } u \in \mathcal{M}_k.$$

Again, we assume that (3.4) holds with a constant  $C_R$  independent of  $k$ . By an obvious change of variable, we see that (3.4) is equivalent to

$$(3.5) \quad \frac{\|A_k u\|_k^2}{\lambda_k} \leq C_R A((I - K_k)u, u) \quad \text{for all } u \in \mathcal{M}_k.$$

*Remark 3.1.* Note that a simple choice of the smoothing operator is  $R_k = \bar{\lambda}_k^{-1} I_k$  where  $\bar{\lambda}_k$  is any upper bound for  $\lambda_k$  (we need  $K_k$  nonnegative) and  $I_k$  is the identity on  $\mathcal{M}_k$ . Furthermore, if one takes  $\bar{\lambda}_k = \lambda_k$  then (3.4) holds with  $C_R = 1$ .

We can now state and prove the theorem for estimating  $\delta_k$  in (3.1) for the symmetric  $\mathcal{V}$  cycle.

**THEOREM 1.** Assume that (3.3) and (3.4) hold and define  $B_j^s$  by Algorithm S with  $p = 1$  and  $m(k) = m$  for all  $k$ . Then (3.1) holds with

$$(3.6) \quad \delta_j = 1 - \varepsilon_j,$$

where

$$(3.7) \quad \varepsilon_j = \frac{m^\alpha}{m^\alpha + M_\alpha(j_0 + j)^{(1-\alpha)/\alpha}}.$$

In (3.7),  $j_0$  is any positive constant and  $M_\alpha$  is defined by first defining

$$(3.8) \quad \tilde{M}_\alpha = \left( \frac{1 + j_0}{j_0} \right)^s \frac{C_R (\alpha C_\alpha^2)^{1/\alpha}}{2},$$

where

$$(3.9) \quad s = \begin{cases} \frac{1-\alpha}{\alpha} & \text{for } \alpha \geq 1/2, \\ \left(\frac{1-\alpha}{\alpha}\right)^2 & \text{for } \alpha < 1/2, \end{cases}$$

and then setting

$$(3.10) \quad M_\alpha = \left(\frac{1 + \tilde{M}_\alpha}{\tilde{M}_\alpha}\right)^{(1-\alpha)/\alpha} \tilde{M}_\alpha.$$

*Proof.* We shall prove that

$$(3.11) \quad A((I - B_i^s A_i)u, u) \leq \delta_i A(u, u) \quad \text{for all } u \in \mathcal{M}_i,$$

by induction on  $i$ . For  $i = 0$ , there is nothing to prove. Assume that (3.11) holds for  $i = k - 1$ . By (2.8) and the induction hypothesis,

$$(3.12) \quad \begin{aligned} A((I - B_k^s A_k)u, u) &\leq A((I - P_{k-1})K_k^m u, K_k^m u) \\ &\quad + \delta_{k-1} A(P_{k-1}K_k^m u, K_k^m u) \end{aligned}$$

$$(3.13) \quad = (1 - \delta_{k-1})A((I - P_{k-1})K_k^m u, K_k^m u) + \delta_{k-1}A(K_k^m u, K_k^m u).$$

By (3.3) and a generalized arithmetic-geometric mean inequality,

$$(3.14) \quad \begin{aligned} &A((I - P_{k-1})K_k^m u, K_k^m u) \\ &\leq C_\alpha^2 \left\{ \alpha \gamma_k \frac{\|A_k K_k^m u\|_k^2}{\lambda_k} + (1 - \alpha) \gamma_k^{-\alpha/(1-\alpha)} A(K_k^m u, K_k^m u) \right\} \end{aligned}$$

holds for any positive  $\gamma_k$ . By (3.5) and the symmetry of  $K_k$ ,

$$(3.15) \quad \frac{\|A_k K_k^m u\|_k^2}{\lambda_k} \leq C_R A((I - K_k)K_k^{2m} u, u).$$

Since the spectrum of  $K_k$  is contained in the interval  $[0, 1]$ ,

$$(3.16) \quad \begin{aligned} A((I - K_k)K_k^{2m} u, u) &\leq \frac{1}{2m} \sum_{i=0}^{2m-1} A((I - K_k)K_k^i u, u) \\ &= \frac{1}{2m} A((I - K_k^{2m})u, u). \end{aligned}$$

Combining (3.13)–(3.16) gives

$$(3.17) \quad \begin{aligned} A((I - B_k^s A_k)u, u) &\leq [(1 - \delta_{k-1})C_\alpha^2(1 - \alpha)\gamma_k^{-\alpha/(1-\alpha)} + \delta_{k-1}]A(K_k^{2m} u, u) \\ &\quad + (1 - \delta_{k-1})C_\alpha^2 C_R \frac{\alpha}{2m} \gamma_k A((I - K_k^{2m})u, u). \end{aligned}$$

To prove that (3.11) holds for  $k$ , it suffices to choose  $\gamma_k$  so that

$$(3.18) \quad (1 - \delta_{k-1})C_\alpha^2(1 - \alpha)\gamma_k^{-\alpha/(1-\alpha)} + \delta_{k-1} \leq \delta_k$$

and

$$(3.19) \quad (1 - \delta_{k-1})C_\alpha^2 C_R \frac{\alpha}{2m} \gamma_k \leq \delta_k.$$

We set  $\gamma_k$  by

$$(3.20) \quad (1 - \delta_{k-1})C_\alpha^2 C_R \frac{\alpha}{2m} \gamma_k = \delta_{k-1}.$$

Note that by definition,  $\delta_k$  is an increasing function of  $k$  and hence (3.20) immediately implies (3.19). We need only verify that (3.18) holds for this choice of  $\gamma_k$ . This is equivalent to showing

$$(3.21) \quad (1 - \delta_{k-1})C_\alpha^2(1 - \alpha)\gamma_k^{-\alpha/(1-\alpha)} \leq \delta_k - \delta_{k-1}.$$

Let  $\mathcal{D}(k) = m^\alpha + M_\alpha(j_0 + k)^{(1-\alpha)/\alpha}$ . A direct computation gives that

$$(3.22) \quad \delta_k - \delta_{k-1} = \frac{M_\alpha m^\alpha [(j_0 + k)^{(1-\alpha)/\alpha} - (j_0 + k - 1)^{(1-\alpha)/\alpha}]}{\mathcal{D}(k)\mathcal{D}(k-1)}.$$

The left-hand side of (3.21) can be written

$$(3.23) \quad (C_\alpha^2)^{1/(1-\alpha)} \frac{(1 - \alpha)}{(j_0 + k - 1)\mathcal{D}(k-1)} \left( \frac{C_R \alpha}{2M_\alpha} \right)^{\alpha/(1-\alpha)}.$$

A straightforward exercise in calculus, noting that  $k \geq 1$ , gives

$$(3.24) \quad \begin{aligned} & (j_0 + k - 1)[(j_0 + k)^{(1-\alpha)/\alpha} - (j_0 + k - 1)^{(1-\alpha)/\alpha}] \\ & \geq \left( \frac{1 - \alpha}{\alpha} \right) (j_0 + k)^{(1-\alpha)/\alpha} \left( \frac{j_0}{1 + j_0} \right)^l, \end{aligned}$$

where

$$l = \begin{cases} 1 & \text{if } \alpha \geq 1/2, \\ \frac{1 - \alpha}{\alpha} & \text{if } \alpha < 1/2. \end{cases}$$

Combining (3.22)–(3.24) shows that (3.21) holds if

$$(3.25) \quad \begin{aligned} & (\alpha C_\alpha^2)^{1/(1-\alpha)} \left( \frac{C_R}{2} \right)^{\alpha/(1-\alpha)} \\ & \leq M_\alpha^{\alpha/(1-\alpha)} \left( \frac{j_0}{1 + j_0} \right)^l \frac{m^\alpha M_\alpha (j_0 + k)^{(1-\alpha)/\alpha}}{\mathcal{D}(k)}. \end{aligned}$$

We note that  $\tilde{M}_\alpha$  was chosen so that

$$(\alpha C_\alpha^2)^{1/(1-\alpha)} \left( \frac{C_R}{2} \right)^{\alpha/(1-\alpha)} = \left( \frac{j_0}{1 + j_0} \right)^l \tilde{M}_\alpha^{\alpha/(1-\alpha)}.$$

The definition of  $M_\alpha$  implies

$$(\alpha C_\alpha^2)^{1/\alpha} \left( \frac{C_R}{2} \right)^{\alpha/(1-\alpha)} \leq \left( \frac{j_0}{1 + j_0} \right)^l M_\alpha^{\alpha/(1-\alpha)} \frac{M_\alpha}{1 + M_\alpha}.$$

Inequality (3.25) then follows from

$$\frac{M_\alpha}{1 + M_\alpha} \leq \frac{m^\alpha M_\alpha (j_0 + k)^{(1-\alpha)/\alpha}}{\mathcal{D}(k)}.$$

This completes the proof of Theorem 1.

**Remark 3.2.** As is clear from the proof of the theorem, any positive  $j_0$  leads to a bound for  $\delta_j$  with  $\varepsilon_j$  given by (3.7). The best choice is evidently a value which minimizes  $\delta_j$ .

Our second theorem applies to the nonsymmetric  $\mathcal{V}$  cycle.

**THEOREM 2.** Assume that (3.3) and (3.4) hold and define  $B_j^n$  by Algorithm N with  $p = 1$  and  $m(k) = m$  for all  $k$ . Then (3.2) holds with

$$\delta_j = 1 - \varepsilon_j,$$

where  $\varepsilon_j$  is defined as in Theorem 1 (see (3.7)–(3.10)).

*Proof.* We shall prove that

$$(3.26) \quad A((I - B_i^n A_i)u, (I - B_i^n A_i)u) \leq \delta_i A(u, u) \quad \text{for all } u \in M_i,$$

by induction on  $i$ . For  $i = 0$ , there is nothing to prove. Assume that (3.26) holds for  $i = k - 1$ . By (2.10) and the induction hypothesis,

$$(3.27) \quad \begin{aligned} A((I - B_k^n A_k)u, (I - B_k^n A_k)u) &\leq A((I - P_{k-1})K_k^m u, K_k^m u) \\ &\quad + \delta_{k-1} A(P_{k-1}K_k^m u, K_k^m u). \end{aligned}$$

Notice that the terms on the right-hand sides of Eqs. (3.12) and (3.27) are identical. The arguments following (3.12) show that the right-hand side of (3.27) can be bounded by  $\delta_k$ . This completes the proof of Theorem 2.

*Remark 3.3.* Theorems 1 and 2 give bounds for the convergence factor  $\delta_k$  in (3.1) and (3.2) in terms of  $\alpha$ ,  $m$  and  $j$ . In finite element applications,  $j$  (the number of levels) is proportional to a logarithm of the mesh size. Theorems 1 and 2 show that the multigrid  $\mathcal{V}$  cycles still converge for any  $m$  but the convergence rate may deteriorate with a power of  $j$  depending upon the  $\alpha$  for which (3.3) holds. A discussion of the relationship between  $\alpha$  and domain/operator regularity is given in Section 5 (see Proposition 5.1).

*Remark 3.4.* The theorems guarantee that the convergence factor  $\delta_j$  goes to zero faster than  $c/m^\alpha$  when  $j$  is held fixed and  $m$  tends to infinity. These rates are in agreement with the results of numerical computations presented in Section 6.

*Remark 3.5.* The results given in Theorems 1 and 2 for  $\alpha < 1$  are new. The proof gives rise to a somewhat simpler analysis than that already in the literature [2], [4], [9] for  $\alpha = 1$ . Indeed, by (3.17), if  $\alpha = 1$ , it suffices to take  $\delta_k$  to be the solution  $\delta$  of

$$(1 - \delta)C_\alpha^2 \frac{C_R}{2m} = \delta,$$

i.e.,

$$(3.28) \quad \delta = \frac{C_\alpha^2 C_R}{2m + C_\alpha^2 C_R}.$$

This value for  $\delta$  agrees with the results derived by the earlier analysis [2], [4]. Note also that the expression for  $\delta_j$  given by (3.6) tends to (3.28) as  $\alpha \rightarrow 1$ .

We next give results for the  $\mathcal{W}$  cycle multigrid algorithms. As previously mentioned, the uniform convergence of the  $\mathcal{W}$  cycle algorithm independent of the number of levels was shown in [13], [14]. Our results give convergence bounds which exhibit the explicit dependence on  $\alpha$  and  $m$ .

**THEOREM 3.** Assume that (3.3) and (3.4) hold and define  $B_j^s$  by Algorithm S with  $p = 2$  and  $m(k) = m$  for all  $k$ . Then (3.1) holds with  $\delta_k = \delta$  (independent of  $k$ ) given by

$$(3.29) \quad \delta = (1 + m/M_\alpha)^{-\alpha},$$

where

$$(3.30) \quad M_\alpha = 2^{1/\alpha} (C_\alpha^2)^{1/\alpha} \frac{\alpha C_R}{2} (1 - \alpha)^{(1-\alpha)/\alpha}.$$



*Proof.* We proceed as in the proof of Theorem 1. However, since  $p = 2$ , (3.12) and (3.13) get replaced by

$$(3.31) \quad A((I - B_k^s A_k)u, u) \leq A((I - P_{k-1})K_k^m u, K_k^m u) + \delta^2 A(P_{k-1}K_k^m u, K_k^m u)$$

$$(3.32) \quad \leq (1 - \delta^2)A((I - P_{k-1})K_k^m u, K_k^m u) + \delta^2 A(K_k^m u, K_k^m u).$$

The same reasoning which leads to (3.17) gives that

$$(3.33) \quad A((I - B_k^s A_k)u, u) \leq [(1 - \delta^2)C_\alpha^2(1 - \alpha)\gamma^{-\alpha/(1-\alpha)} + \delta^2]A(K_k^{2m}u, u) + (1 - \delta^2)C_\alpha^2 C_R \frac{\alpha}{2m} \gamma A((I - K_k^{2m})u, u)$$

holds for any positive  $\gamma$ . We define  $\gamma$  by the equation

$$(3.34) \quad (1 - \delta^2)C_\alpha^2 C_R \frac{\alpha}{2m} \gamma = \delta.$$

It then suffices to show that for  $\gamma$  defined by (3.34),

$$(1 - \delta^2)C_\alpha^2(1 - \alpha)\gamma^{-\alpha/(1-\alpha)} + \delta^2 \leq \delta,$$

or

$$(3.35) \quad (1 - \delta)^{\alpha/(1-\alpha)} (C_\alpha^2)^{1/(1-\alpha)} (1 - \alpha) \left( \frac{C_R \alpha}{2m} \right)^{\alpha/(1-\alpha)} \leq \left( \frac{\delta}{1 + \delta} \right)^{1/(1-\alpha)}.$$

A straightforward manipulation shows that (3.35) will be satisfied if

$$(3.36) \quad 2^{-1/\alpha} M_\alpha = (C_\alpha^2)^{1/\alpha} \frac{\alpha C_R}{2} (1 - \alpha)^{(1-\alpha)/\alpha} \leq \left( \frac{\delta}{1 + \delta} \right)^{1/\alpha} m (1 - \delta)^{-1}.$$

However, for  $y = 1/\delta$ ,

$$(3.37) \quad \left( \frac{\delta}{1 + \delta} \right)^{1/\alpha} \frac{m}{M_\alpha} (1 - \delta)^{-1} = \frac{y^{1+1/\alpha} - y}{(y - 1)(1 + y)^{1/\alpha}}$$

$$(3.38) \quad \geq \frac{y^{1/\alpha}}{(1 + y)^{1/\alpha}} \geq 2^{-1/\alpha},$$

where we used  $y^{1/\alpha} > y$  for  $y > 1$ . The theorem follows from (3.36) and (3.38).

**THEOREM 4.** Assume that (3.3) and (3.4) hold and define  $B_j^n$  by Algorithm N with  $p = 2$  and  $m(k) = m$  for all  $k$ . Then (3.2) holds with  $\delta_k = \delta$  given by (3.29).

*Proof.* The proof of Theorem 4 is a slight modification of the proof of Theorem 3. We use (2.10) to get

$$(3.39) \quad A((I - B_k^s A_k)u, (I - B_k^s A_k)u) \leq A((I - P_{k-1})K_k^m u, K_k^m u) + \delta^2 A(P_{k-1}K_k^m u, K_k^m u).$$

Notice that the terms on the right-hand sides of Eqs. (3.31) and (3.39) are identical. The arguments following (3.31) show that the right-hand side of (3.39) can be bounded by  $\delta$ . This completes the proof of Theorem 4.

**Remark 3.6.** Theorems 3 and 4 show that the multigrid  $\mathcal{W}$  cycles converge for any  $m$  and any  $\alpha$ . The convergence estimates tend to zero as  $m$  gets larger and deteriorate as  $\alpha$  tends to zero.

We next consider a generalized symmetric  $\mathcal{V}$  cycle.

**THEOREM 5.** Assume that (3.3) and (3.4) hold and define  $B_j^s$  by Algorithm S with  $p = 1$ . Assume that  $m(k)$  satisfies

$$(3.40) \quad \beta_0 m(k) \leq m(k-1) \leq \beta_1 m(k).$$

Here we assume that  $\beta_0$  and  $\beta_1$  are constants which are greater than one and independent of  $k$ . Then (3.1) holds with

$$(3.41) \quad \delta_j = 1 - \varepsilon_j,$$

where

$$(3.42) \quad \varepsilon_j = \frac{m(j)^\alpha}{m(j)^\alpha + M_\alpha}.$$

In (3.42),  $M_\alpha$  is defined by

$$M_\alpha = \left( \frac{1 + \tilde{M}_\alpha}{\tilde{M}_\alpha} \right)^{(1-\alpha)/\alpha} \tilde{M}_\alpha,$$

where

$$\tilde{M}_\alpha = (C_\alpha^2)^{1/\alpha} (1-\alpha)^{(1-\alpha)/\alpha} \left( \frac{\alpha C_R \beta_1}{2} \right) (\beta_0^\alpha - 1)^{-(1-\alpha)/\alpha}.$$

*Proof.* The proof of this theorem follows along the lines of the proof of Theorem 1. In fact, following the arguments in the proof of Theorem 1 (see (3.21)), we see that it suffices to show that

$$(3.43) \quad (1 - \delta_{k-1}) C_\alpha^2 (1 - \alpha) \gamma_k^{-\alpha/(1-\alpha)} \leq \delta_k - \delta_{k-1}$$

holds with  $\delta_k$  defined by (3.41)–(3.42), where  $\gamma_k$  is defined by

$$(3.44) \quad (1 - \delta_{k-1}) C_\alpha^2 C_R \frac{\alpha}{2m(k)} \gamma_k = \delta_{k-1}.$$

Let  $\mathcal{D}(k) = m(k)^\alpha + M_\alpha$ . A direct computation gives that

$$(3.45) \quad \delta_k - \delta_{k-1} \geq \frac{M_\alpha m(k)^\alpha (\beta_0^\alpha - 1)}{\mathcal{D}(k) \mathcal{D}(k-1)}.$$

The left-hand side of (3.43) can be written

$$(3.46) \quad (C_\alpha^2)^{1/(1-\alpha)} \frac{(1-\alpha)}{\mathcal{D}(k-1)} \left( \frac{\alpha C_R m(k-1)}{2M_\alpha m(k)} \right)^{\alpha/(1-\alpha)}.$$

Combining (3.45) and (3.46) shows that (3.43) holds if

$$(3.47) \quad (C_\alpha^2)^{1/(1-\alpha)} (1-\alpha) \left( \frac{\alpha C_R \beta_1}{2} \right)^{\alpha/(1-\alpha)} \leq (\beta_0^\alpha - 1) M_\alpha^{\alpha/(1-\alpha)} \frac{m(k)^\alpha M_\alpha}{\mathcal{D}(k)}.$$

However, the definitions of  $\tilde{M}_\alpha$  and  $M_\alpha$  imply

$$\begin{aligned} & (C_\alpha^2)^{1/(1-\alpha)} (1-\alpha) \left( \frac{\alpha C_R \beta_1}{2} \right)^{\alpha/(1-\alpha)} \\ &= (\beta_0^\alpha - 1) \tilde{M}_\alpha^{\alpha/(1-\alpha)} \leq (\beta_0^\alpha - 1) M_\alpha^{\alpha/(1-\alpha)} \frac{M_\alpha}{1 + M_\alpha} \\ &\leq (\beta_0^\alpha - 1) M_\alpha^{\alpha/(1-\alpha)} \frac{m(k)^\alpha M_\alpha}{\mathcal{D}(k)}. \end{aligned}$$

This completes the proof of the theorem.

Remark 2.1 gives one example of a  $\mathcal{V}$  cycle multigrid algorithm with variable  $m$  satisfying the hypothesis of the theorem ( $\beta_0 = \beta_1 = 2$ ). For a fixed  $\beta_0 > 1$ , one can define a sequence of  $m(k)$ 's as follows. Let  $m(j)$  be an arbitrary positive integer and define, by induction,  $m(k)$  to be the least integer greater than or equal to  $\beta_0 m(k+1)$ . For example, taking  $\beta_0 = 3/2$ ,  $\beta_1 = 2$  and  $m(j) = 1$  gives rise to the sequence  $m(j), m(j-1), \dots = 1, 2, 3, 5, 8, 12, 18, \dots$

The following result gives the analogous theorem in the nonsymmetric case.

**THEOREM 6.** *Assume that (3.3) and (3.4) hold and define  $B_j^n$  by Algorithm N with  $p = 1$  and  $m(k)$  satisfying (3.40). Then (3.2) holds with  $\delta_j$  given by (3.41)–(3.42).*

*Proof.* The theorem easily follows from (3.27) and the estimates derived in the proof of Theorem 5.

**4. Using Multigrid to Solve (2.1) and Related Systems.** In this section we shall consider a number of iterative techniques which use multigrid to solve (2.1) and related systems. We shall see that Theorems 1–6 give rise to estimates on the rate of convergence for the resulting iterative schemes. These observations are not new, however are included to indicate some of the ways in which multigrid can be used to solve problems.

We first consider the symmetric multigrid operators. Combining (2.9), (3.1) and an obvious change of variable gives that

$$(4.1) \quad (1 - \delta)(A_j^{-1}u, u)_j \leq (B_j^s u, u)_j \leq (A_j^{-1}u, u)_j \quad \text{for all } u \in \mathcal{M}_j$$

holds for  $\delta$  given by (3.6), (3.29) or (3.41). Inequalities of the form of (4.1) imply that  $B_j^s$  is a good preconditioner for  $A_j$ , and hence many preconditioned iterative techniques can be applied to solve (2.1) or similar problems corresponding to other comparable operators. Specifically, let  $\tilde{A}_j: \mathcal{M}_j \mapsto \mathcal{M}_j$  be another symmetric (with respect to the  $(\cdot, \cdot)_j$ -inner product) operator which satisfies comparability estimates of the form

$$(4.2) \quad c_0(\tilde{A}_j u, u)_j \leq (A_j u, u)_j \leq c_1(\tilde{A}_j u, u)_j \quad \text{for all } u \in \mathcal{M}_j.$$

Then  $B_j^s$  can be used as a preconditioner for the solution of the problem: Given  $f \in \mathcal{M}_j$ , find  $v \in \mathcal{M}_j$  satisfying

$$(4.3) \quad \tilde{A}_j v = f.$$

The most straightforward preconditioned iterative method is the linear method given by

$$(4.4) \quad v^{n+1} = v^n + \tau B_j^s(f - \tilde{A}_j v^n).$$

Let

$$||| \cdot ||| \equiv (\tilde{A}_j \cdot, \cdot)^{1/2}$$

and set  $e^n = v - v^n$ . Then iteration (4.4) is convergent for an appropriate choice of  $\tau$ . Furthermore, if  $\tilde{A}_j = A_j$ , then (4.4) is convergent for  $\tau < 2$  and

$$(4.5) \quad |||e^{n+1}||| \leq \rho |||e^n|||,$$

where  $\rho = \max(|1 - \tau|, |1 - \tau + \tau\delta|)$ . Taking  $\tau = 1$  gives a rate of  $\delta$  per step, while the optimal convergence is achieved by setting  $\tau = 2/(2 - \delta)$ .

It is not difficult to see that  $e^n = (I + P_\tau(B_j^s \tilde{A}_j))e^0$  for an appropriate polynomial  $P_\tau$ . More generally, the iterative scheme defined by

$$v^n = v^0 - P(B_j^s A_j)(v - v^0)$$

gives rise to the error equation

$$e^n = (I + P(B_j^s \tilde{A}_j))e^0.$$

An accelerated linear scheme is defined by making an appropriate choice of  $P$ . An optimal choice of the  $P$  can be made by use of Chebyshev polynomials (see, for example, [7]). For such a choice, the error  $e^n \equiv v - v^n$  can be bounded by

$$(4.6) \quad |||e^n||| \leq 2 \left( \frac{\sqrt{K} - 1}{\sqrt{K} + 1} \right)^n |||e^0|||,$$

where  $K$  (the condition number of  $B_j^s \tilde{A}_j$ ) is bounded by  $c_1/(c_0(1 - \delta))$ . The coefficients of the polynomial  $P$  depend upon *a priori* estimates for the largest and smallest eigenvalues of  $B_j^s \tilde{A}_j$ .

It is possible to set up nonlinear iterations which use  $B_j^s$  as a preconditioner for the solution of (4.3). One well-known candidate results from the conjugate gradient method. The solution  $v$  of (4.3) satisfies

$$(4.7) \quad B_j^s \tilde{A}_j v = B_j^s f.$$

The operator  $B_j^s \tilde{A}_j$  is symmetric in the  $(\tilde{A}_j \cdot, \cdot)_j$ -inner product (see Remark 2.2) and is positive definite by (4.1) and (4.2). The conjugate gradient method can be directly applied to (4.7) in this inner product to produce a sequence  $\{v_n\}$  which converges to  $v$ . In fact, the error for this iteration is also bounded by (4.6). The conjugate gradient scheme has the additional property that estimates for eigenvalues of  $B_j^s \tilde{A}_j$  need not be known *a priori* for its application.

We next consider the use of nonsymmetric multigrid algorithms for the solution of (2.1). The linear iteration (4.4) with  $\tau = 1$  converges and satisfies the inequality

$$(4.8) \quad |||e^{n+1}||| \leq \delta^{1/2} |||e^n|||,$$

where  $\delta$  satisfies (3.2).

**Remark 4.1.** The estimates for  $\delta_j$  satisfying (3.1) given by Theorem 1 are identical to those for  $\delta_j$  satisfying (3.2) given by Theorem 2. Using the estimates of the theorems, we find that  $\tau = 1$  in (4.5) should converge at a rate which is twice as fast as that of (4.8). Thus, the extra smoothings used in the symmetric algorithm lead to an iterative method which converges in half the number of iterations. A similar observation holds for the symmetric and nonsymmetric  $\mathcal{W}$  and generalized  $\mathcal{V}$  cycle algorithms.

**Remark 4.2.** We have considered applying the multigrid algorithms to the solution of the ‘algebraic’ problem (2.1) or the related problem (4.3). In the case of finite element or finite difference applications, the so-called full multigrid process can be used to get an ‘accurate’ approximation to the solution of (2.1) with computational work proportional to that required for a reduction on the finest grid level (cf. [3]).

**5. Estimate (3.3) and Applications.** We give a proof of the regularity and approximation estimate (3.3) in this section in the case of a typical finite element application of multigrid. Although the proof is given in [3], we include

it for completeness. We next discuss application to finite differences. We finally consider an example where symmetric Gauss-Seidel iteration is used to define  $R_k$ .

We shall consider the problem of approximating the solution  $U$  of

$$(5.1) \quad \begin{aligned} LU &= F && \text{in } \Omega, \\ U &= 0 && \text{on } \partial\Omega. \end{aligned}$$

Here  $\Omega$  is a domain in  $n$ -dimensional Euclidean space and  $L$  is given by

$$Lv = - \sum_{i,j=1}^n \frac{\partial}{\partial x_i} \left( a_{ij} \frac{\partial v}{\partial x_j} \right),$$

with  $\{a_{ij}\}$  uniformly positive definite and bounded on  $\Omega$ . The form  $A$  used in the multigrid algorithm is the bilinear form corresponding to the operator  $L$  and is defined by

$$(5.2) \quad A(v, w) = \sum_{i,j=1}^n \int_{\Omega} a_{ij} \frac{\partial v}{\partial x_i} \frac{\partial w}{\partial x_j} dx.$$

This form is defined for all  $v$  and  $w$  in the Sobolev space  $H^1(\Omega)$  (the space of distributions in  $L^2(\Omega)$  with square integrable first derivatives). Clearly,  $U$  is the solution of

$$A(U, \theta) = (F, \theta) \quad \text{for all } \theta \in H_0^1(\Omega),$$

where  $H_0^1(\Omega)$  is the subspace of  $H^1(\Omega)$  of functions which vanish in the appropriate sense on  $\partial\Omega$  and  $(\cdot, \cdot)$  denotes the  $L^2$ -inner product on  $\Omega$ .

We assume that  $\Omega$  has been triangulated with a sequence of quasi-uniform triangulations  $\Omega = \bigcup_i \tau_k^i$  of size  $h_k$  for  $k = 0, \dots, j$ , where the quasi-uniformity constants are independent of  $k$  (cf. [3]). We further assume that there is a constant  $c$ , independent of  $k$ , such that  $h_{k-1} \leq ch_k$ . These triangulations should be nested in the sense that any triangle  $\tau_{k-1}^l$  can be written as a union of triangles of  $\{\tau_k^i\}$ . We define  $\mathcal{M}_k$  to be the set of piecewise linear functions (with respect to the triangulation  $\bigcup_i \tau_k^i$ ) which vanish on  $\partial\Omega$ .

To avoid the inversion of  $L^2$  Gram matrices, we define  $(\cdot, \cdot)_k$  as a discrete  $L^2$ -inner product. Let  $\{y_k^i\}$  be the collection of nodes corresponding to the triangulation for  $\mathcal{M}_k$ . We set

$$(5.3) \quad (u, v)_k = h_k^n \sum_i u(y_k^i) v(y_k^i).$$

Note that the quasi-uniformity of the triangulations implies that the discrete form  $(\cdot, \cdot)_k$  is equivalent to the form  $(\cdot, \cdot)$  on the subspace  $\mathcal{M}_k$ . We seek the Galerkin approximation  $v \in \mathcal{M}_j$  (to the solution  $U$  of (5.1)) defined by

$$(5.4) \quad A(v, \phi) = (F, \phi) \quad \text{for all } \phi \in \mathcal{M}_j.$$

Equation (5.4) can be rewritten as

$$(5.5) \quad A(v, \phi) = (\tilde{F}, \phi)_j \quad \text{for all } \phi \in \mathcal{M}_j,$$

with an obvious choice of  $\tilde{F} \in \mathcal{M}_j$ . We derive (3.3) under the following elliptic regularity assumption. There exists a constant  $C$  such that

$$(5.6) \quad \|U\|_{H^{1+\alpha}(\Omega)} \leq C \|F\|_{H^{\alpha-1}(\Omega)}$$

holds for solutions  $U$  of (5.1). The norms  $H^s(\Omega)$  are Sobolev norms of order  $s$  and are defined in, for example, [10], [15].

The following lemma is given in [3] and depends on the quasi-uniformity assumptions on the mesh defining  $\mathcal{M}_k$ .

LEMMA 5.1. *Let  $0 \leq s \leq 1$ . There are constants  $c_0$  and  $c_1$  which are independent of  $M_0 \subset \cdots \subset M_j$  and satisfy*

$$c_0 \|A_k^{s/2} u\|_k \leq \|u\|_{H^s(\Omega)} \leq c_1 \|A_k^{s/2} u\|_k \quad \text{for all } u \in M_k.$$

Using the above lemma, we shall prove the following proposition.

PROPOSITION 5.1. *Assume that (5.6) holds for some  $\alpha$  in  $(0, 1]$ . Then (3.3) also holds for that  $\alpha$ .*

*Remark 5.1.* It is well known that estimates of the form of (5.6) do not in general hold for all  $\alpha$ , and the range of  $\alpha$ 's for which they hold depends upon the regularity of the coefficients defining  $L$  and the smoothness of  $\partial\Omega$ . We have computational evidence (see Example 6.1) which suggests that in such instances (3.3) will not hold for all  $\alpha$  with constants  $C_\alpha^2$  independent of  $h$ . Thus, the discrete result (3.3) is tied strongly to the elliptic regularity result (5.6).

*Proof of Proposition 5.1.* Let  $u \in M_k$ . Applying Schwarz's inequality and Lemma 5.1 gives

$$\begin{aligned} A((I - P_{k-1})u, u) &\leq \|A_k^{\alpha/2} u\| \|A_k^{-\alpha/2} (I - P_{k-1})u\| \\ (5.7) \quad &\leq \|A_k^{\alpha/2} u\| \|A_k^{(1-\alpha)/2} (I - P_{k-1})u\|_k \\ &\leq \|A_k^{\alpha/2} u\| \|(I - P_{k-1})u\|_{H^{1-\alpha}(\Omega)}. \end{aligned}$$

By Hölder's inequality,

$$(5.8) \quad \|A_k^{\alpha/2} u\| \leq \left( A(u, u)^{1-\alpha} \|A_k u\|_k^{2\alpha} \right)^{1/2}.$$

By standard error analysis techniques for finite element methods, employing duality and (5.6) (cf. [1], [6]), it follows that

$$(5.9) \quad \|(I - P_{k-1})u\|_{H^{1-\alpha}(\Omega)} \leq ch_{k-1}^\alpha A((I - P_{k-1})u, u)^{1/2}.$$

By the quasi-uniformity assumption of the mesh,  $\lambda_k \leq ch_k^{-2}$ . Hence, the proposition results from combining (5.7)–(5.9).

We next consider applying the theorems in the finite difference case. Here we consider a uniform rectangular grid with nodes  $(i, l)$ . We are to invert the five-point operator which is defined for grid points  $(i, l) \in \Omega$  by

$$(5.10) \quad (\tilde{A}_j u)_{i,l} = h_j^{-2} (4u_{i,l} - u_{i-1,l} - u_{i+1,l} - u_{i,l-1} - u_{i,l+1}),$$

where we set  $u_{i,l} = (\tilde{A}_j u)_{i,l} = 0$  for  $(i, l) \notin \Omega$ . We define a triangulation by breaking each rectangle into two triangles and set  $M_j$  to be the space of piecewise linear functions with respect to this triangulation which vanish on the nodes not in  $\Omega$ . It is well known that

$$(\tilde{A}_j u, u)_j = (A_j u, u)_j,$$

where  $A_j$  is the operator defined by (5.2) and (5.3) with  $L$  equal to minus the Laplacian  $(-\Delta)$ . Consequently, if we define a nested sequence of subspaces  $M_k$  as described above, then Proposition 5.1 applies and shows that the multigrid algorithms (defined using these spaces, (5.2), (5.3) and  $A_j$ ) satisfy the convergence estimates of the theorems. These are clearly multigrid algorithms for  $\tilde{A}_j$ . Note that for these algorithms, both the 'interpolation operators' and 'residual computation operators' which appear in the so-called 'finite difference' multigrid are defined by

the way nodal basis elements in  $\mathcal{M}_{k+1}$  are combined to get nodal basis elements in  $\mathcal{M}_k$ .

We finally consider an example where symmetric Gauss-Seidel iteration is used to define  $R_k$ . We consider the case where  $\tilde{A}_j = A_j$  is given by (5.10). Note that we do not require that the domain be a square. Then the matrix  $M_k$  corresponding to  $A_j$  in the usual nodal basis with any ordering can be written

$$M_k = \frac{4}{h_k^2}(I - L_k - U_k),$$

where  $U_k$  and  $L_k$  are strictly upper and lower triangular matrices and  $I$  is the identity matrix. We define  $R_k$  by

$$R_k = \frac{h_k^2}{4}(I - L_k)^{-1}(I - U_k)^{-1}.$$

The matrix corresponding to  $K_k$  is

$$I - (I - L_k)^{-1}(I - U_k)^{-1}(I - L_k - U_k)$$

which is similar to

$$(5.11) \quad (I - U_k)^{-1}U_kL_k(I - L_k)^{-1}.$$

The matrix (5.11) is clearly nonnegative and hence  $K_k$  is also nonnegative. We need only verify (3.4). The largest eigenvalue of  $A_j$  is greater than or equal to  $4/h^2$ , and hence (3.4) is equivalent to

$$(5.12) \quad \|(I - U_k)u\|_k^2 \leq C_R \|u\|_k^2.$$

But the sum of the absolute value of entries in any column of  $L_k + U_k$  is less than or equal to one, from which it follows that

$$(5.13) \quad \|U_k u\|_k \leq \|u\|_k.$$

Hence, (3.4) holds with  $C_R = 4$ .

**6. Numerical Examples.** Numerical experiments are presented in this section which illustrate some of the theoretical properties derived earlier in this paper. In some cases, actual values for  $C_\alpha$  are computed and the results of numerical experiments are compared with the theoretical bounds given by the theorems of Section 3 with these values of  $C_\alpha$ . These results suggest that, for finite element applications, the theorems provide reasonable convergence bounds. We also give an example which suggests that (3.3) does not in general hold for  $\alpha = 1$  with constant  $C_\alpha$  independent of  $h$ .

All of the examples presented will be for the problem

$$\begin{aligned} -\Delta U &= F && \text{in } \Omega, \\ U &= 0 && \text{on } \partial\Omega, \end{aligned}$$

where  $\Omega$  is either the unit square, the unit square with the upper right-hand sub-square removed (i.e., the ‘L-shaped’ domain), or the slit domain whose boundary consists of the boundary of the unit square together with the set  $\{(1/2, y) \mid y \in [1/2, 1]\}$ . We will consider a regular rectangular mesh where each rectangle is split into two triangles with piecewise linear elements and use the finite element multi-grid framework discussed in Section 5. As discussed in Section 5, the resulting

multigrid algorithm can be used to solve either the corresponding finite element or finite difference equations.

*Example 1.* In this example, we compute the best constant  $C_1(h_k)$  satisfying the inequality

$$A((I - P_{k-1})u, u) \leq C_1^2(h_k) \left( \frac{\|A_k u\|_k^2}{\lambda_k} \right) \quad \text{for all } u \in \mathcal{M}_k,$$

for both the square and slit domains. For the square, Proposition 5.1 gives that  $C_1(h_k)$  can be bounded independently of  $h_k$ . In fact, by using an odd extension and an analysis (cf. [2]) on the periodic domain, it can be shown that  $C_1^2(h_k) \leq 4 + 2\sqrt{2} = 6.828\dots$ . The computed values for  $C_1^2(h_k)$  in Table 6.1 satisfy the above estimate. In contrast, for the slit domain, Proposition 5.1 can only be applied for  $0 < \alpha < 1/2$ . Note that the results in Table 6.1 clearly suggest that it is not possible to satisfy (3.3) with  $\alpha = 1$  for the slit domain. In fact,  $C_1^2(h_k)$  seems to be growing like  $ch_k^{-1}$ . This example clearly indicates that the multigrid analysis for  $\alpha < 1$  is important since the assumption (3.3) cannot be expected to hold in general for  $\alpha = 1$ .

TABLE 6.1  
*Values for  $C_1^2(h_k)$  for the Square and Slit Domains.*

$h_k$	Square	Slit
1/16	5.8	7.8
1/32	6.3	12
1/64	6.6	20
1/128	6.6	36

*Example 2.* In this example, we consider the symmetric  $\mathcal{V}$  cycle of multigrid applied to the slit domain. We shall consider two algorithms. The first algorithm uses  $m(k) = 1$ , while the second uses variable smoothing defined by  $m(k) = 2^{j-k}$ . Both algorithms use  $R_k = \tilde{\lambda}_k^{-1} I_k$  where  $\tilde{\lambda}_k$  is the largest eigenvalue for the  $A_k$  with the square domain. We compute the best values of  $\delta_j$  satisfying (3.1) for the corresponding algorithms and set  $\varepsilon_j = 1 - \delta_j$ . According to Proposition 5.1, (3.3) holds for  $\alpha$  in the interval  $(0, 1/2)$  and from Theorem 1 we expect that  $\varepsilon_j$  ( $m = 1$ ) should go to zero nearly like

$$(6.1) \quad \tilde{\varepsilon}_j (m = 1) = (1 + M_\alpha(j_0 + j))^{-1}.$$

Unfortunately, we do not know the corresponding constant  $C_{1/2}$  defining  $M_{1/2}$  in Theorem 1. Instead, we have chosen  $M_{1/2} = .268$  and  $j_0 = 1.93$  in (6.1) to fit the computed results. The fact that this function fits the computed values of  $\varepsilon_j$  ( $m = 1$ ) so closely indicates that the  $\log(h)$  is really reflected in the computational behavior. However, the logarithmic growth is quite slow and the corresponding reductions (i.e.,  $\delta_j = .722$  at  $h_j = 1/256$ ) are rather remarkable. We also include the values of  $\varepsilon_j$  (variable  $m$ ) corresponding to the variable smoothing algorithm. Note that, as predicted by Theorem 5, these values for  $\varepsilon_j$  remain bounded away from zero independently of  $h$ .



TABLE 6.2  
*Values of  $\varepsilon_j$  for  $\mathcal{V}$  Cycle Algorithms on the Slit Domain.*

$h_j$	$\varepsilon_j$ ( $m = 1$ )	$\tilde{\varepsilon}_j$ ( $m = 1$ )	$\varepsilon_j$ (variable $m$ )
1/8	.45	.43	.414
1/16	.386	.390	.428
1/32	.347	.350	.424
1/64	.318	.320	.422
1/128	.296	.295	.420
1/256	.278	.273	.420

*Example 3.* In this example, we compare the values of  $\varepsilon_j$  for the L-shaped and square domains. We use the  $\mathcal{V}$  cycle with  $m = 1$  and  $R_k = \tilde{\lambda}_k^{-1} I_k$ . In the case of the L-shaped domain, Proposition 5.1 implies that (3.3) holds for  $0 < \alpha < 2/3$ . As can be seen from Table 6.3, the computed values of  $\varepsilon_j$  for the L-shaped domain are somewhat smaller than those for the square domain and somewhat larger than those for the slit domain. The  $\varepsilon_j$ 's corresponding to the L-shaped domain also decrease faster than those for the square while not as fast as those for the slit. This is in qualitative agreement with the theorems.

TABLE 6.3  
*Computed Values of  $\varepsilon_j$  for the Square and L-shaped Domains.*

$h_j$	$\varepsilon_j$ (square)	$\varepsilon_j$ (L-shaped)
1/8	.48	.46
1/16	.43	.42
1/32	.42	.40
1/64	.41	.38
1/128	.41	.37
1/256	.41	.36

*Example 4.* We include this example to indicate that the behavior for large  $m$  suggested by the theorems is consistent with the results of actual computations. We shall use the symmetric  $\mathcal{V}$  cycle on a grid of size  $h_j = 1/64$  with  $R_k = \tilde{\lambda}_k^{-1} I_k$  and  $m(k) = m$  for all  $k$ .

We first consider the case of  $\alpha = 1$ , i.e., the square domain. We shall compare the computed best constant  $\delta_j$  (computed) satisfying (3.1) with the theoretical estimate given by Theorem 1 as a function of  $m$ . In this case,

$$C_1^2 \leq 6.828\dots$$

as noted earlier. In this application,  $R_k = \tilde{\lambda}_k^{-1} I_k = \lambda_k^{-1} I_k$  and hence  $C_R = 1$ . Theorem 1 guarantees that

$$\delta_j \text{ (computed)} \leq \delta_j \text{ (theoretical)} = \frac{C_1^2}{2m + C_1^2}.$$

Table 6.4 gives the values of  $\delta_j$  (computed) and  $\delta_j$  (theoretical) as a function of  $m$ . We also fit the computed results (at  $m = 13$ ) with a function of the form

$$(6.2) \quad \delta_j \text{ (fit-1)} = \frac{C}{2m + C}.$$

TABLE 6.4  
*Computed, Theoretical, and Fit Values of  $\delta_j$  on the Square.*

$m$	$\delta_j$ (computed)	$\delta_j$ (theoretical)	$\delta_j$ (fit-1)
1	.59	.77	.55
5	.20	.40	.19
13	.085	.20	.085
25	.045	.12	.046
41	.027	.08	.029

Note that the fitted function provides a very good approximation to the computed values of  $\delta_j$  for large  $m$ .

We next consider the slit domain. We again report the computed values of  $\delta_j$  and compare with fitted functions. Since, for this example, (3.3) holds for  $0 < \alpha < 1/2$ , we should expect to be able to fit  $\delta_j$  with

$$(6.3) \quad \delta_j \text{ (fit-1/2)} = \frac{C}{\sqrt{2m} + C}.$$

Table 6.5 compares this fit (at  $m = 13$ ) with the computed results for  $\delta_j$ . Note that the computed results are going to zero slightly faster than the fit. For comparison, we also have fit the computed results ( $\delta_j$  (fit-1)) to a function of the form (6.2). The computed results go to zero more slowly than the fit to (6.2). Thus the actual computed results seem to show an asymptotic behavior which is somewhere between  $m^{-1}$  and  $m^{-1/2}$ . This is consistent with the results of the theorems since the theorems provide only pessimistic convergence bounds.

TABLE 6.5  
*Computed and Fit Values of  $\delta_j$  on the Slit.*

$m$	$\delta_j$ (computed)	$\delta_j$ (fit-1/2)	$\delta_j$ (fit-1)
1	.682	.38	.69
5	.287	.22	.30
13	.146	.146	.146
25	.089	.11	.08
41	.060	.09	.05

Department of Mathematics  
 Cornell University  
 Ithaca, New York 14853

Brookhaven National Laboratory  
 Upton, New York 11973

1. A. K. AZIZ & I. BABUŠKA, "Survey lectures on the mathematical foundations of the finite element method," in *The Mathematical Foundations of the Finite Element Method with Applications to Partial Differential Equations. Part I* (A. K. Aziz, ed.), Academic Press, New York, 1972, pp. 1-362.

2. R. E. BANK & C. C. DOUGLAS, "Sharp estimates for multigrid rates of convergence with general smoothing and acceleration," *SIAM J. Numer. Anal.*, v. 22, 1985, pp. 617-633.

3. R. E. BANK & T. F. DUPONT, "An optimal order process for solving elliptic finite element equations," *Math. Comp.*, v. 36, 1981, pp. 35–51.
4. D. BRAESS & W. HACKBUSCH, "A new convergence proof for the multigrid method including the  $V$ -cycle," *SIAM J. Numer. Anal.*, v. 20, 1983, pp. 967–975.
5. A. BRANDT, "Multi-level adaptive solutions to boundary-value problems," *Math. Comp.*, v. 31, 1977, pp. 333–390.
6. P. G. CIARLET, *The Finite Element Method for Elliptic Problems*, North-Holland, New York, 1978.
7. T. DUPONT, R. P. KENDALL & H. H. RACHFORD, "An approximate factorization procedure for solving self-adjoint elliptic difference equations," *SIAM J. Numer. Anal.*, v. 5, 1968, pp. 559–573.
8. R. P. FEDORENKO, "The speed of convergence of one iterative process," *USSR Comput. Math. and Math. Phys.*, v. 1, 1961, pp. 1092–1096.
9. W. HACKBUSCH, *Multi-Grid Methods and Applications*, Springer-Verlag, New York, 1985.
10. J. L. LIONS & E. MAGENES, *Problèmes aux Limites non Homogènes et Applications*, Dunod, Paris, 1968.
11. J. F. MAITRE & F. MUSY, "Algebraic formalization of the multigrid method in the symmetric and positive definite case—a convergence estimation for the  $V$ -cycle," *Multigrid Methods for Integral and Differential Equations* (D. J. Paddon and H. Holstein, eds.), Clarendon Press, Oxford, 1985.
12. J. MANDEL, "Multigrid convergence for nonsymmetric, indefinite variational problems and one smoothing step," in *Proc. Copper Mtn. Conf. Multigrid Methods*, Applied Math. Comput., 1986, pp. 201–216.
13. J. MANDEL, "Algebraic study of multigrid methods for symmetric, definite problems." (Preprint)
14. J. MANDEL, S. F. MCCORMICK & J. RUGE, "An algebraic theory for multigrid methods for variational problems." (Preprint)
15. S. F. MCCORMICK, "Multigrid methods for variational problems: Further results," *SIAM J. Numer. Anal.*, v. 21, 1984, pp. 255–263.
16. S. F. MCCORMICK, "Multigrid methods for variational problems: General theory for the  $V$ -cycle," *SIAM J. Numer. Anal.*, v. 22, 1985, pp. 634–643.
17. J. NEČAS, *Les Méthodes Directes en Théorie des Équations Elliptiques*, Academia, Prague, 1967.
18. R. A. NICOLAIDES, "On the  $l^2$  convergence of an algorithm for solving finite element equations," *Math. Comp.*, v. 31, 1977, pp. 892–906.