

THE ANALYSIS OF SMOOTHERS FOR MULTIGRID ALGORITHMS

JAMES H. BRAMBLE AND JOSEPH E. PASCIAK

ABSTRACT. The purpose of this paper is to provide a general technique for defining and analyzing smoothing operators for use in multigrid algorithms. The smoothing operators considered are based on subspace decomposition and include point, line, and block versions of Jacobi and Gauss-Seidel iteration as well as generalizations. We shall show that these smoothers will be effective in multigrid algorithms provided that the subspace decomposition satisfies two simple conditions. In many applications, these conditions are trivial to verify.

1. INTRODUCTION

In recent years, multigrid methods have been used extensively as tools for obtaining approximations to the solutions of partial differential equations (see the references in [10, 11, 12]). In conjunction, there has been intensive research into the theoretical understanding of these methods (cf. [1, 2, 3, 4, 6, 9, 11, 12, 13, 14, 18] and others). Many of the above papers present various analyses of multigrid methods which are often based on certain assumptions concerning the smoothing process. These assumptions are sometimes verified for specific examples. It is the purpose of this paper to present a general approach for developing smoothing operators and show that they work in multigrid methods provided that a few simple hypotheses are satisfied in the construction. For other estimates concerning smoothing operators in multigrid procedures, we refer the reader to [16] and the extended bibliography included there.

The smoothers for a given space are defined to be either the additive or multiplicative iterative scheme associated with a decomposition of the space (see (3.2) and Algorithm 3.1). Different smoothers result from distinct decompositions. Depending on the choice of subspaces in this decomposition, the technique can be used to generate many of the popular smoothing schemes used in multigrid iteration. For example, it can be used to generate point, line, and block Jacobi smoothing as well as point, line, and block Gauss-Seidel smoothing.

The construction of iterative schemes based on subspace decomposition is not a new idea. In fact, this technique has been used extensively for the construc-

Received October 8, 1990; revised July 1, 1991.

1991 *Mathematics Subject Classification.* 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 non-exclusive, 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 by the U.S. Army Research Office through the Mathematical Sciences Institute, Cornell University.

tion of preconditioners using overlapping domain decomposition (also known as Schwarz domain decomposition methods) [8, 20]. The hypotheses on the subspace decomposition required in this paper for multigrid smoothers are easier to satisfy than those required for the construction of effective preconditioners. Thus, the subspace decomposition used in most smoothing procedures will not give rise to a good preconditioner. We illustrate this in the following discussion.

Let \mathcal{M}_k be a finite-dimensional space with inner product $(\cdot, \cdot)_k$ and consider the problem of computing the solution $u \in \mathcal{M}_k$ of

$$A_k u = f,$$

for given $f \in \mathcal{M}_k$. Here, A_k is a symmetric and positive definite operator on \mathcal{M}_k . The additive and multiplicative versions of the smoothers are defined in terms of a decomposition

$$\mathcal{M}_k = \sum_{i=1}^l \mathcal{M}_k^i.$$

The basic hypothesis which is used to show that the resulting smoother is effective in a multigrid iteration is that there exists a constant c_0 such that every $v \in \mathcal{M}_k$ can be decomposed into $v = \sum v_i$ with $v_i \in \mathcal{M}_k^i$ satisfying

$$(1.1) \quad \sum_{i=1}^l (v_i, v_i)_k \leq c_0 (v, v)_k.$$

The corresponding hypothesis which is used to show that the smoother is an effective preconditioner for A_k is that there exists a constant C_0 such that every $v \in \mathcal{M}_k$ can be decomposed into $v = \sum v_i$ with $v_i \in \mathcal{M}_k^i$ satisfying

$$(1.2) \quad \sum_{i=1}^l (A_k v_i, v_i)_k \leq C_0 (A_k v, v)_k.$$

In finite element discretization of second-order elliptic partial differential equations, for functions in \mathcal{M}_k , $(\cdot, \cdot)_k$ is often equivalent to the L^2 inner product on the domain of consideration. In contrast, $(A_k \cdot, \cdot)_k$ is usually equivalent to the norm on the Sobolev space of order one. Condition (1.1) is often trivial to verify for many subspace decompositions. Most subspace decompositions which are used as multigrid smoothers (and satisfy (1.1) with bounded c_0) satisfy (1.2) only with a constant C_0 which grows large as the mesh parameter becomes small. Thus, the multigrid smoother would not be effective as a stand-alone iterative method.

The outline of the remainder of this paper is as follows. Section 2 describes the basic multigrid algorithm in an abstract setting and gives some of the conditions on the smoothers which are commonly assumed in various multigrid analyses. The general smoothing procedures based on subspace decomposition are described and analyzed in §3. In §4, we give theorems providing estimates for multigrid algorithms using these smoothers. Computer implementation of the smoothers is discussed in §5. In particular, it is shown that the commonly used multigrid smoothers can be generated by this technique with appropriate selection of the subspace decompositions. Finally, in §6, we discuss the finite element multigrid application.

2. THE MULTIGRID ALGORITHMS

In this section, we describe a symmetric multigrid cycling algorithm. For convenience, this algorithm is developed in an abstract Hilbert space setting and uses general smoothing operators. We then state two conditions involving these smoothing operators which are assumed for various analyses of multigrid.

We start by describing the general multigrid algorithm in an abstract setting. We assume that we are given a sequence of finite-dimensional inner product spaces

$$\mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_j.$$

The inner product on \mathcal{M}_k will be denoted by $(\cdot, \cdot)_k$. In addition, we assume that symmetric positive definite operators $A_k: \mathcal{M}_k \mapsto \mathcal{M}_k$ for $k = 1, \dots, j$ and “interpolation” operators $I_k: \mathcal{M}_{k-1} \mapsto \mathcal{M}_k$ are given. The multigrid algorithm gives rise to iterative procedures for the solution of the problem on \mathcal{M}_j , i.e., given $f \in \mathcal{M}_j$ find $u \in \mathcal{M}_j$ satisfying

$$(2.1) \quad A_j u = f.$$

The final ingredient needed to define the general multigrid algorithm is a sequence of linear (smoothing) operators $R_k: \mathcal{M}_k \mapsto \mathcal{M}_k$, for $k = 2, \dots, j$. We shall always take $R_1 = A_1^{-1}$. The point of this paper is to present a general approach for the definition of these operators as well as a unified analysis for showing that these operators are effective in multigrid procedures. We set

$$R_k^{(l)} = \begin{cases} R_k & \text{if } l \text{ is odd,} \\ R_k' & \text{if } l \text{ is even.} \end{cases}$$

Here, and throughout this manuscript, t will denote adjoint with respect to the inner product $(\cdot, \cdot)_k$.

We next define a general multigrid process for iteratively computing the solution of (2.1). This process is defined in the following algorithm in terms of a mathematical induction involving the subspace level. On each subspace \mathcal{M}_k , the multigrid iterative procedure can be viewed as a process which acts on both a function $F_k \in \mathcal{M}_k$ and an “approximation” W_k to the solution of

$$(2.2) \quad A_k U_k = F_k$$

and produces an improved approximation in \mathcal{M}_k to U_k (denoted by $\text{Mg}_k(W_k, F_k)$).

Algorithm 2.1. For $k = 1$, define $\text{Mg}_1(W_1, F_1) = A_1^{-1}F_1$, i.e., solve (2.2) exactly. For $k > 1$, $\text{Mg}_k(W_k, F_k)$ is defined in terms of $\text{Mg}_{k-1}(\cdot, \cdot)$ as follows:

- (1) Set $X^0 = W_k$ and $Q^0 = 0 \in \mathcal{M}_{k-1}$.
- (2) Define X^i , for $i = 1, \dots, m(k)$ by

$$(2.3) \quad X^i = X^{i-1} + R_k^{(i+m(k))}(F_k - A_k X^{i-1}).$$

- (3) Set $Y^{m(k)} = X^{m(k)} + I_k Q^p$, where Q^i for $i = 1, \dots, p$ is given by

$$(2.4) \quad Q^i = \text{Mg}_{k-1}(Q^{i-1}, P_{k-1}^0(F_k - A_k X^m)).$$

Here, P_{k-1}^0 is defined by

$$(2.5) \quad (P_{k-1}^0 v, \phi)_{k-1} = (v, I_k \phi)_k \quad \text{for all } \phi \in \mathcal{M}_{k-1}.$$

(4) For $i = m(k) + 1, \dots, 2m(k)$ define Y^i by

$$Y^i = Y^{i-1} + R_k^{(i+m(k))}(F_k - A_k Y^{i-1}).$$

(5) Set $\text{Mg}_k(W_k, F_k) = Y^{2m(k)}$.

The above algorithm is more general than those often described [2, 4, 12], in that it allows the use of general symmetric and nonsymmetric smoothers. Note that we have placed very few restrictions on the definition of the linear operators A_k , I_k , and R_k at this time.

The above multigrid procedure can be used to solve (2.1) by the following iteration,

$$(2.6) \quad u^l = \text{Mg}_j(u^{l-1}, f),$$

for initial iterate u^0 and $l = 1, 2, \dots$. Let $e^l = u - u^l$. Then it is possible to show that

$$e^l = E_j e^{l-1}$$

holds for a linear error reduction operator $E_j : \mathcal{M}_j \mapsto \mathcal{M}_j$. Accordingly, we can define a preconditioner B_j associated with the multigrid process by

$$B_j = (I - E_j)A_j^{-1} \quad \text{or} \quad E_j = I - B_j A_j.$$

Consequently, (2.6) is nothing more than the preconditioned linear iterative scheme

$$(2.7) \quad u^l = u^{l-1} + B_j(f - A_j u^{l-1}).$$

Alternatively, the linear operator B_j can be directly defined by the following algorithm.

Algorithm 2.2. Set $B_1 = A_1^{-1}$. Assume that B_{k-1} has been defined and define $B_k g$ for $g \in \mathcal{M}_k$ as follows:

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

$$(2.8) \quad x^i = x^{i-1} + R_k^{(i+m(k))}(g - A_k x^{i-1}).$$

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

$$(2.9) \quad q^i = q^{i-1} + B_{k-1}[P_{k-1}^0(g - A_k x^{m(k)}) - A_{k-1} q^{i-1}].$$

- (4) Define y^i for $i = m(k) + 1, \dots, 2m(k)$ by

$$(2.10) \quad y^i = y^{i-1} + R_k^{(i+m(k))}(g - A_k y^{i-1}).$$

- (5) Set $B_k g = y^{2m(k)}$.

In the above algorithm, we alternate between R_k and R_k^i in Step 2. In Step 4, we use the adjoints of the Step 2 smoothings applied in the reverse order. This results in a symmetric operator B_j . This form of the multigrid algorithm has been suggested in [5]. Nonsymmetric multigrid procedures which, for example,

do not include the smoothing of Step 4 have also been analyzed [4, 11, 12], etc. The hypotheses required on the smoothing operators are exactly the same as those used in the symmetric case.

There are two standard conditions concerning the smoothing operators which are often assumed as hypotheses in the analysis of multigrid algorithms. To describe these, we first define $K_k = I - R_k A_k$ and note that $K_k^* = I - R_k^t A_k$. Here, and throughout this paper, $*$ will denote adjoint with respect to the inner product $(A_k \cdot, \cdot)_k$.

(C.1) There is a constant C_R which does not depend on k such that the smoothing procedure satisfies

$$(2.11) \quad \frac{\|u\|_k^2}{\lambda_k} \leq C_R (\bar{R}_k u, u)_k \quad \text{for all } u \in \mathcal{M}_k.$$

Here, \bar{R}_k is either $(I - K_k^* K_k) A_k^{-1}$ or $(I - K_k K_k^*) A_k^{-1}$, λ_k is the largest eigenvalue of A_k , and $\|\cdot\|_k^2$ denotes the norm corresponding to the inner product $(\cdot, \cdot)_k$.

(C.2) Let $T_k = R_k A_k$. There is a constant $\theta < 2$ not depending on k satisfying

$$(2.12) \quad (A_k T_k v, T_k v)_k \leq \theta (A_k T_k v, v)_k.$$

The point of the present paper is to define general smoothing procedures and prove estimates of the form of (2.11) and (2.12) under simple hypotheses.

We shall state some convergence estimates from [5, 7], and [9] for Algorithm 2.2 in a later section. The following remarks show that the above two conditions are used in other multigrid theories as well.

Remark 2.1. Let $R_{k,\omega}$ correspond to the Richardson smoothing iteration defined by $R_{k,\omega} = \omega \lambda_k^{-1} I$ and $K_{k,\omega} = (I - R_{k,\omega} A_k)$ be the corresponding reducer. Inequality (2.11) in the case of $\bar{R}_k = (I - K_k^* K_k) A_k^{-1}$ can be rewritten as

$$(2.13) \quad (A_k K_k u, K_k u)_k \leq (A_k K_{k,\omega} u, u)_k \quad \text{for all } u \in \mathcal{M}_k,$$

with $\omega = 1/C_R$. This means that the smoothing process applied to any $u \in \mathcal{M}_k$ converges as fast as Richardson's method for some $\omega \in (0, 1)$. A hypothesis of the form of the above inequality was essentially used in [5, 14] and [15]. The Richardson method is perhaps the most natural smoothing procedure.

Remark 2.2. The following condition on the smoothing operator is used by Bank et al. (see [12, (4.6)]): There is a positive constant c satisfying

$$(2.14) \quad \frac{\|A_k K_k u\|_k^2}{\lambda_k} \leq c (A_k (I - K_k^* K_k) u, u)_k \quad \text{for all } u \in \mathcal{M}_k.$$

Note that for the appropriate definition of \bar{R}_k , (2.11) can be rewritten as

$$(2.15) \quad \frac{\|A_k v\|_k^2}{\lambda_k} \leq C_R (A_k (I - K_k K_k^*) v, v)_k \quad \text{for all } v \in \mathcal{M}_k.$$

We shall show that (2.15) implies (2.14). Indeed, taking $v = K_k u$ in (2.15) gives

$$\frac{\|A_k K_k u\|_k^2}{\lambda_k} \leq C_R (A_k (I - \bar{K}_k) \bar{K}_k u, u)_k$$

where $\bar{K}_k = K_k^* K_k$. Note that the spectrum of \bar{K}_k is in $[0,1)$, and hence

$$(A_k(I - \bar{K}_k)\bar{K}_k u, u)_k \leq (A_k(I - \bar{K}_k)u, u)_k = (A_k(I - K_k^* K_k)u, u)_k.$$

Combining the above two estimates proves (2.14).

3. GENERAL SMOOTHING PROCEDURES IN MULTIGRID ALGORITHMS

In this section, we shall define smoothing operators in terms of subspace decompositions. These procedures are related to overlapping domain decomposition and the classical Schwarz method and are generalizations of Jacobi and Gauss-Seidel iteration procedures. In this section, we shall show that the hypotheses (2.11) and (2.12) will follow from scaling, when appropriate, and an easily verified function decomposition inequality. Explicit examples providing such decompositions in the case of finite element multigrid applications are given in a later section.

The technique which we shall study for developing smoothers involves the use of a variant of overlapping domain decomposition. These methods are also referred to as ‘‘Schwarz overlapping’’ methods. We shall develop a smoother for the problem on \mathcal{M}_k . One starts with a decomposition of the space,

$$(3.1) \quad \mathcal{M}_k = \sum_{i=1}^l \mathcal{M}_k^i.$$

This sum may or may not be a direct sum.

Given the decomposition (3.1), there are two types of smoothers which can be defined. The first will be called the additive smoother and is defined by

$$(3.2) \quad R_k = \gamma \sum_{i=1}^l A_{k,i}^{-1} Q_k^i.$$

Here, $A_{k,i} : \mathcal{M}_k^i \mapsto \mathcal{M}_k^i$ is defined by

$$(A_{k,i} v, \chi)_k = (A_k v, \chi)_k \quad \text{for all } \chi \in \mathcal{M}_k^i$$

and $Q_k^i : \mathcal{M}_k \mapsto \mathcal{M}_k^i$ is the projection onto \mathcal{M}_k^i with respect to the inner product $(\cdot, \cdot)_k$. In addition, γ is a positive scaling factor which will be chosen later. We note that R_k is a symmetric operator with respect to the inner product $(\cdot, \cdot)_k$. Implementation issues involving the above smoother will be discussed in a later section.

To analyze the additive algorithm, we shall use a limited interaction hypothesis. To describe this property, we first introduce the projection $P_k^i : \mathcal{M}_k \mapsto \mathcal{M}_k^i$ defined by

$$(A_k P_k^i v, \chi)_k = (A_k v, \chi)_k \quad \text{for all } \chi \in \mathcal{M}_k^i.$$

We then define

$$\kappa_{im} = \begin{cases} 0 & \text{if } P_k^i P_k^m = 0, \\ 1 & \text{otherwise,} \end{cases}$$

and set

$$n_0 = \max_i \sum_{m=1}^l \kappa_{im}.$$

In our applications, n_0 remains small, even when l becomes large. Note that the matrix $\{\kappa_{im}\}$ is symmetric.

We shall use the following two conditions:

- (1) The subspaces satisfy a limited interaction property, i.e.,

$$(3.3) \quad n_0 \leq c_1,$$

with c_1 independent of k .

- (2) There exists a positive constant c_0 not depending on k such that for each $u \in \mathcal{M}_k$, there is a decomposition $u = \sum_{i=1}^l u_i$ with $u_i \in \mathcal{M}_k^i$ satisfying

$$(3.4) \quad \sum_{i=1}^l \|u_i\|_k^2 \leq c_0 \|u\|_k^2.$$

In applications, the above conditions are often trivial to verify. Moreover, under these hypotheses, we can prove the following theorem.

Theorem 3.1. *Let R_k be defined by (3.2) and assume that (3.3) and (3.4) are satisfied. Let $\theta \in (0, 2)$ and set $\gamma = \theta/c_1$. Then (2.12) holds, and (2.11) holds with $C_R = c_0 c_1 / [\theta(2 - \theta)]$.*

Before proving the theorem, we prove the following lemma [8].

Lemma 3.1. *Let n_0 be defined as above and $u_i, v_i \in \mathcal{M}_k^i$ for $i = 1, \dots, l$. Then*

$$(3.5) \quad \left(\sum_{i,m=1}^l |(A_k u_i, v_m)_k| \right)^2 \leq n_0^2 \sum_{i=1}^l (A_k u_i, u_i)_k \sum_{m=1}^l (A_k v_m, v_m)_k.$$

Proof. We note that

$$\begin{aligned} \left(\sum_{i,m=1}^l |(A_k u_i, v_m)_k| \right)^2 &= \left(\sum_{i,m=1}^l \kappa_{im} |(A_k u_i, v_m)_k| \right)^2 \\ &\leq \sum_{i,m=1}^l \kappa_{im} (A_k u_i, u_i)_k \sum_{i,m=1}^l \kappa_{im} (A_k v_m, v_m)_k \\ &\leq n_0^2 \sum_{i=1}^l (A_k u_i, u_i)_k \sum_{m=1}^l (A_k v_m, v_m)_k. \end{aligned}$$

This completes the proof of the lemma.

Proof of Theorem 3.1. We first show that

$$(3.6) \quad \frac{\|u\|_k^2}{\lambda_k} \leq c_0 \gamma^{-1} (R_k u, u)_k \quad \text{for all } u \in \mathcal{M}_k.$$

Let $u = \sum_{i=1}^l u_i$ be the decomposition of (3.4). Then

$$\begin{aligned} \|u\|_k^2 &= \sum_{i=1}^l (u_i, Q_k^i u)_k \leq \left(\sum_{i=1}^l \|u_i\|_k^2 \right)^{1/2} \left(\sum_{i=1}^l \|Q_k^i u\|_k^2 \right)^{1/2} \\ &\leq c_0^{1/2} \|u\|_k \left(\sum_{i=1}^l \|Q_k^i u\|_k^2 \right)^{1/2}. \end{aligned}$$

Hence,

$$(3.7) \quad \|u\|_k^2 \leq c_0 \sum_{i=1}^l \|Q_k^i u\|_k^2.$$

Now, it is immediate from the definitions that the largest eigenvalue of the operator $A_{k,i}$ is bounded by λ_k . Consequently,

$$(3.8) \quad \sum_{i=1}^l \|Q_k^i u\|_k^2 \leq \lambda_k \sum_{i=1}^l (A_{k,i}^{-1} Q_k^i u, u)_k = \lambda_k \gamma^{-1} (R_k u, u)_k.$$

Combining (3.7) and (3.8) proves (3.6).

We will show that the spectral radius of $T_k = R_k A_k$ is less than or equal to θ provided that we take $\gamma = \theta/c_1$. Let us temporarily assume this. Note, that R_k is a symmetric operator in the $(\cdot, \cdot)_k$ inner product. By (3.6), it is also positive definite, and hence its square root is well defined. We then have for $u \in \mathcal{M}_k$,

$$(3.9) \quad \begin{aligned} (\bar{R}_k u, u)_k &= ((2R_k - R_k A_k R_k)u, u)_k \\ &= ((2I - R_k^{1/2} A_k R_k^{1/2})R_k^{1/2} u, R_k^{1/2} u)_k \\ &\geq (2 - \theta)(R_k u, u)_k. \end{aligned}$$

The theorem follows combining (3.6) and (3.9), once we provide the desired estimate for the spectral radius of $R_k A_k$.

Let $v_i = P_k^i u$. By the definition of R_k and the identity $Q_k^i A_k = A_{k,i} P_k^i$,

$$(3.10) \quad \begin{aligned} (A_k R_k A_k u, u)_k &= \gamma \sum_{i=1}^l (A_k v_i, v_i)_k = \gamma \left(A_k \sum_{i=1}^l v_i, u \right)_k \\ &\leq \gamma \left(\left(A_k \sum_{i=1}^l v_i, \sum_{i=1}^l v_i \right)_k \right)^{1/2} ((A_k u, u)_k)^{1/2}. \end{aligned}$$

Applying Lemma 3.1 proves the desired bound, i.e.,

$$(3.11) \quad (A_k R_k A_k u, u)_k \leq \theta (A_k u, u)_k.$$

This completes the proof of the theorem.

Remark 3.1. The overlapping domain decomposition techniques (e.g. (3.2)) can be used directly to develop preconditioners for the operator A_k (see [8]). However, in this case the subspaces must be chosen in a much more restricted way. To prove that the additive preconditioner (3.2) provides a good preconditioner, one replaces (3.4) by the existence of a decomposition satisfying

$$(3.12) \quad \sum_{i=1}^l (A_k u_i, u_i)_k \leq C_0 (A_k u, u)_k.$$

As we shall see from later examples, it is much easier to construct subspaces satisfying (3.4). In general, the subspaces used for developing smoothers will not satisfy (3.12).

Remark 3.2. An obvious alternative to hypothesis (3.3) in the case of Theorem 3.1 is the assumption that

$$(3.13) \quad \sum_{i=1}^l (A_k P_k^i u, P_k^i u)_k \leq c_1 (A_k u, u)_k \quad \text{for all } u \in \mathcal{M}_k.$$

With such an assumption, (3.11) follows immediately from the first equality of (3.10) and provides a simpler proof. However, the limited interaction condition was introduced because it is also used for the analysis of the multiplicative algorithms to be subsequently described.

Remark 3.3. When developing preconditioners (instead of smoothers), it is often important to include a “coarse” subspace \mathcal{M}_k^0 which interacts with all of the other subspaces, i.e., $\kappa_{0k} \neq 0$ for all k . This is not necessary in the case of smoothers. However, it would still be possible to analyze the resulting algorithm using the above arguments and those presented in [8].

We define the multiplicative smoother based on the above subspace decomposition of \mathcal{M}_k in the following algorithm.

Algorithm 3.1. Let $f \in \mathcal{M}_k$. We define $R_k f \in \mathcal{M}_k$ as follows:

- (1) Set $v_0 = 0$.
- (2) Define v_i for $i = 1, \dots, l$ by

$$(3.14) \quad v_i = v_{i-1} + A_{k,i}^{-1} Q_k^i (f - A_k v_{i-1}).$$

- (3) Set $R_k f = v_l$.

It immediately follows from the identity $A_{k,i} P_k^i = Q_k^i A_k$ that

$$(3.15) \quad K_k = (I - P_k^l) \cdots (I - P_k^1).$$

That is, the error propagator associated with the smoother defined by Algorithm 3.1 is a product of orthogonal projections onto the complements of the subspaces. The next theorem provides an estimate for (2.11) and (2.12) with this definition of R_k .

Theorem 3.2. Let R_k be defined by Algorithm 3.1 and assume that (3.3) and (3.4) hold. Then (2.11) holds with

$$(3.16) \quad C_R = (2c_0(1 + c_1^2)).$$

In addition, (2.12) holds with $\theta = 2c_1/(c_1 + 1)$.

Proof. The proof of this theorem uses techniques of [8]. First, we define the operator

$$(3.17) \quad E_i = (I - P_k^i)(I - P_k^{i-1}) \cdots (I - P_k^1)$$

for $i = 1, \dots, l$. For convenience, we let $E_0 = I$ and note that $E_l = K_k$.

We will prove inequality (2.11) for $\bar{R}_k = (I - K_k^* K_k) A_k^{-1}$ by proving the equivalent inequality (2.13). Note that E_j^* is obtained by reversing the order of the factors in (3.17). With this observation, it is possible to use the same proof for the case of $\bar{R}_k = (I - K_k K_k^*) A_k^{-1}$.

We shall first derive some identities involving the above operators. We clearly have for $i = 1, \dots, l$,

$$(3.18) \quad E_{i-1} - E_i = P_k^i E_{i-1},$$

from which it follows that

$$(3.19) \quad I - E_i = \sum_{m=1}^i P_k^m E_{m-1}.$$

It is obvious from (3.18) that for $v \in \mathcal{M}_k$,

$$(3.20) \quad (A_k E_{i-1} v, E_{i-1} v)_k - (A_k E_i v, E_i v)_k = (A_k P_k^i E_{i-1} v, P_k^i E_{i-1} v)_k.$$

Summing (3.20) gives that

$$(3.21) \quad (A_k v, v)_k - (A_k E_l v, E_l v)_k = \sum_{i=1}^l (A_k P_k^i E_{i-1} v, E_{i-1} v)_k.$$

We note that (2.13) can be rewritten as

$$(3.22) \quad \omega \lambda_k^{-1} \|A_k v\|_k^2 \leq (A_k v, v)_k - (A_k E_l v, E_l v)_k.$$

But, by (3.7),

$$\begin{aligned} \|A_k v\|_k^2 &\leq c_0 \sum_{i=1}^l \|Q_k^i A_k v\|_k^2 \\ &= c_0 \sum_{i=1}^l (A_{k,i} P_k^i v, A_{k,i} P_k^i v)_k \leq c_0 \lambda_k \sum_{i=1}^l (A_k P_k^i v, P_k^i v)_k. \end{aligned}$$

Hence, (3.22) will follow if we can show that

$$(3.23) \quad \sum_{i=1}^l (A_k P_k^i v, P_k^i v)_k \leq 2(1 + c_1^2) \sum_{i=1}^l (A_k P_k^i E_{i-1} v, E_{i-1} v)_k.$$

It is shown in [8] that (3.23) holds under assumption (3.3) (cf. inequality (2.23) of [8]).

We include the proof of (3.23) for completeness. By (3.19),

$$(3.24) \quad (A_k P_k^i v, v)_k = (A_k P_k^i v, E_{i-1} v)_k + \sum_{m=1}^{i-1} (A_k P_k^i v, P_k^m E_{m-1} v)_k.$$

Summing gives

$$\sum_{i=1}^l (A_k P_k^i v, P_k^i v)_k = \sum_{i=1}^l (A_k P_k^i v, P_k^i E_{i-1} v)_k + \sum_{i=1}^l \sum_{m=1}^{i-1} (A_k P_k^i v, P_k^m E_{m-1} v)_k.$$

Thus, by the arithmetic-geometric mean inequality and Lemma 3.1,

$$\begin{aligned} \left(\sum_{i=1}^l (A_k P_k^i v, P_k^i v)_k \right)^2 &\leq 2 \left\{ \sum_{i=1}^l (A_k P_k^i v, P_k^i v)_k \sum_{i=1}^l (A_k P_k^i E_{i-1} v, E_{i-1} v)_k \right. \\ &\quad \left. + \left[\sum_{i,m=1}^l |(A_k P_k^i v, P_k^m E_{m-1} v)_k| \right]^2 \right\} \\ &\leq 2(1 + n_0^2) \sum_{i=1}^l (A_k P_k^i v, P_k^i v)_k \sum_{i=1}^l (A_k P_k^i E_{i-1} v, E_{i-1} v)_k. \end{aligned}$$

This completes the proof of (3.23).

Finally, we provide the estimate for θ . Note that for $u \in \mathcal{M}_k$, by Lemma 3.1,

$$\begin{aligned} (A_k T_k u, T_k u)_k &= (A_k(I - E_l)u, (I - E_l)u)_k = \sum_{i,m=1}^l (A_k P_k^i E_{i-1} u, P_k^m E_{m-1} u)_k \\ &\leq n_0 \sum_{i=1}^l (A_k P_k^i E_{i-1} u, E_{i-1} u)_k. \end{aligned}$$

Applying (3.21) gives

$$\begin{aligned} (3.25) \quad (A_k T_k u, T_k u)_k &\leq n_0[(A_k u, u)_k - (A_k E_l u, E_l u)_k] \\ &= n_0[2(A_k T_k u, u)_k - (A_k T_k u, T_k u)_k]. \end{aligned}$$

This shows that (2.12) holds for $\theta \leq 2n_0/(n_0 + 1)$ and hence completes the proof of the theorem.

Remark 3.4. We note that Theorem 3.1 provides the estimate $C_R = c_0 c_1$ for (2.11) when $\theta = 1$ and the smoother is defined by (3.2). In contrast, Theorem 3.2 provides the estimate $C_R = 2c_0(1 + c_1^2)$ when Algorithm 3.1 is used. This suggests that the additive version may work better in practice. As far as we know, this is not the case. In all of the examples which we have considered, numerical evidence suggests that the multiplicative smoother always works slightly better than the additive smoother using the same subspaces.

4. CONVERGENCE ESTIMATES FOR MULTIGRID ALGORITHMS

In this section, we apply the results of the theorems of the previous section to get convergence for multigrid Algorithm 2.2. We make no attempt to survey all possible applications but, instead, provide the theorems to illustrate the type of convergence results available utilizing the estimates on the smoothing operators provided by Theorems 3.1 and 3.2. Modifying a proof given in [9], we also provide a “no-regularity” convergence estimate in the case of the product smoothing operator defined by Algorithm 3.1.

As observed earlier, the multigrid process gives rise to the iterative reduction matrix $I - B_j A_j$, where B_j is given by Algorithm 2.2. Thus, bounds for the iterative convergence rate of either (2.6) or (2.7) follow from norm estimates for the operator $I - B_j A_j$. Alternatively, one can use the operator B_j directly in a preconditioned iteration for the solution of (2.1). Since B_j is symmetric in the inner product $(\cdot, \cdot)_j$ (cf. [5]), bounds for preconditioned iterative schemes follow from estimates for the condition number $K(B_j A_j)$, which is defined to be the ratio of the largest eigenvalue of $B_j A_j$ to the smallest.

We start by illustrating the convergence and preconditioning results for Algorithm 2.2 under the following regularity and approximation hypothesis: There exists a fixed number $\alpha \in (0, 1]$ and a positive constant C_α which does not depend on A_k such that for $k = 2, \dots, j$

$$(4.1) \quad |(A_k(I - I_k P_{k-1})u, u)_k| \leq C_\alpha^2 \left(\frac{\|A_k u\|_k^2}{\lambda_k} \right)^\alpha (A_k u, u)_k^{1-\alpha} \quad \text{for all } u \in \mathcal{M}_k.$$

Here $P_{k-1}: \mathcal{M}_k \mapsto \mathcal{M}_{k-1}$ is defined by $P_{k-1}v = w$, where w is the unique function in \mathcal{M}_{k-1} satisfying

$$(A_{k-1}w, \phi)_{k-1} = (A_k v, I_k \phi)_k \quad \text{for all } \phi \in \mathcal{M}_{k-1}.$$

The following two theorems are a consequence of Theorems 3.1 and 3.2 and the results in [5]. The first gives estimates for the reduction operator $I - B_j A_j$ in the norm $\|\cdot\|_j = ((A_j \cdot, \cdot)_j)^{1/2}$. The second gives estimates for the condition number $K(B_j A_j)$.

Theorem 4.1. *Let R_k be defined by either (3.2) or Algorithm 3.1 and assume that (4.1), (3.3), and (3.4) hold. Furthermore, assume that*

$$(A_k I_k v, I_k v)_k \leq (A_{k-1} v, v)_{k-1} \quad \text{for all } v \in \mathcal{M}_{k-1}.$$

Let B_j be defined by Algorithm 2.2 with $p = 1$ and $m(k) = m$ for all k . Then

$$(4.2) \quad \|(I - B_j A_j)v\|_j \leq \delta \|v\|_j \quad \text{for all } v \in \mathcal{M}_j,$$

where

$$\delta = \frac{M_\alpha j^{(1-\alpha)/\alpha}}{M_\alpha j^{(1-\alpha)/\alpha} + m^\alpha}.$$

If, instead, $m(k)$ satisfies

$$(4.3) \quad \beta_0 m(k) \leq m(k-1) \leq \beta_1 m(k)$$

(β_0 and β_1 are constants which are greater than one and independent of k), then (4.2) holds with

$$\delta = \frac{M_\alpha}{M_\alpha + m(j)^\alpha}.$$

The constant M_α above is independent of j .

Theorem 4.2. *Let R_k be defined by either (3.2) or Algorithm 3.1 and assume that (4.1), (3.3), and (3.4) hold. Let B_j be defined by Algorithm 2.2 with $p = 1$ and $\{m(k)\}$ satisfying (4.3). Then $K(B_j A_j) \leq \eta_1/\eta_0$, where η_0 and η_1 are given by*

$$\eta_0 = \frac{m(j)^\alpha}{M_\alpha + m(j)^\alpha}$$

and

$$\eta_1 = \frac{M_\alpha + m(j)^\alpha}{m(j)^\alpha},$$

i.e., the system $B_j A_j$ is well conditioned independently of j .

Multigrid is often applied to sequences of operators approximating the solution of an elliptic partial differential equation. In this case, the validity of (4.1) is inherently related to the regularity properties of solutions of this partial differential equation. Alternative hypotheses which avoid these regularity assumptions have been used to provide multigrid results (see [7, 8, 9]). These are as follows:

- (1) The subspaces $\mathcal{M}_1, \dots, \mathcal{M}_j$ are nested and the operators are inherited, i.e., $\mathcal{M}_{k-1} \subseteq \mathcal{M}_k$ and

$$(4.4) \quad (A_k v, v)_k = (A_j v, v)_j \quad \text{for all } v \in \mathcal{M}_k.$$

- (2) There exists a sequence of linear operators $Q_k: \mathcal{M}_j \mapsto \mathcal{M}_k$ for $k = 1, \dots, j$, with $Q_j = I$ satisfying the following properties. There are constants C_1 and C_2 not depending on k for which

$$(4.5) \quad \begin{aligned} \|(Q_k - Q_{k-1})u\|_k^2 &\leq C_1 \lambda_k^{-1} A(u, u) \quad \text{for } k = 2, \dots, j, \\ A(Q_k u, Q_k u) &\leq C_2 A(u, u) \quad \text{for } k = 1, \dots, j - 1. \end{aligned}$$

The inequalities in (4.5) hold for all $u \in \mathcal{M}_j$. Inequalities of the form of (4.5) can be verified without the use of elliptic regularity estimates (see [7]).

The hypotheses required for the smoother in the case of the “regularity-free” estimates are less stringent. Loosely, the smoother R_k need only “smooth” on a subspace of \mathcal{M}_k containing the image of $Q_k - Q_{k-1}$. To this end, let $\widetilde{\mathcal{M}}_k$ be a subspace of \mathcal{M}_k which contains the range of the operator $Q_k - Q_{k-1}$. Assume that we are given a decomposition

$$(4.6) \quad \widetilde{\mathcal{M}}_k = \sum_{i=1}^l \mathcal{M}_k^i$$

satisfying assumptions (3.3) and (3.4) (with $\widetilde{\mathcal{M}}_k$ replacing \mathcal{M}_k). Let $\widetilde{R}_k: \widetilde{\mathcal{M}}_k \mapsto \widetilde{\mathcal{M}}_k$ be defined by either (3.2) (with $\gamma = 1/c_1$) or Algorithm 3.1 using these spaces. In addition, set $R_k = \widetilde{R}_k \widetilde{P}_k^0: \mathcal{M}_k \mapsto \widetilde{\mathcal{M}}_k$, where \widetilde{P}_k^0 denotes the $(\cdot, \cdot)_k$ orthogonal projection onto $\widetilde{\mathcal{M}}_k$. Note that Theorems 3.1 and 3.2 provide estimates for a constant C_R satisfying

$$(4.7) \quad \frac{\|u\|_k^2}{\lambda_k} \leq C_R (\overline{R}_k u, u)_k \quad \text{for all } u \in \widetilde{\mathcal{M}}_k.$$

Here,

$$(4.8) \quad \overline{R}_k = \widetilde{R}_k + \widetilde{R}_k^t - \widetilde{R}_k^t \widetilde{A}_k \widetilde{R}_k,$$

where $\widetilde{A}_k: \widetilde{\mathcal{M}}_k \mapsto \widetilde{\mathcal{M}}_k$ is defined by

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

The following theorem provides estimates for the rate of convergence of the multigrid algorithm with this R_k under the above assumptions. The proof in the case of (3.2) follows directly from results in [9] and Theorem 3.1. The proof of the theorem in the case of Algorithm 3.1 is a modification to that given in [9]. A somewhat more restricted result in the case of nonsymmetric R_k (also based on [9]) was given in [19].

Theorem 4.3. *Assume that (4.4) and (4.5) hold. Let R_k be defined as above and assume that (3.3) and (3.4) hold with $\widetilde{\mathcal{M}}_k$ replacing \mathcal{M}_k . Let B_j be defined by either Algorithm 2.2 or the nonsymmetric smoothing (with corresponding operator denoted by B_j^n) version (see [9]). Then (4.2) holds with $\delta = \delta_j^2$, where*

$$(4.9) \quad \delta_j = 1 - \frac{1}{C(j-1)}$$

and $C = [(1 + C_2^{1/2})(2c_1)^{1/2} + (C_R C_1)^{1/2}]^2$. The constant C_R satisfies (4.7) and is provided by either Theorem 3.1 or Theorem 3.2. In the case of the nonsymmetric smoothing version,

$$(4.10) \quad \|(I - B_j^n A_j)v\|_j \leq \delta_j \|v\|_j \quad \text{for all } v \in \mathcal{M}_j.$$

Proof. For the purpose of this proof, we shall let $A(\cdot, \cdot) = (A_j \cdot, \cdot)_j$. We need only prove the result in the case of nonsymmetric R_k . Moreover, we shall prove the result for $p = 1$. The results for higher p follow from arguments given in [9].

First of all, it was observed in [9] that under the above assumptions,

$$(I - B_j A_j) = (I - B_j^n A_j)^*(I - B_j^n A_j),$$

where B_j^n denotes the multigrid operator which involves smoothing only before correction. Set

$$\bar{K}_k^{(m(k))} = \begin{cases} (K_k^* K_k)^{m(k)/2} & \text{if } m(k) \text{ is even,} \\ (K_k^* K_k)^{(m(k)-1)/2} K_k^* & \text{if } m(k) \text{ is odd.} \end{cases}$$

It was also observed in [9] that for $T_k = (I - (\bar{K}_k^{(m(k))})^*)P_k$,

$$(4.11) \quad (I - B_j^n A_j)^* = (I - T_j)(I - T_{j-1}) \cdots (I - T_1).$$

We use a product analysis similar to that used in Theorem 3.2 and also Theorem 1 of [9]. To this end, we set $E_0 = I$ and

$$E_k = (I - T_k)(I - T_{k-1}) \cdots (I - T_1) = (I - T_k)E_{k-1}.$$

As in the proof of Theorem 3.2 (compare with (3.21)),

$$(4.12) \quad \begin{aligned} A(u, u) - A(E_j u, E_j u) &= \sum_{k=1}^j [A(E_{k-1} u, E_{k-1} u) - A(E_k u, E_k u)] \\ &= \sum_{k=1}^j A((2I - T_k)E_{k-1} u, T_k E_{k-1} u). \end{aligned}$$

Note that $I - B_j^n A_j = E_j^*$, and hence inequalities (4.2) and (4.10) will follow if we can show that

$$(4.13) \quad \begin{aligned} A(u, u) &\leq C(j-1)[A(u, u) - A(E_j u, E_j u)] \\ &= C(j-1) \sum_{k=1}^j A((2I - T_k)E_{k-1} u, T_k E_{k-1} u). \end{aligned}$$

Proceeding as in [9], we use the fact that $Q_j = I$ and write

$$u = \sum_{k=2}^j (Q_k - Q_{k-1})u + Q_1 u.$$

Thus,

$$\begin{aligned}
 A(u, u) &= \sum_{k=2}^j A(u, (Q_k - Q_{k-1})u) + A(u, Q_1 u) \\
 (4.14) \quad &= \sum_{k=2}^j A(E_{k-1}u, (Q_k - Q_{k-1})u) + A(u, Q_1 u) \\
 &\quad + \sum_{k=2}^j A((I - E_{k-1})u, (Q_k - Q_{k-1})u).
 \end{aligned}$$

For the the first sum on the right-hand side of (4.14), we see that

$$\begin{aligned}
 \sum_{k=2}^j A(E_{k-1}u, (Q_k - Q_{k-1})u) &= \sum_{k=2}^j A(P_k E_{k-1}u, (Q_k - Q_{k-1})u) \\
 &= \sum_{k=2}^j (\tilde{P}_k^0 A_k P_k E_{k-1}u, (Q_k - Q_{k-1})u)_k \\
 &\leq \sum_{k=2}^j \|\tilde{P}_k^0 A_k P_k E_{k-1}u\|_k \|(Q_k - Q_{k-1})u\|_k.
 \end{aligned}$$

Applying (4.5) gives

$$\sum_{k=2}^j A(E_{k-1}u, (Q_k - Q_{k-1})u) \leq (C_1)^{1/2} A^{1/2}(u, u) \sum_{k=2}^j \lambda_k^{-1/2} \|\tilde{P}_k^0 A_k P_k E_{k-1}u\|_k.$$

For \bar{R}_k defined by (4.8), Theorem 3.2 gives

$$\begin{aligned}
 &\sum_{k=2}^j A(E_{k-1}u, (Q_k - Q_{k-1})u) \\
 &\leq (C_R C_1 (j-1))^{1/2} A^{1/2}(u, u) \left(\sum_{k=2}^j (\bar{R}_k \tilde{P}_k^0 A_k P_k E_{k-1}u, \tilde{P}_k^0 A_k P_k E_{k-1}u) \right)^{1/2}.
 \end{aligned}$$

It is easy to check that for $v \in \tilde{\mathcal{M}}_k$,

$$\begin{aligned}
 (4.15) \quad \bar{R}_k v &= \tilde{R}_k v + \tilde{R}_k^t v - \tilde{R}_k^t \tilde{A}_k \tilde{R}_k v \\
 &= R_k v + R_k^t v - R_k^t A_k R_k v = (I - K_k^* K_k) A_k^{-1} v.
 \end{aligned}$$

The last equality in (4.15) defines an extension of \bar{R}_k to \mathcal{M}_k . This extension, which we shall still denote by \bar{R}_k , is symmetric with respect to $(\cdot, \cdot)_k$. Moreover, R_k^t is defined by cycling through Algorithm 3.1 in reverse order, and hence its image is contained in $\tilde{\mathcal{M}}_k$. Thus, it follows that $\bar{R}_k \tilde{P}_k^0 = \bar{R}_k$, and hence

$$\begin{aligned}
 \sum_{k=2}^j A(E_{k-1}u, (Q_k - Q_{k-1})u) &\leq (C_R C_1 (j-1))^{1/2} A^{1/2}(u, u) \\
 &\quad \cdot \left(\sum_{k=2}^j A((I - K_k^* K_k) P_k E_{k-1}u, P_k E_{k-1}u) \right)^{1/2}.
 \end{aligned}$$

Let $w = P_k E_{k-1} u$. The spectrum of $K_k^* K_k$ is in $[0, 1]$, and hence

$$\begin{aligned} A((I - K_k^* K_k)w, w) &\leq A((I - \bar{K}_k^{(m(k))})(\bar{K}_k^{(m(k))})^*)w, w) \\ &= A((2I - T_k)w, T_k w) = A((2I - T_k)E_{k-1}u, T_k E_{k-1}u). \end{aligned}$$

Thus,

$$(4.16) \quad \sum_{k=2}^j A(E_{k-1}u, (Q_k - Q_{k-1})u) \leq (C_R C_1 (j-1))^{1/2} A^{1/2}(u, u) \cdot \left(\sum_{k=2}^j A((2I - T_k)E_{k-1}u, T_k E_{k-1}u) \right)^{1/2}.$$

For the remaining terms in (4.14), we have

$$(4.17) \quad \begin{aligned} &\sum_{k=2}^j A((I - E_{k-1})u, (Q_k - Q_{k-1})u) + A(u, Q_1 u) \\ &= \sum_{k=2}^{j-1} A((E_k - E_{k-1})u, Q_k u) + A((I - E_{j-1})u, u). \end{aligned}$$

But, $E_{k-1} - E_k = T_k E_{k-1}$, and it follows that

$$I - E_{j-1} = \sum_{k=1}^{j-1} T_k E_{k-1}.$$

From this and (4.17),

$$(4.18) \quad \begin{aligned} &\sum_{k=2}^j A((I - E_{k-1})u, (Q_k - Q_{k-1})u) + A(u, Q_1 u) \\ &= \sum_{k=2}^{j-1} A(T_k E_{k-1}u, (I - Q_k)u) + A(T_1 u, u) \\ &\leq (j-1)^{1/2} (1 + C_2^{1/2}) \left(\sum_{k=1}^{j-1} A(T_k E_{k-1}u, T_k E_{k-1}u) \right)^{1/2} A^{1/2}(u, u). \end{aligned}$$

We shall show that

$$(4.19) \quad A(T_k w, T_k w) \leq 2c_1 A((2I - T_k)w, T_k w).$$

If $m(k)$ is even, then this is evidently true, since $c_1 \geq 1$ and T_k is symmetric in $A(\cdot, \cdot)$ with spectrum in $[0, 1]$. For $m(k)$ odd, set $\tilde{K}_k = (K_k K_k^*)^{(m(k)-1)/2}$. Then \tilde{K}_k is symmetric in $A(\cdot, \cdot)$ with spectrum in $[0, 1]$ and $T_k = (I - \tilde{K}_k K_k)P_k$. Clearly, it suffices to prove (4.19) for $w \in \mathcal{M}_k$. Now

$$\begin{aligned} A(T_k w, T_k w) &= A((I - \tilde{K}_k K_k)w, (I - \tilde{K}_k K_k)w) \\ &\leq 2 \left[A((I - K_k)w, (I - K_k)w) + A((I - \tilde{K}_k)K_k w, (I - \tilde{K}_k)K_k w) \right]. \end{aligned}$$

By Theorem 3.2,

$$A((I - K_k)w, (I - K_k)w) \leq \frac{2c_1}{c_1 + 1} A((I - K_k)w, w).$$

This can be rewritten (see (3.25)) as

$$A((I - K_k)w, (I - K_k)w) \leq c_1 [A(w, w) - A(K_k w, K_k w)].$$

Using the symmetry of \tilde{K}_k and the fact that its spectrum is in $[0, 1]$ gives

$$A((I - \tilde{K}_k)K_k w, (I - \tilde{K}_k)K_k w) \leq A(K_k w, K_k w) - A(\tilde{K}_k K_k w, \tilde{K}_k K_k w).$$

Combining the above and noting that $c_1 \geq 1$ gives

$$\begin{aligned} A(T_k w, T_k w) &\leq 2c_1 [A(w, w) - A(\tilde{K}_k K_k w, \tilde{K}_k K_k w)] \\ &= 2c_1 A((2I - T_k)w, T_k w), \end{aligned}$$

i.e., (4.19) holds.

Combining (4.14), (4.16), (4.18), and (4.19) gives

$$A(u, u) \leq C(j - 1) \left(\sum_{k=1}^j A((2I - T_k)E_{k-1}u, T_k E_{k-1}u) \right),$$

for $C = [(1 + C_2^{1/2})(2c_1)^{1/2} + (C_R C_1)^{1/2}]^2$. This completes the proof of the theorem.

Remark 4.1. The requirement that inequality (4.7) need only hold on $\tilde{\mathcal{M}}_k$ is important in local refinement applications. These are discussed in more detail in [9]. However, to apply Theorems 3.1 and 3.2, one need only provide a decomposition of the subspace $\tilde{\mathcal{M}}_k$. The resulting smoothing operator R_k only involves computation in the subdomain where the new nodes are being added.

5. IMPLEMENTATION OF THE SMOOTHING PROCEDURE

In this section, we consider implementation of the smoothing procedures described in §3. We shall see that the additive schemes correspond to generalizations of block Jacobi iteration. The product schemes correspond to generalizations of block Gauss-Seidel iteration. The observations that, e.g., Gauss-Seidel iteration is a product scheme of the form of Algorithm 3.1 are not new (cf., for example, [17]). We include this section only to stress the point that the results provided earlier apply to the smoothers commonly used in multigrid algorithms.

We first consider computer implementation of the parts of Algorithm 2.2 which are relevant to the smoothing procedures. Assume that a decomposition of \mathcal{M}_k of the form of (3.1) is given which satisfies (3.3) and (3.4). Moreover, assume that there is a basis $\{\phi_k^i\}$ for \mathcal{M}_k such that each \mathcal{M}_k^i has a basis consisting of a subset of $\{\phi_k^i\}$. Let M denote the stiffness matrix associated with this basis, i.e., $M_{im} = (A_k \phi_k^i, \phi_k^m)_k$. In implementation, one seldom is required to solve (2.1) but rather the equivalent matrix equation

$$MU = F,$$

where U is related to the solution of (2.1) by $u = \sum_i U_i \phi_j^i$ and F is a known vector of coefficients ($F_i = (f, \phi_j^i)_j$). Consequently, in the multigrid implementation, we are required to compute the action of R_k on a function $g \in \mathcal{M}_k$ which is represented by the inner product vector $G_i = (g, \phi_k^i)_k$.

We first consider the case of the additive smoothing operator defined by (3.2). Let S_i denote the indices of $\{\phi_k^m\}$ which correspond to the basis functions of \mathcal{M}_k^i . We note that the vector W^i representing the function $w^i = A_{k,i}^{-1} Q_k^i g$ satisfies the equation

$$M^i W^i = G^i,$$

where

$$M_{lm}^i = \begin{cases} M_{lm} & \text{if } l, m \in S_i, \\ 0 & \text{if } l \neq m \text{ and either } l \notin S_i \text{ or } m \notin S_i, \\ 1 & \text{if } l = m \text{ and } l \notin S_i \end{cases}$$

and

$$G_m^i = \begin{cases} G_m & \text{if } m \in S_i, \\ 0 & \text{otherwise.} \end{cases}$$

We now consider the case when all of the subspaces are disjoint. Then, we may partition the basis elements into groups corresponding to the subspaces. Under this ordering, the vector $W = \{W_i\}$ representing the function $w = \sum A_{k,i}^{-1} Q_k^i g$ satisfies the equation

$$\bar{M}_a W = G,$$

where M_a is the block diagonal part of the matrix M . In the case when each subspace has one degree of freedom associated with a given basis function, then W is given by

$$W_i = (A_k \phi_k^i, \phi_k^i)^{-1} G_i,$$

i.e., the smoother corresponds to the Jacobi method applied to the diagonally scaled stiffness matrix.

We next consider the case of the multiplicative smoother. Again, we look at the case when all of the subspaces are disjoint and the basis elements are ordered into groups accordingly. The matrix M has a block structure corresponding to this ordering with blocks denoted by $M^{i,m}$, $i, m = 1, \dots, l$. As usual, we write

$$M = L + D + U,$$

where L , D , and U are respectively, block lower diagonal, block diagonal, and block upper diagonal. Let F^i be the vector of data corresponding to these blocks, i.e., $F^i = ((f, \phi_{m_1}^k)_k, \dots, (f, \phi_{m_2}^k)_k)^t$, where m_1 and m_2 denote the first and last basis element corresponding to the subspace \mathcal{M}_k^i . Let the vectors V_i be the vectors of coefficients representing the functions v_i appearing in Algorithm 3.1. These are partitioned in a similar manner. We first note that the i th step in (3.14) only changes the i th component. Thus, the i th component of V_l is defined by the equation

$$M_{i,i} V_l^i = F_i - \sum_{m < i} M_{i,m} V_{i-1}^m = F_i - \sum_{m < i} M_{i,m} V_l^m - \sum_{m > i} M_{i,m} V_0^m.$$

This can be rewritten as

$$(L + D)V_l = -UV_0 + F,$$

and corresponds to block Gauss-Seidel iteration applied to the stiffness matrix M .

6. TYPICAL FINITE ELEMENT APPLICATIONS

In this section, we discuss developing smoothers using the techniques of §3 for finite element multigrid applied to a second-order elliptic boundary value problem. First, we consider the case when the subspace is defined in terms of a quasi-uniform triangulation which approximates the original domain. This often leads to spaces $\{\mathcal{M}_k\}$ which are not nested. The case of mesh refinement is discussed next. We will see that it is easy to apply the techniques presented earlier to develop smoothers which only require computation where new nodes are being added and give rise to effective multigrid algorithms.

We shall consider the problem of approximating the solution U of

$$(6.1) \quad \begin{aligned} LU &= F \quad \text{in } \Omega, \\ U &= 0 \quad \text{on } \partial\Omega. \end{aligned}$$

Here Ω is a domain (not necessarily polygonal) 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 $\bar{\Omega}$. The form A corresponding to the above operator is given by

$$(6.2) \quad A(v, w) = \sum_{i,j=1}^n \int_{\Omega} a_{ij} \frac{\partial v}{\partial x_i} \frac{\partial w}{\partial x_j} dx.$$

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, $U \in H_0^1(\Omega)$ 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 Ω .

We will first discuss the case of quasi-uniform triangulation. We assume that Ω has been approximately triangulated with a sequence of quasi-uniform triangulations $\Omega_k = \bigcup_i \tau_k^i$ of size h_k for $k = 1, \dots, j$, where the quasi-uniformity constants are independent of k . 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 $\partial\Omega_k$. If Ω is polygonal, then it is possible to take $\Omega_k = \Omega$ and construct triangulations which are nested.

We next define the inner product $(\cdot, \cdot)_k$. We do not have complete freedom here, since we must choose an inner product so that either (4.1) or (4.5) are satisfied, depending on the application. Let $\{y_k^i\}$ be the collection of nodes corresponding to the triangulation for \mathcal{M}_k . It suffices to take

$$(6.3) \quad (u, v)_k = h_k^n \sum_i u(y_k^i) v(y_k^i).$$

Note that the quasi-uniformity of the triangulations implies that the norm $\|\cdot\|_k$ is equivalent to the $L^2(\Omega)$ norm on the subspace \mathcal{M}_k . The operators A_k , $k = 1, \dots, j$, are then defined by

$$(6.4) \quad (A_k v, \phi)_k = A(v, \phi) \quad \text{for all } \phi \in \mathcal{M}_k.$$

Finally, the operators I_k are defined by nodal interpolation, i.e., $I_k w$ is defined to be the unique function in \mathcal{M}_k which equals w at the nodes of \mathcal{M}_k .

Let $\{\phi_k^i\}_{i=1}^{N_k}$ denote the usual nodal basis associated with the subspace \mathcal{M}_k . Partition the integers $\{1, 2, \dots, N_k\}$ into sets S_1, S_2, \dots, S_l and define \mathcal{M}_k^i to be the span of the basis functions with indices in S_i . The discussion in the previous section shows that implementation of (3.2) and Algorithm 3.1 reduce to block Jacobi and block Gauss-Seidel iteration on the stiffness matrices. These subspaces provide a direct sum decomposition of the space \mathcal{M}_k and hence, the decomposition $u = \sum u_i$ with $u_i \in \mathcal{M}_k^i$ is uniquely defined. In addition, since the matrix with entries $N_{im} = (\phi_k^i, \phi_k^m)_k$ is diagonal,

$$\sum_{i=1}^l \|u_i\|_k^2 = \|u\|_k^2,$$

i.e., (3.4) holds with $c_0 = 1$.

The constant c_1 appearing in (3.3) is related to the geometry of the subspaces. For $i = 1, 2, \dots, l$, let Ω_k^i denote the union of the supports of the basis functions defining \mathcal{M}_k^i . Note that κ_{im} is nonzero only if $\Omega_k^i \cap \Omega_k^m \neq \emptyset$. Let χ_k^i be the number of subdomains $\{\Omega_k^m\}$ which intersect Ω_k^i . Then we can take c_1 in (3.3) to be the maximum of $\{\chi_k^i\}$ for $i = 1, 2, \dots, l$. In the case of point relaxation (i.e., $\mathcal{M}_k^i = \{c\phi_k^i\}$), c_1 can be taken to be one plus the maximum number of triangles which meet at a given vertex. Alternatively, for line relaxation, the grid consists of a regular rectangular mesh and the i th subspace is defined to be, for example, the span of the basis functions on the i th horizontal mesh line. In this case, $c_1 = 3$. Obviously, many other examples are possible.

We next consider the case when the mesh results from a local refinement. To illustrate this situation, we consider the case of two spatial dimensions. We note that for refinement applications, it is only possible to prove (4.1) with a C_α which grows with powers of the ratio of the diameters of largest to smallest triangle in the refined mesh. In contrast, estimates of the form (4.5) hold with constants independent of the mesh parameters. Consequently, we shall only consider the case of nested spaces and inherited operators.

We start with the definition of the nested refined grids. These grids are defined in terms of a given sequence of nested subdomains

$$\Omega_j \subseteq \Omega_{j-1} \subseteq \dots \subseteq \Omega_0 = \Omega.$$

We assume that we are given a coarse triangulation of $\Omega = \bigcup_m \tau_0^m$. This coarse triangulation provides the first grid $\{\tau_0^m\}$. Given that a grid $\{\tau_{k-1}^m\}$ has been defined, the grid $\{\tau_k^m\}$ is defined by refining those triangles of $\{\tau_{k-1}^m\}$ which are in Ω_k . This refinement is done, for example, by breaking each triangle of the mesh $\{\tau_{k-1}^m\}$ in Ω_k into four triangles by connecting the midpoints of the edges. We assume $\partial\Omega_k$ aligns with the mesh $\{\tau_{k-1}^m\}$.

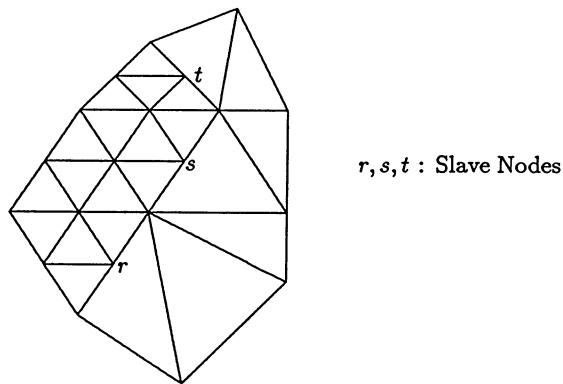


FIGURE 6.1
A mesh transition region

The space \mathcal{M}_k is defined to be the set of continuous functions on Ω which are piecewise linear with respect to the grid $\{\tau_k^m\}$ and vanish on $\partial\Omega$. We note that the continuity constraint implies that there are no new degrees of freedom corresponding to nodes on $\partial\Omega_k$ (see Figure 6.1). These new nodes on $\partial\Omega_k$ will be called slave nodes since, by continuity, their values are determined by the values of their neighboring nodes (which were already in the previous grid). It is easy to see that the space \mathcal{M}_k has a nodal basis consisting of the vertices of $\{\tau_k^m\}$ excluding the slave nodes.

For this application, we shall take $(\cdot, \cdot)_k$ to be the $L^2(\Omega)$ inner product. The operators $\{A_k\}$ are defined by (6.4) and the operators I_k are defined to be the natural injection of \mathcal{M}_{k-1} into \mathcal{M}_k .

A sequence of operators Q_k , $k = 1, \dots, l$, are constructed in [9] satisfying (4.5). These operators, in addition, satisfy $(Q_k - Q_{k-1})v \in \tilde{\mathcal{M}}_k$ for all $v \in \mathcal{M}_j$, where

$$(6.5) \quad \tilde{\mathcal{M}}_k = \{\phi \in \mathcal{M}_k \mid \text{supp } \phi \subseteq \Omega_k\}.$$

Now, to apply Theorem 4.3, we need only provide a decomposition of the subspace $\tilde{\mathcal{M}}_k$. Note that $\tilde{\mathcal{M}}_k$ is a finite element space corresponding to a quasi-uniform triangulation of Ω_k . Accordingly, the constructions given above can be used. Note that this leads to smoothing algorithms which only require computation involving the nodes of Ω_k and not on all of the nodes of the space \mathcal{M}_k .

BIBLIOGRAPHY

1. R. E. Bank and C. C. Douglas, *Sharp estimates for multigrid rates of convergence with general smoothing and acceleration*, SIAM J. Numer. Anal. **22** (1985), 617–633.
2. R. E. Bank and T. Dupont, *An optimal order process for solving finite element equations*, Math. Comp. **36** (1981), 35–51.
3. D. Braess and W. Hackbusch, *A new convergence proof for the multigrid method including the V-cycle*, SIAM J. Numer. Anal. **20** (1983), 967–975.
4. J. H. Bramble and J. E. Pasciak, *New convergence estimates for multigrid algorithms*, Math. Comp. **49** (1987), 311–329.
5. J. H. Bramble, J. E. Pasciak, and J. Xu, *The analysis of multigrid algorithms with nonnested spaces or noninherited quadratic forms*. Math. Comp. **56** (1991). 1–34.

6. ———, *The analysis of multigrid algorithms for nonsymmetric and indefinite elliptic problems*, Math. Comp. **51** (1988), 389–414.
7. ———, *Parallel multilevel preconditioners*, Math. Comp. **55** (1990), 1–22.
8. J. H. Bramble, J. E. Pasciak, J. Wang, and J. Xu, *Convergence estimates for product iterative methods with applications to domain decomposition*, Math. Comp. **56** (1991), 1–21.
9. ———, *Convergence estimates for multigrid algorithms without regularity assumptions*, Math. Comp. **56** (1991), 23–45.
10. A. Brandt, *Multi-level adaptive solutions to boundary-value problems*, Math. Comp. **31** (1977), 333–390.
11. W. Hackbusch, *Multi-grid methods and applications*, Springer-Verlag, New York, 1985.
12. J. Mandel, S. McCormick, and R. Bank, *Variational multigrid theory*, Multigrid Methods (S. McCormick, ed.), SIAM, Philadelphia, PA, 1987, pp. 131–178.
13. J. Mandel, S. F. McCormick, and J. Ruge, *An algebraic theory for multigrid methods for variational problems*, (Preprint).
14. S. F. McCormick, *Multigrid methods for variational problems: General theory for the V-cycle*, SIAM J. Numer. Anal. **22** (1985), 634–643.
15. ———, *Multigrid methods for variational problems: Further results*, SIAM J. Numer. Anal. **21** (1984), 255–263.
16. S. McCormick (Ed.), *Multigrid methods*, SIAM, Philadelphia, PA, 1987.
17. S. F. McCormick and J. Ruge, *Unigrid for multigrid simulation*, Math. Comp. **41** (1983), 43–62.
18. R. Verfürth, *A multilevel algorithm for mixed problems*, SIAM J. Numer. Anal. **21** (1984), 264–284.
19. J. Wang, *Convergence analysis without regularity assumptions for multigrid algorithms based on SOR smoothing*, (preprint).
20. O. Widlund, *Optimal iterative refinement methods*, Technical Report No. 391, Courant Institute of Mathematical Sciences, 1988.

DEPARTMENT OF MATHEMATICS, CORNELL UNIVERSITY, ITHACA, NEW YORK 14853
E-mail address: bramble@mssun7.msi.cornell.edu

DEPARTMENT OF APPLIED SCIENCE, BROOKHAVEN NATIONAL LABORATORY, UPTON, NEW YORK
11973
E-mail address: pasciak@bnl.gov