

CONVERGENCE ESTIMATES FOR PRODUCT ITERATIVE METHODS WITH APPLICATIONS TO DOMAIN DECOMPOSITION

JAMES H. BRAMBLE, JOSEPH E. PASCIAK, JUNPING WANG, AND JINCHAO XU

ABSTRACT. In this paper, we consider iterative methods for the solution of symmetric positive definite problems on a space \mathcal{V} which are defined in terms of products of operators defined with respect to a number of subspaces. The simplest algorithm of this sort has an error-reducing operator which is the product of orthogonal projections onto the complement of the subspaces. New norm-reduction estimates for these iterative techniques will be presented in an abstract setting. Applications are given for overlapping Schwarz algorithms with many subregions for finite element approximation of second-order elliptic problems.

1. INTRODUCTION

In this paper, we shall be concerned with solving problems in an abstract Hilbert space \mathcal{V} with inner product $A(\cdot, \cdot)$. Denote by \mathcal{V}' the dual of the Hilbert space \mathcal{V} with $\langle \chi, \cdot \rangle$ being the action of $\chi \in \mathcal{V}'$ on \mathcal{V} . Given a function $f \in \mathcal{V}'$, we seek the solution $u \in \mathcal{V}$ of the equation

$$(1.1) \quad A(u, \phi) = \langle f, \phi \rangle \quad \text{for all } \phi \in \mathcal{V}.$$

We will consider iterative methods for (1.1) based on a sequence of subspaces. To this end, let $\mathcal{V}_1, \dots, \mathcal{V}_J$ denote closed subspaces of \mathcal{V} and define $A_i: \mathcal{V}_i \rightarrow \mathcal{V}'_i$ by

$$\langle A_i v, \phi \rangle = A(v, \phi) \quad \text{for all } v, \phi \in \mathcal{V}_i.$$

Denote by A the corresponding operator defined on the whole space \mathcal{V} . Thus, the problem (1.1) is equivalent to the problem of finding $u \in \mathcal{V}$ such that

$$(1.2) \quad Au = f.$$

Received February 5, 1990.

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

Key words and phrases. Second-order elliptic equation, domain decomposition.

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, DMS88-05311-04 and by the U.S. Army Research Office through the Mathematical Science Institute, Cornell University. The third author's research was supported by Office of Naval Research Contract No. 0014-88-K-0370 and by the Institute for Scientific Computation of the University of Wyoming through National Science Foundation Grant No. RII-8610680.

©1991 American Mathematical Society
0025-5718/91 \$1.00 + \$.25 per page

We assume that linear operators $R_i : \mathcal{V}_i' \mapsto \mathcal{V}_i$ for $i = 1, \dots, J$ are given. In practice, the operator R_i will in some sense “approximately invert” A_i . We will consider iterative methods of the following type for solving (1.1).

Algorithm 1. Given $u^l \in \mathcal{V}$, an approximation to the solution u of (1.2), we define the next iterate $u^{l+1} \in \mathcal{V}$ as follows:

- (1) Set $Y_0 = u^l$.
- (2) For $i = 1, \dots, J$ define Y_i by

$$Y_i = Y_{i-1} + R_i Q_i (f - AY_{i-1}),$$

where Q_i denotes the projection onto the subspace \mathcal{V}_i' defined by

$$\langle \chi - Q_i \chi, \phi \rangle = 0 \quad \text{for all } \phi \in \mathcal{V}_i.$$

- (3) Set $u^{l+1} = Y_J$.

Let $e_0 = u - u^l$ and $e_i = u - Y_i$ for $i = 1, \dots, J$. Let P_i denote the orthogonal projection into the subspace \mathcal{V}_i , i.e., $P_i v = w$ where w is the unique function in \mathcal{V}_i satisfying

$$A(w, \phi) = A(v, \phi) \quad \text{for all } \phi \in \mathcal{V}_i.$$

A simple computation shows that $Q_i A = A_i P_i$ and hence $e_i = (I - R_i A_i P_i) e_{i-1}$. Consequently,

$$(1.3) \quad u - u^{l+1} = (I - T_J)(I - T_{J-1}) \cdots (I - T_1)(u - u^l),$$

where $T_i = R_i A_i P_i$. Thus, estimates for the rate of convergence for the iterative method follow directly from norm bounds on the product in (1.3). The purpose of this paper is to provide new techniques for estimating norms of products of operators of the form appearing on the right-hand side of (1.3).

One natural example of a choice of R_i is $R_i = A_i^{-1}$. In this case the above product reduces to a product of orthogonal projections onto the complements of the subspaces. Note, however, that the action of the inverse of A_i must be computed as part of the iterative procedure. For this reason, it is often more efficient to use various preconditioners for A_i . This will be clearly illustrated in the applications to be presented.

The most important application of the results of this paper is to the computation of the solutions of the discrete equations which result from the numerical approximation of elliptic boundary value problems. In the case of finite element approximation, the inner product $A(\cdot, \cdot)$ is the form corresponding to the differential operator and \mathcal{V} is the finite element approximation space. The subspaces \mathcal{V}_i are either associated with subdomains in domain decomposition applications or coarser grids in multilevel applications. We shall discuss primarily the applications to domain decomposition in this paper, even though the theorems of this paper could be directly applied to provide new estimates for multigrid algorithms. However, it is possible to modify the analysis presented

here and develop sharper estimates for multigrid algorithms. This is done in [10] and provides a theory for multigrid algorithms which does not require regularity estimates for the underlying boundary value problem.

Iterative algorithms involving a product of projectors have been studied by other researchers (cf. [15], [17], [19]). The case of two subspaces has been thoroughly understood (cf. [15], [16]). The results for greater than two projectors only gave either that the methods were convergent (without any estimate on the rate of convergence [15]) or provided a convergence rate which approached 1 faster than exponentially with the number of projectors [17], [19]. Our analysis provides much better bounds for the convergence rate of these methods. In fact, there are applications where our analysis shows that the convergence rate remains bounded away from one even though the number of projectors becomes large (cf. §4).

Another important aspect of our analysis is that it shows that the projectors can be replaced by properly scaled preconditioners without significant deterioration in the convergence rates. Our result applies to the many-level case and to other applications as well.

The paper is organized in the following way. We will provide an abstract analysis for estimating the norms of operators of the form of (1.3) in §2. In §3, we provide estimates for the sum of projectors and define the additive variant of Algorithm 1. In §4, we apply the abstract results of §2 to the multiplicative iterative methods resulting from overlapping domain decomposition. The methods discussed there are extensions of the classical Schwarz alternating algorithm. Finally, the results of numerical experiments illustrating the rapid convergence of the product algorithms will be given in §5.

2. ABSTRACT ANALYSIS OF PRODUCT ALGORITHMS

In this section, we shall present an abstract analysis of products of operators of the form appearing in (1.3). To this end, we assume that we are given a Hilbert space \mathcal{V} with inner product (\cdot, \cdot) and a sequence of linear operators $\{T_i\}$ mapping \mathcal{V} into \mathcal{V} for $i = 1, \dots, J$ which are selfadjoint, positive semidefinite and of norm bounded by a constant $\omega < 2$. The main results of this section are Theorems 2.1 and 2.2, which give explicit bounds for the norm of the product operator in (1.3) in terms of a number of assumptions to be described.

To begin our analysis, let

$$(2.1) \quad E_i = (I - T_i)(I - T_{i-1}) \cdots (I - T_1)$$

for $i = 1, \dots, J$. For convenience, we let $E_0 = I$, the identity operator on \mathcal{V} , and $E = E_J$. We clearly have for $i = 1, \dots, J$

$$(2.2) \quad E_{i-1} - E_i = T_i E_{i-1},$$

from which it follows that

$$(2.3) \quad I - E_i = \sum_{j=1}^i T_j E_{j-1}.$$

The following lemma will be a fundamental part of the analysis of this section.

Lemma 2.1. *Let T_i and E_i , $i = 1, \dots, J$, be as above. Then*

$$(2.4) \quad (2 - \omega) \sum_{i=1}^J (T_i E_{i-1} v, E_{i-1} v) \leq \|v\|^2 - \|Ev\|^2,$$

where $\|\cdot\| = (\cdot, \cdot)^{1/2}$.

Proof. It is obvious from (2.2) that

$$(2.5) \quad \|E_{i-1} v\|^2 - \|E_i v\|^2 = \|T_i E_{i-1} v\|^2 + 2(T_i E_{i-1} v, E_i v).$$

We note that

$$(T_i E_{i-1} v, E_i v) = (T_i (I - T_i) E_{i-1} v, E_{i-1} v),$$

and hence the right-hand side of (2.5) can be bounded by

$$(2.6) \quad \|T_i - E_{i-1} v\|^2 + 2(T_i E_{i-1} v, E_i v) = ((2I - T_i) T_i E_{i-1} v, E_{i-1} v) \\ \geq (2 - \omega)(T_i E_{i-1} v, E_{i-1} v).$$

Combining (2.5), (2.6) and summing gives (2.4). This completes the proof of the lemma. \square

A fundamental assumption for the analysis to be presented in this section involves an inequality regarding the sum of the operators $\{T_i\}$. Specifically, we assume that there is a positive constant C_0 satisfying

$$(2.7) \quad \|v\|^2 \leq C_0 \sum_{i=1}^J (T_i v, v) \quad \text{for all } v \in \mathcal{V}.$$

We can now state and prove the first theorem of this section.

Theorem 2.1. *Assume that (2.7) holds. Then*

$$(2.8) \quad \|Ev\|^2 \leq \gamma \|v\|^2$$

for

$$(2.9) \quad \gamma = 1 - \frac{2 - \omega}{C_0(J + \omega^2 J(J - 1)/2)}$$

or

$$(2.10) \quad \gamma = 1 - \frac{2 - \omega}{2C_0(1 + \omega^2 J(J - 1)/2)}.$$

Proof. We note that it clearly suffices to prove that for $v \in \mathcal{V}$

$$\|v\|^2 \leq C_0(J + \omega^2 J(J - 1)/2)/(2 - \omega)(\|v\|^2 - \|Ev\|^2)$$

and

$$\|v\|^2 \leq 2C_0(1 + \omega^2 J(J-1)/2)/(2 - \omega)(\|v\|^2 - \|Ev\|^2).$$

Applying Lemma 2.1 and (2.7), we see that the theorem will be proved if we can show

$$(2.11) \quad \sum_{i=1}^J (T_i v, v) \leq (J + \omega^2 J(J-1)/2) \sum_{i=1}^J (T_i E_{i-1} v, E_{i-1} v)$$

and

$$(2.12) \quad \sum_{i=1}^J (T_i v, v) \leq 2(1 + \omega^2 J(J-1)/2) \sum_{i=1}^J (T_i E_{i-1} v, E_{i-1} v).$$

By (2.3),

$$(2.13) \quad (T_i v, v) = (T_i v, E_{i-1} v) + \sum_{j=1}^{i-1} (T_i v, T_j E_{j-1} v).$$

Using the bound for the operators $\{T_i\}$ and the Schwarz inequality gives

$$(2.14) \quad \begin{aligned} (T_i v, v) &\leq (T_i v, v)^{1/2} (T_i E_{i-1} v, E_{i-1} v)^{1/2} \\ &\quad + \omega (T_i v, v)^{1/2} \sum_{j=1}^{i-1} (T_j E_{j-1} v, E_{j-1} v)^{1/2} \\ &\leq (T_i v, v)^{1/2} \left((T_i E_{i-1} v, E_{i-1} v)^{1/2} \right. \\ &\quad \left. + \omega \sqrt{i-1} \left(\sum_{j=1}^{i-1} (T_j E_{j-1} v, E_{j-1} v) \right)^{1/2} \right). \end{aligned}$$

Thus, by first eliminating $(T_i v, v)$ on the right-hand side and then applying the Schwarz inequality, (2.14) can be bounded as follows:

$$(2.15) \quad \begin{aligned} (T_i v, v) &\leq \left((T_i E_{i-1} v, E_{i-1} v)^{1/2} \right. \\ &\quad \left. + \omega \sqrt{i-1} \left(\sum_{j=1}^{i-1} (T_j E_{j-1} v, E_{j-1} v) \right)^{1/2} \right)^2 \\ &\leq (1 + \omega^2(i-1)) \sum_{j=1}^J (T_j E_{j-1} v, E_{j-1} v). \end{aligned}$$

Alternatively,

$$(2.16) \quad (T_i v, v) \leq 2 \left((T_i E_{i-1} v, E_{i-1} v) + \omega^2(i-1) \left(\sum_{j=1}^{i-1} (T_j E_{j-1} v, E_{j-1} v) \right) \right).$$

Summing (2.15) and (2.16) over i proves (2.11) and (2.12). This completes the proof of the theorem. \square

Remark 2.1. Note that (2.9) provides a better estimate when, for example, $\omega \geq 1$. In contrast, (2.10) provides a better estimate when ω is small and $j > 2$. In fact, the form of estimate (2.10) suggests that it may be possible to accelerate the convergence of the algorithms by scaling the T_i 's. Assume that we are given a sequence of symmetric positive semidefinite operators \tilde{T}_i , $i = 1, \dots, J$, with norm bounded by ω and satisfying

$$\|v\|^2 \leq \tilde{C}_0 \sum_{i=1}^J (\tilde{T}_i v, v) \quad \text{for all } v \in \mathcal{V}.$$

Defining $T_i = \alpha \tilde{T}_i$ and applying (2.10) gives that the reduction corresponding to the algorithm with $\{T_i\}$ is bounded, for example, by

$$\gamma = 1 - \frac{\alpha(2 - \omega\alpha)}{2\tilde{C}_0(1 + \alpha^2\omega^2J(J-1)/2)}.$$

Taking $\alpha = 1/(\omega\sqrt{J(J-1)})$ gives that the reduction rate is bounded, for example, by

$$\gamma = 1 - \frac{1}{3\tilde{C}_0\omega J}.$$

Remark 2.2. One important application of the theorems of this section is the case where the operator $T_i = P_i$, the orthogonal projection onto a subspace \mathcal{V}_i of \mathcal{V} . In this case, the proof of (2.7) reduces to the construction of a decomposition of $v \in \mathcal{V}$ of the form $v = \sum_{i=1}^J v_i$ with $v_i \in \mathcal{V}_i$ satisfying

$$(2.17) \quad \sum_{i=1}^J \|v_i\|^2 \leq C_0 \|v\|^2.$$

This was observed in [15]. In fact, if (2.17) holds, then

$$\begin{aligned} \|v\|^2 &= \sum_{i=1}^J (v, v_i) = \sum_{i=1}^J (P_i v, v_i) \\ &\leq \left(\sum_{i=1}^J \|v_i\|^2 \right)^{1/2} \left(\sum_{i=1}^J (T_i v, v) \right)^{1/2} \leq C_0 \sum_{i=1}^J (T_i v, v). \end{aligned}$$

Theorem 2.1 provides good bounds for a number of applications. However, there is an important class of applications (see §4) where $\{T_i\}$ satisfies an interaction property. To describe this property, we first define

$$\kappa_{ij} = \begin{cases} 0 & \text{if } T_i T_j = 0, \\ 1 & \text{otherwise.} \end{cases}$$

We consider a set $I_0 \subseteq [1, \dots, J]$ and define J_0 to be the number of integers in I_0 and

$$N_0 = \max_{j \notin I_0} \sum_{i \notin I_0} \kappa_{ij}.$$

It can happen that the numbers J_0 and N_0 remain small even when J becomes large. Basically, the indices in I_0 correspond to T_i 's which interact with many of the other T_i 's. The remaining T_i 's interact with at most N_0 of the T_i 's with indices not in I_0 . We shall also let \tilde{I}_0 denote the integers in $[1, \dots, J]$ which are not in I_0 . Under these assumptions, we have the following theorem.

Theorem 2.2. *Let N_0 and J_0 be defined as above. If (2.7) holds, then*

$$\|Ev\|^2 \leq \gamma \|v\|^2$$

for

$$\gamma = 1 - \frac{2 - \omega}{3C_0(1 + \omega^2(J_0 + N_0)K_0)},$$

where $K_0 = \max(J_0, N_0)$.

Before proving the theorem, we prove the following lemma.

Lemma 2.2. *Let J_0 and N_0 be defined as above and $u_i, v_i \in \mathcal{M}$ for $i = 1, \dots, J$. If S_1 is a subset of $\tilde{I}_0 \times \tilde{I}_0$, then*

$$(2.18) \quad \left(\sum_{(i,j) \in S_1} (T_i u_i, T_j v_j) \right)^2 \leq \omega^2 N_0^2 \sum_{i \in \tilde{I}_0} (T_i u_i, u_i) \sum_{j \in \tilde{I}_0} (T_j v_j, v_j).$$

If S_2 is a subset of $I_0 \times I_0$, then

$$(2.19) \quad \left(\sum_{(i,j) \in S_2} (T_i u_i, T_j v_j) \right)^2 \leq \omega^2 J_0^2 \sum_{i \in I_0} (T_i u_i, u_i) \sum_{j \in I_0} (T_j v_j, v_j).$$

Finally, if S_3 is a subset of $I_0 \times \tilde{I}_0$, then

$$(2.20) \quad \left(\sum_{(i,j) \in S_3} (T_i u_i, T_j v_j) \right)^2 \leq \omega^2 J_0 N_0 \sum_{i \in I_0} (T_i u_i, u_i) \sum_{j \in \tilde{I}_0} (T_j v_j, v_j).$$

Proof. The bound on the $\{T_i\}$ implies that

$$|(T_i u_i, T_j v_j)| \leq \omega \kappa_{ij} (T_i u_i, u_i)^{1/2} (T_j v_j, v_j)^{1/2}.$$

Consequently,

$$\begin{aligned} \left(\sum_{(i,j) \in S_1} (T_i u_i, T_j v_j) \right)^2 &\leq \omega^2 \sum_{i \in \tilde{I}_0} (T_i u_i, u_i) \sum_{i \in \tilde{I}_0} \left(\sum_{j \in \tilde{I}_0} \kappa_{ij} (T_j v_j, v_j)^{1/2} \right)^2 \\ &\leq \omega^2 N_0 \sum_{i \in \tilde{I}_0} (T_i u_i, u_i) \sum_{i \in \tilde{I}_0} \sum_{j \in \tilde{I}_0} \kappa_{ij} (T_j v_j, v_j) \\ &\leq \omega^2 N_0^2 \sum_{i \in \tilde{I}_0} (T_i u_i, u_i) \sum_{j \in \tilde{I}_0} (T_j v_j, v_j). \end{aligned}$$

This proves (2.18). The proof of (2.19) is similar.

Let $\tilde{I}_0(i)$ denote the possibly empty set of indices $j \in \tilde{I}_0$ such that $(i, j) \in S_3$. To prove (2.20),

$$\begin{aligned}
 \left(\sum_{(i,j) \in S_3} (T_i u_i, T_j v_j) \right)^2 &= \left(\sum_{i \in I_0} \left(T_i u_i, \sum_{j \in \tilde{I}_0(i)} T_j v_j \right) \right)^2 \\
 (2.21) \quad &\leq J_0 \sum_{i \in I_0} \|T_i u_i\|^2 \left\| \sum_{j \in \tilde{I}_0(i)} T_j v_j \right\|^2 \\
 &\leq \omega J_0 \sum_{i \in I_0} (T_i u_i, u_i) \left\| \sum_{j \in \tilde{I}_0(i)} T_j v_j \right\|^2.
 \end{aligned}$$

We clearly have that for any i , $\tilde{I}_0(i) \subseteq \tilde{I}_0$, and hence (2.18) implies

$$(2.22) \quad \left\| \sum_{j \in \tilde{I}_0(i)} T_j v_j \right\|^2 \leq \omega N_0 \sum_{j \in \tilde{I}_0} (T_j v_j, v_j).$$

Combining (2.21) and (2.22) proves (2.20). \square

Proof of Theorem 2.2. As in the proof of Theorem 2.1 (compare with (2.11)), it suffices to prove that for $v \in \mathcal{V}$

$$(2.23) \quad \sum_{i=1}^J (T_i v, v) \leq 3(1 + \omega^2(J_0 + N_0)K_0) \sum_{i=1}^J (T_i E_{i-1} v, E_{i-1} v).$$

Using (2.13) and partitioning the (i, j) indices appearing in the double sum into sets S_2 and S_3 yield

$$\begin{aligned}
 \sum_{i \in I_0} (T_i v, v) &= \sum_{i \in I_0} (T_i v, E_{i-1} v) + \sum_{i \in I_0} \sum_{j=1}^{i-1} (T_i v, T_j E_{j-1} v) \\
 &= \sum_{i \in I_0} (T_i v, E_{i-1} v) + \sum_{(i,j) \in S_2} (T_i v, T_j E_{j-1} v) \\
 &\quad + \sum_{(i,j) \in S_3} (T_i v, T_j E_{j-1} v),
 \end{aligned}$$

where

$$S_2 \subseteq I_0 \times I_0 \quad \text{and} \quad S_3 \subseteq I_0 \times \tilde{I}_0.$$

Thus, by the arithmetic-geometric mean inequality and Lemma 2.2,

$$\begin{aligned}
 \left(\sum_{i \in I_0} (T_i v, v) \right)^2 &\leq 3 \left\{ \sum_{i \in I_0} (T_i v, v) \sum_{i \in I_0} (T_i E_{i-1} v, E_{i-1} v) \right. \\
 &\quad + \left[\sum_{(i,j) \in S_2} (T_i v, T_j E_{j-1} v) \right]^2 \\
 &\quad \left. + \left[\sum_{(i,j) \in S_3} (T_i v, T_j E_{j-1} v) \right]^2 \right\} \\
 (2.24) \quad &\leq 3 \sum_{i \in I_0} (T_i v, v) \left\{ \sum_{i \in I_0} (T_i E_{i-1} v, E_{i-1} v) \right. \\
 &\quad + \omega^2 J_0^2 \sum_{i \in I_0} (T_i E_{i-1} v, E_{i-1} v) \\
 &\quad \left. + \omega^2 J_0 N_0 \sum_{i \in \tilde{I}_0} (T_i E_{i-1} v, E_{i-1} v) \right\}.
 \end{aligned}$$

It follows from (2.24) that

$$\begin{aligned}
 \sum_{i \in \tilde{I}_0} (T_i v, v) &\leq 3 \left\{ \sum_{i \in I_0} (T_i E_{i-1} v, E_{i-1} v) + \omega^2 J_0^2 \sum_{i \in I_0} (T_i E_{i-1} v, E_{i-1} v) \right. \\
 (2.25) \quad &\quad \left. + \omega^2 J_0 N_0 \sum_{i \in \tilde{I}_0} (T_i E_{i-1} v, E_{i-1} v) \right\}.
 \end{aligned}$$

Similarly,

$$\begin{aligned}
 \sum_{i \in \tilde{I}_0} (T_i v, v) &\leq 3 \left\{ \sum_{i \in \tilde{I}_0} (T_i E_{i-1} v, E_{i-1} v) + \omega^2 N_0^2 \sum_{i \in \tilde{I}_0} (T_i E_{i-1} v, E_{i-1} v) \right. \\
 (2.26) \quad &\quad \left. + \omega^2 J_0 N_0 \sum_{i \in I_0} (T_i E_{i-1} v, E_{i-1} v) \right\}.
 \end{aligned}$$

Combining (2.25) with (2.26) proves (2.23). This completes the proof of the theorem. \square

Remark 2.3. Theorem 2.2 suggests that a smaller K_0 results in a better convergence estimate. Since the numbers J_0 and N_0 merely characterize the interaction between T_i 's, $i = 1, \dots, J$, one might think that less interaction yields

a better convergence rate. However, less interaction between subdomains may result in a larger constant C_0 in (2.7).

3. ADDITIVE PRECONDITIONING ALGORITHMS

Section 2 developed a theory which related the constant C_0 in (2.7) to the norms of products of the form (1.3). In this section, we shall provide converse estimates, i.e., assuming that the norms of the product is bounded by some constant $\gamma < 1$, we shall give a simple bound for C_0 . We will then discuss the implications of these results to a natural additive variant of Algorithm 1.

Theorem 3.1. *Assume that the norm of the operators $\{T_i\}$ are bounded by a positive constant ω less than 2 and there exists a constant $\gamma \in (0, 1)$ such that*

$$\|Ev\|^2 \leq \gamma \|v\|^2 \quad \text{for all } v \in \mathcal{V}.$$

Then (2.7) holds with $C_0 \leq 4(1 - \gamma)^{-1}(2 - \omega)^{-1}$.

Proof. Let $v \in \mathcal{V}$. We first note that by Lemma 2.1,

$$\begin{aligned} (3.1) \quad & (2 - \omega) \sum_{i=1}^J (T_i E_{i-1} v, E_{i-1} v) \\ & \leq \|v\|^2 - \|Ev\|^2 = 2(v, (I - E)v) - \|(I - E)v\|^2 \\ & \leq 2(v, (I - E)v) = 2 \sum_{i=1}^J (v, T_i E_{i-1} v) \\ & \leq 2 \left(\sum_{i=1}^J (T_i v, v) \right)^{1/2} \left(\sum_{i=1}^J (T_i E_{i-1} v, E_{i-1} v) \right)^{1/2}. \end{aligned}$$

Hence,

$$\left(\sum_{i=1}^J (T_i E_{i-1} v, E_{i-1} v) \right)^{1/2} \leq 2(2 - \omega)^{-1} \left(\sum_{i=1}^J (T_i v, v) \right)^{1/2}.$$

By (3.1) and the definition of γ ,

$$\begin{aligned} (1 - \gamma) \|v\|^2 & \leq \|v\|^2 - \|Ev\|^2 \\ & \leq 2 \left(\sum_{i=1}^J (T_i v, v) \right)^{1/2} \left(\sum_{i=1}^J (T_i E_{i-1} v, E_{i-1} v) \right)^{1/2} \\ & \leq 4(2 - \omega)^{-1} \sum_{i=1}^J (T_i v, v). \end{aligned}$$

This completes the proof of the theorem. \square

Theorem 3.1 provides a lower bound for the smallest eigenvalue of the operator $\sum_i T_i$. To estimate rates of convergence for the additive preconditioned

algorithms to be described, we must also bound the largest eigenvalue. We obviously have that

$$\sum_{i=1}^J (T_i v, v) \leq \omega J \|v\|^2.$$

A somewhat better bound may be derived when one assumes the interaction property of §2. Such a result is given by the following proposition.

Proposition 3.1. *Let N_0 and J_0 be as in §2. Assume that the norms of the operators T_i are bounded by a positive constant ω (which may be greater than or equal to 2). Then*

$$(3.2) \quad \left(\sum_{i=1}^J T_i v, v \right) \leq \omega (J_0 + N_0) \|v\|^2 \quad \text{for all } v \in \mathcal{V}.$$

Proof. Note that

$$\left(\sum_{i=1}^J T_i v, v \right) \leq \left\| \sum_{i=1}^J T_i v \right\| \|v\|.$$

Define the sets

$$\begin{aligned} S_1 &= \tilde{I}_0 \times \tilde{I}_0, & S_2 &= I_0 \times I_0, \\ S_3 &= I_0 \times \tilde{I}_0, & S_4 &= \tilde{I}_0 \times I_0. \end{aligned}$$

Then

$$\begin{aligned} \left\| \sum_{i=1}^J T_i v \right\|^2 &= \sum_{(i,j) \in S_1} (T_i v, T_j v) + \sum_{(i,j) \in S_2} (T_i v, T_j v) \\ &\quad + \sum_{(i,j) \in S_3} (T_i v, T_j v) + \sum_{(i,j) \in S_4} (T_i v, T_j v). \end{aligned}$$

Applying Lemma 2.2 gives

$$\begin{aligned} \left\| \sum_{i=1}^J T_i v \right\|^2 &\leq \omega N_0 \sum_{\tilde{I}_0} (T_i v, v) + \omega J_0 \sum_{I_0} (T_i v, v) \\ &\quad + 2\omega \sqrt{N_0 J_0} \left(\sum_{\tilde{I}_0} (T_i v, v) \right)^{1/2} \left(\sum_{I_0} (T_i v, v) \right)^{1/2} \\ &\leq \omega (J_0 + N_0) \sum_{i=1}^J (T_i v, v). \end{aligned}$$

The proposition follows by combining the above inequalities. \square

In the remainder of this section, we apply the above results to an additive variant of the algorithm described in the Introduction. We include this discussion for completeness, since the algorithms are so closely related to those discussed earlier. We note that the additive algorithms may have an advantage

when used on a computer with a parallel architecture, since the terms can always be evaluated concurrently. However, compared with the corresponding multiplicative variants, these algorithms often produce a somewhat slower rate of convergence in practice, mainly because of a larger upper eigenvalue (see §6).

The additive algorithms are defined in terms of a preconditioner B for the operator A in (1.2). Specifically, we set

$$(3.3) \quad B = \sum_{i=1}^J R_i Q_i$$

and note that

$$BA = \sum_{i=1}^J T_i,$$

where $T_i = R_i A_i P_i$ as defined in the Introduction. Effective algorithms for the solution of (1.1) are obtained by iterative methods applied to the preconditioned equations

$$BAu = Bf.$$

Clearly, BA is a symmetric operator in the inner product $(\cdot, \cdot) \equiv A(\cdot, \cdot)$, and one can apply, for example, the conjugate gradient iterative procedure. Alternatively, one could use the simplest linear iterative scheme as in the following algorithm.

Algorithm 2. Given $u^l \in \mathcal{V}$, define

$$(3.4) \quad u^{l+1} = u^l + \tau B(f - Au^l).$$

In the above algorithm, τ is a positive iteration parameter. The sequence of iterates converges to u provided that τ times the maximum eigenvalue of BA is less than two. The optimal rate of convergence is obtained by taking $\tau = 2/(\lambda_0 + \lambda_1)$ and results in a reduction of $(K(BA) - 1)/(K(BA) + 1)$ per iteration. Here, λ_0 and λ_1 denote respectively the smallest and largest eigenvalue of BA , and $K(BA) = \lambda_1/\lambda_0$ is the condition number. The results in this section can be used to provide bounds for $K(BA) = K(\sum_i T_i)$.

From (3.4), it is clear that the correction due to any subspace \mathcal{V}_i in the additive algorithm is computed by applying T_i to the original error $u - u^l = A^{-1}(f - Au^l)$ and adding the resulting contributions. In contrast, the product algorithm (Algorithm 1) involves applying T_i consecutively to the latest error.

4. OVERLAPPING DOMAIN DECOMPOSITION METHODS

In this section, the abstract results developed earlier will be applied to examples where the subspaces have been defined by overlapping domain decomposition. We shall consider both the continuous case as well as the case of finite element approximation. The results of §2 show that in order to bound the rate of convergence for the product algorithms, it suffices to prove inequality (2.7) and apply Theorem 2.1 or 2.2. For the applications of this section, the

main part of the proof of (2.7) was given in [13]. We first consider the case of overlapping subdomains of quasi-uniform size d without the use of a “coarse” problem. The convergence rate for this method deteriorates as the number of subdomains increases. Finally, to develop a method with a uniform rate of convergence (not depending upon the mesh size or the number of subdomains), the sequence of subspaces is augmented with a coarse subspace, i.e., a subspace of functions on a mesh of size d .

We will consider the following elliptic boundary value problem. For simplicity, we shall restrict to problems in two-dimensional Euclidean space R^2 . Extensions to higher dimensions are straightforward. Let Ω be a polygonal domain in R^2 and consider the problem

$$(4.1) \quad - \sum_{i,j=1}^2 \partial_i(a_{ij}\partial_j u) + au = f \quad \text{in } \Omega, \\ u = 0 \quad \text{on } \partial\Omega.$$

Here, $\partial_i = \frac{\partial}{\partial x_i}$, and we assume that the matrix $(a_{ij})_{2 \times 2}$ is symmetric for each $x \in \Omega$ and uniformly positive definite. We further assume that the coefficients of $(a_{ij})_{2 \times 2}$ and a are continuously differentiable and that $a \geq 0$. The solution u of (4.1) is in $H_0^1(\Omega)$ (the functions defined on Ω which vanish on $\partial\Omega$ in the appropriate sense and together with their first derivatives are L^2 integrable) and satisfies

$$(4.2) \quad A(u, \chi) = (f, \chi)_0 \quad \text{for all } \chi \in H_0^1(\Omega).$$

Here, $A(\cdot, \cdot)$ denotes the bilinear form associated with the operator in (4.1) defined by

$$A(u, v) = \sum_{i,j=1}^2 \int_{\Omega} a_{ij} \partial_j u \partial_i v \, dx + \int_{\Omega} auv \, dx$$

and $(\cdot, \cdot)_0$ denotes the L^2 -inner product on Ω .

We will consider overlapping domain decomposition applied to both the continuous problem (i.e., $\mathcal{V} = H_0^1(\Omega)$) and its corresponding finite element approximations. Most of our development will be directed to the finite element exposition, the exposition for the continuous application is similar and will be discussed in accompanying remarks. The continuous case is only of theoretical interest, since actual computer implementation requires the use of finite-dimensional approximation spaces.

We first define the finite element approximation scheme. For simplicity of presentation, we use the simplest finite element spaces. Extensions to more complex elements are possible. Assume that Ω has been triangulated, $\Omega = \bigcup_i \tau_i$, where the triangles $\{\tau_i\}$ are of quasi-uniform size h with $h \in (0, 1]$. By this we mean that there are constants C_0, C_1 not depending on h such that each triangle τ_i is contained in (respectively, contains) a ball of radius $C_1 h$ (respectively $C_0 h$). The finite element space \mathcal{V} is defined to be the functions

which are continuous on Ω , piecewise linear with respect to the triangulation $\{\tau_i\}$, and vanish on $\partial\Omega$. The finite element approximation to the solution of (4.1) is the function $U \in \mathcal{V}$ satisfying

$$(4.3) \quad A(U, \chi) = (f, \chi)_0 \quad \text{for all } \chi \in \mathcal{V}.$$

Estimates for the error between u and U are well known (cf. [1], [2], [12]). We shall be concerned with iterative processes of the form of Algorithm 1 for the computation of U .

To define the overlapping domain decomposition method, we start by assuming that we are given a set of overlapping subdomains $\{\Omega_i\}_{i=1}^J$ whose boundaries align with the mesh triangulation defining \mathcal{V} . Associated with these subdomains, we assume that there is a partition of unity $\sum_{i=1}^J \rho_i = 1$ defined on Ω satisfying

$$(4.4) \quad \text{supp } \rho_i \subseteq \Omega_i \cup \partial\Omega,$$

$$(4.5) \quad \|\rho_i\|_{\infty, \Omega_i} \leq C,$$

$$(4.6) \quad \|\nabla \rho_i\|_{\infty, \Omega_i} \leq Cd^{-1},$$

for $i = 1, \dots, J$. Here, $\|\cdot\|_{\infty, D}$ denotes the L^∞ norm of a function defined on a domain D . One way of defining the subdomains and the associated partition is by starting with disjoint open sets $\{\Omega_i^0\}_{i=1}^J$ with $\bar{\Omega} = \bigcup_{i=1}^J \bar{\Omega}_i^0$ and $\{\Omega_i^0\}_{i=1}^J$ quasi-uniform of size d . The subdomain Ω_i is defined to be a mesh subdomain containing Ω_i^0 with the distance from $\partial\Omega_i \cap \Omega$ to Ω_i^0 greater than or equal to Cd for some prescribed constant C . The construction of functions ρ_i satisfying (4.4)–(4.6) is then straightforward.

Remark 4.1. To define the overlapping domain decomposition method for the continuous problem, we start by partitioning Ω into a sequence of overlapping subdomains $\Omega = \bigcup_{i=1}^J \Omega_i$. We assume that these subdomains have Lipschitz continuous boundaries and that there is a corresponding partition of unity $\sum_{i=1}^J \rho_i = 1$ satisfying (4.4)–(4.6). As above, the subdomains can be defined by starting with disjoint open sets $\{\Omega_i^0\}$ and expanding by Cd .

In either case, the subspace \mathcal{V}_i is defined by

$$\mathcal{V}_i = \{\varphi \mid \varphi \in \mathcal{V}, \text{supp } \varphi \subseteq \Omega_i\}.$$

The first product method associated with these subspaces is when T_i of Algorithm 1 is taken to be P_i , the elliptic projection into \mathcal{V}_i . For the two-domain case, this is the classical Schwarz method (cf. [3], [15], [18]). The inner product (\cdot, \cdot) used in §2 is defined by $(u, v) \equiv A(u, v)$ for all $u, v \in \mathcal{V}$. By Remark 2.2, estimate (2.7) for this case will follow from the existence of a decomposition $v = \sum_{i=1}^J v_i$, with $v_i \in \mathcal{V}_i$ satisfying

$$(4.7) \quad \sum_{i=1}^J A(v_i, v_i) \leq C_0 A(v, v).$$

The decomposition used in [13], $v = \sum_{i=1}^J I_h(\rho_i v)$, where I_h denotes the nodal interpolation operator onto \mathcal{V} , satisfies (4.7) with $C_0 = Cd^{-2}$. In fact (cf. [13]), using (4.4)–(4.6), one shows that

$$(4.8) \quad \sum_{i=1}^J A(I_h(\rho_i v), I_h(\rho_i v)) \leq c\{d^{-2}\|v\|_0^2 + A(v, v)\} \quad \text{for all } v \in \mathcal{V}.$$

In the continuous case, one simply takes I_h to be the identity. We now give the following theorem.

Theorem 4.1. *Let n_i denote the number of subdomains Ω_j with $\Omega_j \cap \Omega_i \neq \emptyset$ and assume that $n_i \leq n$, $i = 1, \dots, J$. In addition, assume that (4.4)–(4.6) hold. Then*

$$\|Ev\|^2 \leq \left(1 - \frac{d^2}{Cn^2}\right) \|v\|^2 \quad \text{for all } v \in \mathcal{V},$$

where the constant C does not depend on d or h . This holds for overlapping domain decomposition in the continuous as well as the finite element application.

Proof. We have already proved estimate (2.7). To apply Theorem 2.2, we need only identify N_0 and I_0 . We set $I_0 = \emptyset$. Note that κ_{ij} is nonzero only if $\Omega_i \cap \Omega_j \neq \emptyset$. Consequently, we can take $N_0 = n$. This completes the proof of the theorem. \square

Remark 4.2. The convergence estimate for the product algorithm in the case of two subdomains was given in [15]. It was also demonstrated in [15] that the product algorithm described above gives rise to the error propagation matrix associated with the classical Schwarz overlapping method [18]. For the case of more than two subdomains, convergence was proved in [15] but no estimate of the rate was given. Weak estimates were given in [17], [19]. Theorem 4.1 provides new estimates in the case of more than two projectors.

Remark 4.3. It is possible to view point-Gauss-Seidel iteration as a product iteration of the above form. For each node x_m of the triangulation, we define the subdomain Ω_m to be the union of the triangles $\tau_l \in \{\tau_k\}$ which have x_m as a vertex. In this case, the subspaces \mathcal{V}_i are one-dimensional and the projectors P_i are trivial to compute. It is then easy to see that the product algorithms with $R_i = A_i^{-1}$ correspond to the Gauss-Seidel method. Inequality (2.7) is a direct consequence of inverse properties of \mathcal{V} , since the decomposition $v = \sum v_i$ is unique and satisfies

$$\sum_i A(v_i, v_i) \leq C \sum_i v(x_i)^2 \leq Ch^{-2} A(v, v).$$

Thus, Theorem 2.2 provides an alternative proof of the convergence rate of Gauss-Seidel iteration.

In the finite element case, this algorithm requires the computation of functions $u_i \in \mathcal{V}_i$ satisfying

$$(4.9) \quad A(u_i, \varphi) = G_i(\varphi) \quad \text{for all } \varphi \in \mathcal{V}_i$$

for appropriate linear functionals G_i . To avoid solving systems of the form (4.9), we introduce T_i by preconditioning. Specifically, let R_i be a scaled preconditioner for A_i , i.e., $R_i A_i$ is a symmetric positive definite operator on \mathcal{V}_i satisfying

$$(4.10) \quad C_R A(w, w) \leq A(R_i A_i w, w) \leq A(w, w) \quad \text{for all } w \in \mathcal{V}_i.$$

Define $T_i = R_i A_i P_i$. Then (4.7) implies

$$(4.11) \quad \begin{aligned} A(w, w) &\leq C_0 \sum_{i=1}^J A(P_i w, w) \\ &\leq C_0 / C_R \sum_{i=1}^J A(T_i w, w) \quad \text{for all } w \in \mathcal{V}, \end{aligned}$$

i.e., (2.7) holds for $\{T_i\}$. The development of preconditioners for (4.9) has been subject to intensive research [4]–[9], [11], [13], [14], etc. Computational results for this algorithm with multigrid V-cycle preconditioning are given in §5. Combining (4.11) and Theorem 2.2 implies the following theorem.

Theorem 4.2. *Assume that the hypotheses for Theorem 4.1 hold. Define $T_i = R_i A_i P_i$ for $i = 1, \dots, J$ and assume that R_i satisfies (4.10) with C_R independent of d and h . Then*

$$\|Ev\|^2 \leq \left(1 - \frac{d^2}{Cn^2}\right) \|v\|^2 \quad \text{for all } v \in \mathcal{V},$$

where the constant C does not depend on d or h . This holds for overlapping domain decomposition in the continuous as well as the finite element application.

To obtain algorithms which converge with rates that are independent of d , one can add a coarse subspace. Let \mathcal{V}_0 be a finite element subspace of \mathcal{V} defined from a quasi-uniform triangulation of Ω of size d . Let Q denote the $L^2(\Omega)$ orthogonal projection onto \mathcal{V}_0 . Then, for $v \in \mathcal{V}$,

$$v = Qv + \sum_{i=1}^J I_h(\rho_i(v - Qv)) = \sum_{i=0}^J v_i$$

is a decomposition of v into $\{\mathcal{V}_i\}_{i=0}^J$ which, by (4.8), satisfies

$$\sum_{i=0}^J A(v_i, v_i) \leq C \{A(Qv, Qv) + d^{-2} \|v - Qv\|_0^2 + A(v - Qv, v - Qv)\}.$$

It is known that (cf. [20])

$$\|v - Qv\|_0 \leq Cd \|v\|_1, \quad \|Qv\|_1 \leq C \|v\|_1,$$

which implies

$$(4.12) \quad \sum_{i=0}^J A(v_i, v_i) \leq CA(v, v),$$

i.e., (2.7) holds with C_0 independent of d . Inequality (4.12) has been given in [14]. We can apply Theorem 2.2 to this formulation. In this case, I_0 contains the one integer corresponding to the coarse subdomain. We have the following theorem.

Theorem 4.3. *Assume that the hypotheses for Theorem 4.2 hold. For the algorithm which includes a coarse grid term $j = 0$, we have*

$$\|Ev\| \leq \gamma \|v\| \quad \text{for all } v \in \mathcal{V},$$

where γ is a constant which is less than one and does not depend on d or h . This holds for overlapping domain decomposition in the continuous as well as the finite element application.

5. NUMERICAL RESULTS

In this section, we provide the results of numerical examples illustrating the theory developed in earlier sections. We shall consider the model problem

$$(5.1) \quad \begin{aligned} -\Delta u &= f && \text{in } \Omega, \\ u &= 0 && \text{on } \partial\Omega, \end{aligned}$$

where Δ denotes the Laplacian and Ω is the unit square $[0, 1] \times [0, 1]$. Problem (5.1) is discretized by the finite element method. Specifically, the domain Ω is first partitioned into $m \times m$ square subdomains of side length $1/m$. Each smaller square is then divided into two triangles by one of the diagonals (e.g., the diagonal which goes from the bottom left to the upper right-hand corners of the square). The approximation space \mathcal{V} is defined to be the set of functions which are continuous on Ω , piecewise linear with respect to the triangulation, and vanish on $\partial\Omega$. We seek the Galerkin solution $U \in \mathcal{V}$ satisfying

$$(5.2) \quad D(U, \phi) = (f, \phi)_0 \quad \text{for all } \phi \in \mathcal{V}.$$

We shall consider a number of overlapping algorithms for the solution of the discrete equations determining U . The first algorithm uses overlapping subdomains defined in terms of strips. The remaining algorithms are based on overlapping subdomains Ω_i of quasi-uniform size as described in §4.

To define the strip overlapping subspaces, we first define $j - 1$ subregions $\Omega_k = [(k - 1)d, (k + 1)d]$ for $k = 1, \dots, j - 1$, where $d = 1/j$. We assume that the mesh aligns with the subdomain boundaries, i.e., j divides m , and set

- (1) $\mathcal{V}_k = \{\phi \in \mathcal{M} \mid \text{supp } \phi \subseteq \Omega_k\}$.
- (2) $A_k: \mathcal{V}_k \mapsto \mathcal{V}_k$ by $A_k V = \Psi$ where Ψ is the unique function in \mathcal{V}_k satisfying

$$(\Psi, \theta) = D(V, \theta) \quad \text{for all } \theta \in \mathcal{V}_k.$$

Note that here we identify \mathcal{V}_k' with \mathcal{V}_k . We define A as above but with \mathcal{V}_k replaced by \mathcal{V} . Note that Problem (5.2) can be rewritten as $AU = F$, where F is the L^2 -projection of f into \mathcal{V} . For this example, we shall use $R_k = A_k^{-1}$ and hence $T_k = P_k$.

TABLE 5.1
The values of γ for overlapping strips

$m = 1/h$	$\gamma (j = 4)$	$\gamma (j = 8)$	$\gamma (j = 16)$	$\gamma (j = 32)$
16	.21	.59	—	—
32	.21	.59	.86	—
64	.21	.59	.86	.96
128	.21	.59	.86	.96

Table 5.1 gives the numerically computed value for the norm-squared reduction rate for Algorithm 1 with T_l defined in terms of overlapping strip domain decomposition as a function of j and m . Note that the norm-squared reduction rate (the minimal value γ satisfying (2.8)) is the largest eigenvalue of the symmetric iteration operator E^*E . Here, E^* is the adjoint of E with respect to the inner product $A(\cdot, \cdot)$ and is computed by reversing the order of the factors in (2.1).

It is not difficult to prove that the inequality (2.7) is satisfied with $C_0 \leq Cj^2$, and thus Theorem 2.1 guarantees that $\gamma \leq (1 - C/j^2)$. We note that the expected asymptotic behavior is seen comparing the $k = 16$ with the $k = 32$ results where $1 - \gamma$ is reduced by almost a factor of 4. The values of γ in Table 5.1 do not appear to depend upon $h = 1/m$.

For the remaining examples, we consider the above problem and discretization but define overlapping subdomains of quasi-uniform size. More precisely, let $d = 1/j$ and for each $i, l = 1, \dots, j-1$ define the subdomain $\Omega_{il} = [(i-1)d, (i+1)d] \times [(l-1)d, (l+1)d]$. We again assume that j divides m and define subspaces

$$\mathcal{V}_{il} = \{\phi \in \mathcal{V} \mid \text{supp } \phi \subseteq \Omega_{il}\}.$$

As seen in §4, (2.7) holds for the sequence of spaces $\{\mathcal{V}_{il} \mid i, l = 1, \dots, j-1\}$ with constant $C_0 \leq Cj^2$. As discussed in §4, the dependence on j can be removed by using a coarse space, e.g., the space \mathcal{V}_0 defined to be the continuous piecewise linear functions on the mesh of size d which vanish on $\partial\Omega$. We assume that the triangles of this coarse grid mesh are defined so that they align with those of \mathcal{V} and hence $\mathcal{V}_0 \subseteq \mathcal{V}$.

We first consider the product algorithm using the coarse subspace \mathcal{V}_0 and the $(j-1)^2$ overlapping subdomains of quasi-uniform size described above. For this example, we again use $R_k = A_k^{-1}$ and hence $T_k = P_k$. This algorithm converges in relatively few iterations but requires the exact solution of the subspace problems on each step of the iteration. The computed values of the reduction rates ($\sqrt{\gamma}$) for Algorithm 1 were less than .2 for combinations of $m = 16, 32, 64, 128$ and $j = 4, 8, 16$ (see Table 5.2). For comparison, we computed the condition numbers for the corresponding additive algorithm. For the same values of m and j , the additive algorithms gave condition numbers of at most 5.3, which corresponds to a reduction of 0.68 for Algorithm 2.

TABLE 5.2
Reduction rates for the $(K - 1)^2$ overlapping domain decomposition algorithm

$m = 1/h$	$\sqrt{\gamma}: K = 4$	$\sqrt{\gamma}: K = 8$	$\sqrt{\gamma}: K = 16$
16	.17(.53)	—	—
32	.17(.52)	.17(.83)	—
64	.17(.52)	.2(.82)	.2(.95)
128	.17(.52)	.2(.82)	.2(.95)

To illustrate the improvement resulting from including the coarse problem, Table 5.2 reports values for $\sqrt{\gamma}$ for the overlapping method without the coarse problem. These are the values given in parentheses and are always larger than the reduction rates for the algorithm with the coarse grid subspace. These results clearly indicate that the use of a coarse grid problem results in a significant improvement in the rate of convergence.

In almost all realistic applications, the direct solution of subproblems is much more expensive than the evaluation of a suitable preconditioner. To illustrate the effect on the convergence rate of the Algorithm 1, we next consider the previous example but with the direct solves on the subspaces replaced by multigrid preconditioners. Specifically, we employ the V-cycle multigrid algorithm (cf. [5]) using one pre and post Jacobi smoothing on each grid level. This leads to a preconditioning operator $R_{il} : \mathcal{V}_{il} \mapsto \mathcal{V}_{il}$ which satisfies

$$(5.3) \quad 0.4A(v, v) \leq A(R_{il}A_{il}v, v) \leq A(v, v) \quad \text{for all } v \in \mathcal{V}_{il}.$$

The constant 0.4 above was computed numerically and holds for all of the subspace problems which are required for this application, including \mathcal{V}_0 . For this example, the reduction rates for the preconditioned product algorithm (Algorithm 1) were all between .63 and .50 for combinations of $m = 16, 32, 64, 128$ and $k = 4, 8, 16$ (see Figure 5.3). Again, we computed condition numbers for the corresponding additive algorithms and found them to be bounded by 9.8 for the same range of k and m . This corresponds to a reduction of .81 for Algorithm 2 with an appropriate choice of iteration parameter τ . Finally, we computed the reduction rates for the product algorithm with preconditioning but without the coarse grid solve (the reductions are the numbers in parentheses in Table 5.3). As in the case of direct solves, the algorithm without the coarse problem shows a significant loss of efficiency.

In all of the above runs, Algorithm 1 with direct solves replaced by multigrid preconditioning would require, at most, four times as many iterations as the direct solve version. On the other hand, our coding of the simple multigrid V-cycle algorithm actually ran more than four times as fast as the corresponding direct (FFT-based) solving version. Hence, even in the special case where fast algorithms are available for the subproblems, the efficient multigrid preconditioning remains competitive. In more complex problems where fast direct

TABLE 5.3
Reduction rates for the domain decomposition with preconditioning

$m = 1/h$	$\sqrt{\gamma}: K = 4$	$\sqrt{\gamma}: K = 8$	$\sqrt{\gamma}: K = 16$
16	.50(.62)	—	—
32	.56(.62)	.52(.87)	—
64	.56(.62)	.55(.87)	.63(.96)
128	.57(.62)	.57(.87)	.63(.96)

methods are not available, the product algorithms using preconditioning will show a significant improvement in computational efficiency.

It is interesting to note that in the above two examples, the product algorithm converges somewhat faster than the corresponding additive algorithm. On the other hand, the additive algorithms will generally be somewhat easier to implement on a parallel machine. This leads one to the question of when an additive algorithm gives rise to a more efficient algorithm even in a parallel environment. Note that if the subspace problems are relatively large and reasonably parallelizable on the given architecture, then little computational gain will be obtained from executing them concurrently. This seems to suggest that the additive algorithm will be more effective than the product only if the computer has a very large number of processors.

BIBLIOGRAPHY

1. J. P. Aubin, *Approximation of elliptic boundary-value problems*, Wiley-Interscience, New York, 1972.
2. A. K. Aziz and I. Babuška, *Part I, Survey lectures on the mathematical foundations of the finite element method*, The Mathematical Foundations of the Finite Element Method with Applications to Partial Differential Equations (A. K. Aziz, ed.), Academic Press, New York, 1972, pp. 1–362.
3. I. Babuška, *On the Schwarz algorithm in the theory of differential equations of mathematical physics*, Czechoslovak Math. J. **8** (1958), 328–342 (Russian).
4. G. Birkhoff and A. Schoenstadt, Editors, *Elliptic problem solvers*, II (G. Birkhoff and A. Schoenstadt, eds.), Academic Press, Orlando, Florida, 1984.
5. J. H. Bramble and J. E. Pasciak, *New convergence estimates for multigrid algorithms*, Math. Comp. **49** (1987), 311–329.
6. J. H. Bramble, J. E. Pasciak, and A. H. Schatz, *The construction of preconditioners for elliptic problems by substructuring*, I, Math. Comp. **47** (1986), 103–134.
7. ———, *The construction of preconditioners for elliptic problems by substructuring*, II, Math. Comp. **49** (1987), 1–16.
8. ———, *The construction of preconditioners for elliptic problems by substructuring*, III, Math. Comp. **51** (1988), 415–430.
9. ———, *The construction of preconditioners for elliptic problems by substructuring*, IV, Math. Comp. **53** (1989), 1–24.
10. J. H. Bramble, J. E. Pasciak, J. Wang, and J. Xu, *Multigrid results which do not depend upon elliptic regularity assumptions* (in preparation).
11. J. H. Bramble, J. E. Pasciak and J. Xu, *Parallel multilevel preconditioners*, Math. Comp. **55** (1990), 1–22.

12. P. G. Ciarlet, *The finite element method for elliptic problems*, North-Holland, New York, 1978.
13. M. Dryja and O. Widlund, *An additive variant of the Schwarz alternating method for the case of many subregions*, Technical Report 339, Courant Institute of Mathematical Sciences, 1987.
14. ———, *Some domain decomposition algorithms for elliptic problems*, Technical Report 438, Courant Institute of Mathematical Sciences, 1989.
15. P. L. Lions, *On the Schwarz alternating method*, Proc. First Internat. Sympos. on Domain Decomposition Methods for Partial Differential Equations (R. Glowinski, G. H. Golub, G. A. Meurant, and J. Périaux, eds.), SIAM, Philadelphia, PA, 1988.
16. J. Mandel and S. F. McCormick, *Iterative solution of elliptic equations with refinement: The two-level case*, Domain Decomposition Methods (T. F. Chan, R. Glowinski, J. Périaux, and O. B. Widlund, eds.), SIAM, Philadelphia, PA, 1989, pp. 81–92.
17. T. P. Mathew, *Domain decomposition and iterative refinement methods for mixed finite element discretizations of elliptic problems*, Thesis, New York University, 1989.
18. H. A. Schwarz, *Ueber einige Abbildungsaufgaben*, J. Reine Angew. Math. **70** (1869), 105–120. [Ges. Math. Abh., vol.2, 65–83].
19. O. Widlund, *A comparison of some domain decomposition and iterative refinement algorithms for elliptic finite element problems*, Technical Report BSC 88/15, IBM Bergen Scientific Centre, Bergen, Norway, 1988.
20. J. Xu, *Theory of multilevel methods*, Dept. Math. Rep. AM-48, Penn. State University, 1989.

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

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

MATHEMATICS DEPARTMENT, THE UNIVERSITY OF WYOMING, LARAMIE, WYOMING 82071

DEPARTMENT OF MATHEMATICS, PENNSYLVANIA STATE UNIVERSITY, UNIVERSITY PARK, PENNSYLVANIA 16802