

## A COMPARISON OF TWO-LEVEL PRECONDITIONERS BASED ON MULTIGRID AND DEFLATION\*

J. M. TANG<sup>†</sup>, S. P. MACLACHLAN<sup>‡</sup>, R. NABBEN<sup>§</sup>, AND C. VUIK<sup>¶</sup>

**Abstract.** It is well known that two-level and multilevel preconditioned conjugate gradient (PCG) methods provide efficient techniques for solving large and sparse linear systems whose coefficient matrices are symmetric and positive definite. A two-level PCG method combines a traditional (one-level) preconditioner, such as incomplete Cholesky, with a projection-type preconditioner to get rid of the effect of both small and large eigenvalues of the coefficient matrix; multilevel approaches arise by recursively applying the two-level technique within the projection step. In the literature, various such preconditioners are known, coming from the fields of deflation, domain decomposition, and multigrid (MG). At first glance, these methods seem to be quite distinct; however, from an abstract point of view, they are closely related. The aim of this paper is to relate two-level PCG methods with symmetric two-grid (V(1,1)-cycle) preconditioners (derived from MG approaches), in their abstract form, to deflation methods and a two-level domain-decomposition approach inspired by the balancing Neumann–Neumann method. The MG-based preconditioner is often expected to be more effective than these other two-level preconditioners, but this is shown to be not always true. For common choices of the parameters, MG leads to larger error reductions in each iteration, but the work per iteration is more expensive, which makes this comparison unfair. We show that, for special choices of the underlying one-level preconditioners in the deflation or domain-decomposition methods, the work per iteration of these preconditioners is approximately the same as that for the MG preconditioner, and the convergence properties of the resulting two-level PCG methods will also be (approximately) the same. This means that, in this respect, the particular choice of the two-level preconditioner is less important than the choice of the parameters. Numerical experiments are presented to emphasize the theoretical results.

**Key words.** deflation, domain decomposition, multigrid, conjugate gradients, two-grid schemes, SPD matrices, two-level preconditioning

**AMS subject classifications.** 65F08, 65F10, 65N55, 65N22

**DOI.** 10.1137/08072084X

**1. Introduction.** The conjugate gradient (CG) method is a well-known iterative method for solving large linear systems of equations,

$$(1.1) \quad Ax = b, \quad A \in \mathbb{R}^{n \times n},$$

---

\*Received by the editors April 10, 2008; accepted for publication (in revised form) by E. Ng January 6, 2010; published electronically March 10, 2010.

<http://www.siam.org/journals/simax/31-4/72084.html>

<sup>†</sup>Department of Computer Science and Engineering, University of Minnesota, 200 Union Street SE, Minneapolis, MN 55455 (jtang@cs.umn.edu). Part of this author's work was funded by the Dutch BSIK/BRICKS project.

<sup>‡</sup>Corresponding author. Department of Mathematics, Tufts University, 503 Boston Avenue, Medford, MA 02155 (scott.maclachlan@tufts.edu). This author's research was partly supported by the European Community's Sixth Framework Programme, through a Marie Curie International Incoming Fellowship, MIF1-CT-2006-021927, and by the National Science Foundation, under grant DMS-0811022.

<sup>§</sup>Institut für Mathematik, Technische Universität Berlin, MA 3-3, Straße des 17. Juni 136, D-10623 Berlin, Germany (nabben@math.tu-berlin.de). Part of this author's work was funded by the Deutsche Forschungsgemeinschaft (DFG), project NA248/2-3.

<sup>¶</sup>Faculty of Electrical Engineering, Mathematics and Computer Science, Delft Institute of Applied Mathematics, Delft University of Technology, J.M. Burgerscentrum, Mekelweg 4, 2628 CD Delft, The Netherlands (c.vuik@tudelft.nl). Part of this author's work was funded by the Dutch BSIK/BRICKS project.

whose coefficient matrix,  $A$ , is sparse, symmetric, and positive definite (SPD). Matrix  $A$  usually comes from a discretization of elliptic PDEs. The convergence rate of CG is naturally bounded in terms of the condition number of  $A$ , denoted by  $\kappa = \kappa(A) := \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)}$ , where  $\lambda_{\max}(A)$  and  $\lambda_{\min}(A)$  are the largest and smallest nonzero eigenvalues of  $A$ , respectively. If  $\kappa$  is large, it is often more favorable to solve a preconditioned system instead of (1.1). This preconditioned system can be represented as  $\hat{A}\hat{x} = \hat{b}$ , with  $\hat{A} = M^{1/2}AM^{1/2}$ ,  $\hat{x} = M^{-1/2}x$ , and  $\hat{b} = M^{1/2}b$ , and  $M \in \mathbb{R}^{n \times n}$  is an SPD preconditioner. This is equivalent to solving the system

$$(1.2) \quad MAx = Mb.$$

To be effective, the preconditioner,  $M$ , should be chosen such that  $MA$  has a smaller condition number or more clustered spectrum than  $A$  and so that the matrix-vector product  $My$  is cheap to compute, relative to the improvement that such preconditioners provide to the convergence rate. Traditional one-level preconditioners, such as diagonal scaling, basic iterative methods, approximate inverse preconditioning, and incomplete Cholesky preconditioners, are widely used. These preconditioners, however, are well known to not lead to scalable solution algorithms for applications with highly refined grids or large coefficient ratios [23]. In these applications, preconditioned CG (PCG), with these one-level preconditioners, suffers from slow convergence due to the presence of relatively small or large eigenvalues, which have a harmful influence on the condition number of the coefficient matrix. In what follows, we shall refer to such preconditioners as “one-level” preconditioners, to distinguish them from their “two-level” counterparts introduced next.

An alternative to these preconditioners is to incorporate a second matrix within the preconditioner to improve the performance of PCG, so that the resulting approach gets rid of the effect of both small and large eigenvalues. This combined preconditioning is also known as “two-level preconditioning,” and the resulting iterative method is called a “two-level PCG method,” abbreviated as “2L-PCG.” The term “two-level” arises because the second preconditioner involves the solution of an auxiliary problem that can, in some cases, be associated with a second, coarser discretization of the continuum operator. Examples of 2L-PCG methods include preconditioners based on multigrid (MG), domain decomposition, or deflation techniques, where these methods explicitly rely on preconditioning on two levels; see [21] and the references therein. Generalizing (1.2), the linear system that is the basis of any PCG method can be expressed as

$$(1.3) \quad \mathcal{P}Ax = \mathcal{P}b, \quad \mathcal{P} \in \mathbb{R}^{n \times n},$$

where  $\mathcal{P}$  is either a one-level or a two-level preconditioner. If  $\mathcal{P} = M$  for a traditional (one-level) choice of  $M$ , we simply recover the standard PCG method in (1.2). When  $\mathcal{P}$  is derived from deflation, domain decomposition, or multigrid approaches, the resulting preconditioners appear, at first glance, to be quite different. However, it has been shown in [21] that some of these methods are closely related, or even equivalent, in their abstract forms.

In this paper, we focus on the comparison between two-level preconditioners inspired by the balancing Neumann–Neumann (BNN) [13], deflation (DEF) [18], and MG [3, 10, 23, 28] approaches. In [16, 21], it was shown that the preconditioned coefficient matrices based on BNN and DEF have almost the same spectral properties; i.e., all eigenvalues are the same, except a few that differ by the shift from 0 to 1. In

addition, it was shown that these properties are quite similar to those of the multigrid  $V(0,1)$ - and  $V(1,0)$ -cycle preconditioners, even though these are typically not considered as allowable preconditioners for CG. Here, we will compare preconditioners based on deflation and BNN to the 2L-PCG method that mimics a multigrid  $V(1,1)$ -cycle preconditioner, denoted as the MG method. This MG method is not compared with the other methods in [21], since it has very different spectral properties and requires a different theoretical treatment, because of the more general choice of one-level preconditioner allowed within MG. The aim of the current research is to fill this gap and compare the abstract versions of MG, DEF, and BNN.

Of course, the MG method [3, 10, 23, 28] and its properties [1, 6, 9, 14, 17, 26] are well known. Our intention is not to reproduce these results (although some known results needed for the comparison are briefly reviewed) but to compare and connect MG to other well-known 2L-PCG methods. A well-known comparison of multigrid and domain-decomposition preconditioners is that of Xu [29], based on subspace corrections. In [29], it is shown that certain multigrid and domain-decomposition algorithms fall into the same mathematical framework, and, consequently, certain choices of the components within the domain-decomposition framework lead, in fact, to a multigrid algorithm. Here, we consider a more abstract formulation than in [29] and, as a result, derive a more general result, based on a spectral analysis. A similar spectral analysis for a two-level preconditioner based on multigrid principles has been carried out in [4]. In that paper, a specific choice of the two-level preconditioner based on exact eigenvectors of  $MA$  is analyzed, allowing for more pre- and postsmoothing steps per iteration. The resulting two-level preconditioner is called a “multiplicative two-grid spectral preconditioner” and is shown to be effective for many practical applications, particularly when sequences of linear systems must be solved. In this paper, we consider a somewhat more general preconditioner, based on a standard multigrid approach (but considering only one pre- and one postsmoothing step), although eigenvectors of  $MA$  are sometimes used to illustrate the theoretical results. A consequence of our main result is that any analysis of multigrid methods, such as that in [1, 2, 6, 11, 14], can be directly applied to the corresponding domain-decomposition or deflation-based preconditioner.

An important feature of the analysis considered here is that it is based on the assumption that the same algorithmic components are used in all three approaches. Because of their distinct origins, each of the deflation, domain-decomposition, and multigrid algorithms has primarily been analyzed based on typical choices of  $M$  and a corresponding coarse-grid correction process appropriate for a particular approach. In contrast, we compare the 2L-PCG methods by adopting fixed algorithmic components for all approaches. Intuitively, we might expect the multigrid-based approach to yield better convergence properties than the other 2L-PCG methods, since it alone relies on the use of two applications of the fine-level preconditioner (in the pre- and postsmoothing steps), in addition to a single coarse-grid correction step per iteration. DEF, on the other hand, has optimal convergence properties in terms of its spectral properties compared with certain other 2L-PCG methods (although not MG); see [21]. Therefore, our comparison focuses on the relationship between the spectral properties of MG and DEF. However, the comparison between MG and BNN is, in some cases, easier to perform, so BNN is also used in the analysis. A numerical comparison of DEF and MG, using components typical of each approach, for problems related to two-phase fluid flow was recently presented in [12]. In that comparison, the performance of a robust multigrid technique is clearly superior to that of deflation; the current research

was motivated by our desire to understand whether these results were because of some fundamental difference between the multigrid and deflation frameworks, or because of the differences in the choices made for the components within each algorithm's individual framework.

Several important practical issues are ignored in the analysis that follows. Most significantly, we consider only two-level PCG algorithms. In practice, the multilevel extensions of these algorithms are necessary to ensure the efficiency needed to solve problems with the millions and billions of degrees of freedom currently required in the field of computational science and engineering. Analysis of these approaches, however, is much more technical than their two-level counterparts and is beyond the scope of this paper. Furthermore, we do not aim to address several questions about the sensitivities of these algorithms to the perturbations that naturally arise in their multilevel extensions, such as to that of starting guesses or inexact solves on the coarse level. For some analysis of these questions, see the discussion in [21].

This paper is organized as follows. In section 2, the two-level preconditioners denoted by DEF, BNN, and MG are described in detail. Some spectral properties of MG are presented in section 3. Thereafter, in section 4, the spectral properties of MG and DEF are compared for special choices of parameters; it is shown there that MG can be less effective than DEF. In section 5, we show that MG is always superior to DEF for more sophisticated preconditioners. Subsequently, section 6 is devoted to the comparison of the MG, BNN, and DEF preconditioners when their parameters are chosen to yield the same cost per iteration. For special choices of preconditioners, we show that they are almost spectrally equivalent. Section 7 is devoted to some numerical experiments in order to illustrate the theoretical results. Conclusions are presented in section 8. An extended version of a previous draft of this paper is available as a technical report [20].

**2. Two-level preconditioned CG methods.** In this section, we describe in detail the 2L-PCG algorithms that will be examined. The following definition is assumed to hold throughout this paper.

**DEFINITION 2.1.** *Let  $A \in \mathbb{R}^{n \times n}$  be a symmetric and positive-definite matrix, and let  $Z \in \mathbb{R}^{n \times k}$  and  $k < n$  be a deflation-subspace matrix with full rank. Define the invertible Galerkin matrix,  $E \in \mathbb{R}^{k \times k}$ , the correction matrix,  $Q \in \mathbb{R}^{n \times n}$ , and the deflation matrix,  $P \in \mathbb{R}^{n \times n}$ , as follows:*

$$P := I - AQ, \quad Q := ZE^{-1}Z^T, \quad E := Z^T AZ.$$

*In addition, let  $M \in \mathbb{R}^{n \times n}$  be a given preconditioning matrix.*

Note that we make no explicit assumptions here on the form of  $M$ . In some cases, we will additionally assume that  $M$  is nonsingular, symmetric, and/or positive definite. However,  $M$  may also, in some cases, be nonsymmetric, singular, or even indefinite. We will, in all cases, specify when further assumptions are needed.

The following lemma will be frequently used; see [21, 27] for more details.

**LEMMA 2.2.** *Suppose that  $A \in \mathbb{R}^{n \times n}$  and  $Z \in \mathbb{R}^{n \times k}$  are given. Let  $Q$  and  $P$  be as in Definition 2.1, and let  $Ax = b$ . Then,*

- $Q = Q^T$ ,
- $(I - P^T)x = Qb$ ,
- $AP^T = PA$ ,
- $QAQ = Q$ ,
- $P^T Z = PAZ = 0$ , and
- $QAZ = Z$ .

*Proof.* See, e.g., [21, 27].  $\square$

The deflation method (DEF) can be described as follows. In order to solve (1.1), we first decompose  $x$  as

$$(2.1) \quad x = (I - P^T)x + P^T x.$$

The first part,  $(I - P^T)x$ , can be computed directly since  $(I - P^T)x = QAx = Qb$ . The second part,  $P^T x$ , can be computed by solving the deflated system,

$$(2.2) \quad PA\tilde{x} = Pb,$$

where the nonunique solution can be written as  $\tilde{x} = x + y$  with  $y \in \mathcal{N}(PA)$ . From Lemma 2.2, it can be shown that  $P^T \tilde{x} = P^T x$  so that (2.1) becomes

$$(2.3) \quad x = P^T \tilde{x} + Qb.$$

Hence, the unique solution of (1.1) can be obtained via (2.2) and (2.3). Moreover, it is possible to solve the deflated system (2.2) by using an SPD preconditioner,  $M$ . This deflated system can be written as (cf. (1.2))

$$(2.4) \quad MPA\tilde{x} = MPb;$$

see [15, 27] for details. Hence, the two-level preconditioner corresponding to DEF is

$$(2.5) \quad \mathcal{P}_{\text{DEF}} = MP.$$

In order to derive the BNN and MG preconditioners, we consider the multiplicative combination of preconditioners. This combination can be explained by considering the stationary iterative methods induced by the preconditioner. Assuming that  $C_1$  and  $C_2$  are given matrices (preconditioners), we combine  $x_{j+\frac{1}{2}} := x_j + C_1(b - Ax_j)$  and  $x_{j+1} := x_{j+\frac{1}{2}} + C_2(b - Ax_{j+\frac{1}{2}})$  to obtain  $x_{j+1} = x_j + \mathcal{P}_{m_2}(b - Ax_j)$ , with

$$(2.6) \quad \mathcal{P}_{m_2} := C_1 + C_2 - C_2AC_1,$$

which is the multiplicative operator consisting of two preconditioners. In addition,  $C_1$  and  $C_2$  could be combined with another preconditioner,  $C_3$ , in a multiplicative way, yielding

$$(2.7) \quad \mathcal{P}_{m_3} = C_1 + C_2 + C_3 - C_2AC_1 - C_3AC_2 - C_3AC_1 + C_3AC_2AC_1.$$

If one chooses  $C_1 = Q$ ,  $C_2 = M$ , and  $C_3 = Q$  in (2.7), we obtain

$$(2.8) \quad \mathcal{P}_{\text{BNN}} = P^T MP + Q,$$

which is the two-level preconditioner corresponding to the abstract balancing Neumann-Neumann (BNN) method when  $M$  is SPD. In [21], it was shown that BNN has the same spectral properties as the 2L-PCG methods based on multigrid V(0,1)- and V(1,0)-cycle preconditioners. In this respect, multigrid V(0,1)- and V(1,0)-cycle preconditioners can be interpreted as “reduced” BNN preconditioners, since they use one fewer projection than the preconditioners in (2.8). In practice, BNN is always implemented based on these “reduced” preconditioners, so that the amount of work per iteration is comparable to that of DEF; see also [13, 22].

On the other hand, we could also use  $M$  twice instead of  $Q$ , i.e.,  $C_1 = M$ ,  $C_2 = Q$ , and  $C_3 = M^T$  in (2.7). The resulting two-level preconditioner, well known as the multigrid V(1,1)-cycle preconditioner, is then explicitly given by

$$(2.9) \quad \mathcal{P}_{\text{MG}} = M^T P + P^T M + Q - M^T P A M.$$

The latter expression for  $\mathcal{P}_{\text{MG}}$  also follows from the error-propagation operator

$$(2.10) \quad V := (I - \mathcal{P}_{\text{MG}} A) = (I - M^T A) P^T (I - M A),$$

which is often written as

$$(2.11) \quad V := S^* P^T S, \quad S := I - M A,$$

where  $S^*$  denotes the adjoint of  $S$  with respect to the  $A$ -inner product. Matrices  $S$  and  $S^*$  are known as the pre- and postsmoothers, respectively, and  $P^T$  is the MG coarse-grid correction operation. While the multigrid literature often distinguishes between this “two-grid” algorithm and true multigrid algorithms, as well as between standalone multigrid and multigrid used as a preconditioner, we will not; the resulting two-level PCG method with  $\mathcal{P}_{\text{MG}}$  is simply called MG in this paper. Here, no assumption on the symmetry of  $M$  is needed to guarantee that  $\mathcal{P}_{\text{MG}}$  is symmetric; definiteness will be examined in section 3.2. Moreover, notice that (2.9) is used only for the analysis of MG, but that the algorithm is never implemented using this explicit form, as the action of  $\mathcal{P}_{\text{MG}}$  can be computed using only a single multiplication with each of  $M$ ,  $M^T$ , and  $Q$ .

It can be observed that the two-level preconditioner corresponding to DEF is included as a term in the two-level MG preconditioner when the same choice of a symmetric  $M$  is used in both algorithms (cf. (2.5) and (2.9)). Hence, we might expect that MG is always more effective than DEF. For common choices of  $M$  and  $Z$ , this is indeed the case; see, e.g., section 7.2. However, it is not true in all cases; see section 4.

To summarize, the abbreviations and the two-level preconditioners corresponding to the proposed 2L-PCG methods are presented in Table 2.1.

TABLE 2.1  
List of two-level PCG methods which are compared in this paper.

| Name | Method                        | Two-level preconditioner, $\mathcal{P}$ |
|------|-------------------------------|---|
| PREC | Traditional Preconditioned CG | $M$                                     |
| DEF  | Deflation                     | $MP$                                    |
| BNN  | Abstract Balancing            | $P^T M P + Q$                           |
| MG   | Multigrid V(1,1)-cycle        | $M^T P + P^T M + Q - M^T P A M$         |

*Remark 2.3.* We emphasize that the parameters of the two-level PCG methods that will be compared can be arbitrary, so the comparison between these methods is based on their abstract versions. This means that the results of the comparison are valid for any SPD matrix  $A$ , full-rank matrix  $Z$ , and matrix  $M$  that satisfies the appropriate assumptions.

**3. Spectral properties of MG.** In this section, we present some results related to the spectral properties of the MG method. We first prove a result analogous to [16, Thm. 2.5], demonstrating that the MG preconditioner also clusters a number of eigenvalues at 1. Thereafter, we discuss necessary and sufficient conditions for the MG preconditioner to be SPD. Note that while these are natural concerns from a

preconditioning point of view, these questions are not commonly considered for MG methods, which are often applied as stationary iterations, in contrast to DEF, which is always used as a preconditioner.

First, we present some notation in Definition 3.1.

DEFINITION 3.1. *Let  $B \in \mathbb{R}^{n \times n}$  be an arbitrary matrix and  $\mathcal{S}$  be a subspace of  $\mathbb{R}^n$ . Then,*

- *the null space and column space of  $B$  are denoted by  $\mathcal{N}(B)$  and  $\mathcal{R}(B)$ , respectively;*
- *the spectrum of  $B$  is denoted by  $\sigma(B)$ ;*
- *$\dim \mathcal{S}$  denotes the dimension of  $\mathcal{S}$ ;*
- *if  $B$  is SPD, then the SPD square root of  $B$  is denoted by  $B^{1/2}$ ; and*
- *$B$  is called convergent in the  $A$ -norm (or  $A$ -norm convergent) if  $\|B\|_A < 1$ .*

**3.1. Unit eigenvalues of  $\mathcal{P}_{\text{MG}}A$ .** We recall that  $k$  is the dimension of the deflation subspace. In [16, 21], it was shown that DEF corresponds to a coefficient matrix that has exactly  $k$  zero eigenvalues, whereas the matrix associated with BNN has at least  $k$  unit eigenvalues. Theorem 3.2 shows that the matrix corresponding to MG also has at least  $k$  unit eigenvalues.

THEOREM 3.2. *Let  $\mathcal{P}_{\text{MG}}$  and  $S$  be as given in (2.9) and (2.11), respectively. Let  $m \geq 0$  be an integer satisfying*

$$(3.1) \quad \dim \mathcal{N}(S) = m.$$

*Then,  $\mathcal{P}_{\text{MG}}A$  has one as an eigenvalue, with geometric multiplicity at least  $k$  and at most  $k + 2m$ .*

*Proof.* In the following, we use the factorization of  $I - \mathcal{P}_{\text{MG}}A = S^*P^T S$  as given in (2.10) and (2.11). Note first that  $\dim \mathcal{N}(S^*) = \dim \mathcal{N}(S) = m$ ; see also [20, Lemma 3.2].

Considering (2.11), there are three ways for a vector,  $v \neq 0$ , to be in  $\mathcal{N}(I - \mathcal{P}_{\text{MG}}A)$ :

- (i)  $v \in \mathcal{N}(S)$ , so that  $Sv = 0$ ;
- (ii)  $Sv \in \mathcal{N}(P^T)$ , yielding  $P^T Sv = 0$ ;
- (iii)  $P^T Sv \in \mathcal{N}(S^*)$ , so that  $S^*P^T Sv = 0$ .

We treat each case separately.

(i) The geometric multiplicity of the zero eigenvalue of  $I - \mathcal{P}_{\text{MG}}A$  must be at least  $m$ , due to (3.1). This accounts exactly for all contributions to  $\mathcal{N}(I - \mathcal{P}_{\text{MG}}A)$  from null space vectors of the first type.

(ii) Counting the geometric multiplicity of vectors of the second type is only slightly more complicated. The fundamental theorem of linear algebra (see, e.g., [19]) gives an orthogonal decomposition of  $\mathbb{R}^n$  as

$$(3.2) \quad \mathbb{R}^n = \mathcal{R}(S) \oplus \mathcal{N}(S^T).$$

Since  $\dim \mathcal{R}(S) = n - m$ , it must be the case that

$$(3.3) \quad \dim \mathcal{N}(S^T) = m.$$

Now, consider the intersection of  $\mathcal{R}(Z)$  with subspaces  $\mathcal{R}(S)$  and  $\mathcal{N}(S^T)$ :

$$\mathcal{Z}_1 := \mathcal{R}(Z) \cap \mathcal{R}(S), \quad \mathcal{Z}_2 := \mathcal{R}(Z) \cap \mathcal{N}(S^T),$$

and let  $\dim \mathcal{Z}_1 = k_1$  and  $\dim \mathcal{Z}_2 = k_2$ . Note that necessarily  $k_1 + k_2 = k$  since  $Z \in \mathbb{R}^{n \times k}$ , and that  $k_2$  is no bigger than  $m$ , because of (3.3). Since  $\mathcal{N}(P^T) = \mathcal{R}(Z)$ , we

have  $\mathcal{N}(P^T S) \setminus \mathcal{N}(S) = \mathcal{R}(Z) \cap \mathcal{R}(S) = \mathcal{Z}_1$ . This results in  $\dim \mathcal{N}(P^T S) \setminus \mathcal{N}(S) = k_1$ , which is the contribution to the dimension of the null space by vectors of the second type. Since  $k_1 + k_2 = k$  for  $k_2 \leq m$ , the total dimension of the null space arising from vectors of the first and second types must satisfy  $k \leq k_1 + m \leq k + m$ .

(iii) Similarly, we can determine the dimension of the null space of the third type. Note first that (cf. (3.2))

$$\mathbb{R}^n = \mathcal{R}(P^T S) \oplus \mathcal{N}(S^T P).$$

Let  $\mathcal{M} := \mathcal{N}(S^*)$  and  $\mathcal{M}_1 = \mathcal{M} \cap \mathcal{R}(P^T S)$ . Then, the number of unit eigenvalues of the third type is

$$m_1 = \dim \mathcal{M}_1 \leq \dim \mathcal{M} = m.$$

Thus,  $\dim \mathcal{N}(\mathcal{P}_{\text{MG}} A) = m + k_1 + m_1$ , which can be bounded by

$$k \leq m + k_1 + m_1 \leq k + 2m.$$

Since counting the geometric multiplicity of zero eigenvalues of  $I - \mathcal{P}_{\text{MG}} A$  is trivially equal to counting the geometric multiplicity of unit eigenvalues of  $\mathcal{P}_{\text{MG}} A$ , the proof is complete.  $\square$

*Remark 3.3.*

- $\mathcal{P}_{\text{MG}} A$  has at least  $k$  unit eigenvalues, even if  $S$  is singular.
- If zero is not an eigenvalue of  $S$ , then it is also not an eigenvalue of  $S^*$  (which is similar to  $S^T$ ). Thus, Theorem 3.2 then says that  $\mathcal{P}_{\text{MG}} A$  has exactly  $k$  unit eigenvalues in this case.
- Since  $M$  may be nonsymmetric, the geometric and algebraic multiplicities of the zero eigenvalue of  $S$  (or, equivalently, the unit eigenvalues of  $MA$ ) should be distinguished, since they might differ.<sup>1</sup>
- In a manner similar to that in Theorem 3.2, it can be shown that  $\mathcal{P}_{\text{BNN}} A$  has at least  $k$  and at most  $2k + m$  unit eigenvalues.

**3.2. Positive definiteness of  $\mathcal{P}_{\text{MG}}$ .** A 2L-PCG method is guaranteed to converge if  $\mathcal{P}$ , as given in (1.3), is SPD or can be transformed into an SPD matrix; see, e.g., [5] for more details. This is certainly satisfied for BNN and DEF when  $M$  is SPD; see [21]. Here, we examine this issue for MG. It is obvious that, while  $\mathcal{P}_{\text{MG}}$  is always symmetric, it (and, therefore, also  $\mathcal{P}_{\text{MG}} A$ ) is not positive definite for all choices of  $Z$  and  $M$ , as in the next example.

*Example 3.4.* Suppose that  $M = I$  and  $Z = [v_1 \cdots v_k]$ , where  $\{v_i\}$  is the set of orthonormal eigenvectors corresponding to eigenvalues  $\{\lambda_i\}$  of  $A$ . We assume that all  $\lambda_i$  (and, correspondingly, the  $v_i$ ) are ordered so that  $\lambda_{k+1} \leq \lambda_j \leq \lambda_n$  for all  $j$  such that  $k < j \leq n$ . Eigenvalues  $\lambda_1$  to  $\lambda_k$  are not assumed to be ordered and are not necessarily the smallest eigenvalues of  $A$ . Then,

$$(3.4) \quad \mathcal{P}_{\text{MG}} = P + P^T + Q - PA = 2I - 2ZZ^T + Z\Lambda^{-1}Z^T - A + ZZ^T A,$$

where  $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_k)$ . Multiplying (3.4) by  $v_i$  gives us

$$\mathcal{P}_{\text{MG}} v_i = 2v_i - 2ZZ^T v_i + Z\Lambda^{-1}Z^T v_i - \lambda_i v_i + \lambda_i ZZ^T v_i.$$

<sup>1</sup>A simple example is Gauss–Seidel for the one-dimensional (1D) Poisson problem with homogeneous Dirichlet boundary conditions. Take  $A = \text{tridiag}(-1, 2, -1)$  and  $M^{-1}$  to be the lower-triangular part of  $A$ . Then,  $S$  has eigenvalue 0 with algebraic multiplicity  $\frac{n}{2}$ , assuming that  $n$  is even. Since there is only one eigenvector corresponding to this eigenvalue, the geometric multiplicity is 1.



This implies

$$(3.5) \quad \mathcal{P}_{\text{MG}}v_i = \begin{cases} \frac{1}{\lambda_i}v_i & \text{for } i = 1, \dots, k, \\ (2 - \lambda_i)v_i & \text{for } i = k + 1, \dots, n. \end{cases}$$

Hence, if  $A$  has eigenvalues  $\{\lambda_i\}$ , then the spectrum of  $\mathcal{P}_{\text{MG}}$  is given by

$$\left\{ \frac{1}{\lambda_1}, \dots, \frac{1}{\lambda_k}, 2 - \lambda_{k+1}, \dots, 2 - \lambda_n \right\}.$$

In this case,  $\mathcal{P}_{\text{MG}}$  is positive definite if and only if  $\lambda_n < 2$ . Furthermore, since  $\mathcal{P}_{\text{MG}}$  is always symmetric by construction,  $\mathcal{P}_{\text{MG}}$  is SPD if and only if  $\lambda_n < 2$ .

Example 3.4 shows that  $\mathcal{P}_{\text{MG}}$  can be indefinite for some choices of  $Z$  and  $M$ . This highlights an important difference between MG and DEF. Indeed, many preconditioners,  $M$ , that make sense with DEF lead to indefinite  $\mathcal{P}_{\text{MG}}$ , while choices of  $M$  that lead to SPD  $\mathcal{P}_{\text{MG}}$  might give nonsymmetric operators for  $\mathcal{P}_{\text{DEF}}$ .

**THEOREM 3.5.** *Let  $M \in \mathbb{R}^{n \times n}$  and  $Z \in \mathbb{R}^{n \times k}$  be as defined in Definition 2.1. Let  $\mathcal{P}_{\text{MG}}$  be as given in (2.9). A necessary and sufficient condition for  $\mathcal{P}_{\text{MG}}$  to be SPD is that  $Z$  and  $M$  satisfy*

$$(3.6) \quad \min_{\substack{w: w \perp AZ \\ w \neq 0}} w^T (M + M^T - MAM^T) w > 0.$$

*Proof.* By definition,  $\mathcal{P}_{\text{MG}}$  is positive definite if and only if  $u^T \mathcal{P}_{\text{MG}} u > 0$  for all vectors  $u \neq 0$ . Taking  $u := A^{1/2}y$ , this means that  $\mathcal{P}_{\text{MG}}$  is SPD if and only if  $y^T A^{1/2} \mathcal{P}_{\text{MG}} A^{1/2} y > 0$  for all  $y$ , or that  $A^{1/2} \mathcal{P}_{\text{MG}} A^{1/2}$  is positive definite. Moreover,  $A^{1/2} \mathcal{P}_{\text{MG}} A^{1/2}$  is symmetric, and, so, it is SPD if and only if its smallest eigenvalue is greater than 0. This, in turn, is equivalent to requiring that  $I - A^{1/2} \mathcal{P}_{\text{MG}} A^{1/2}$  has largest eigenvalue less than 1. But  $I - A^{1/2} \mathcal{P}_{\text{MG}} A^{1/2}$  is a similarity transformation of  $V$  (see (2.10)),  $A^{1/2} V A^{-1/2} = I - A^{1/2} \mathcal{P}_{\text{MG}} A^{1/2}$ , which can be written as  $A^{1/2} V A^{-1/2} = (R\tilde{S})^T (R\tilde{S})$ , for

$$R := I - A^{1/2} Q A^{1/2}, \quad \tilde{S} := I - A^{1/2} M A^{1/2}.$$

Note that the eigenvalues of  $(R\tilde{S})^T (R\tilde{S})$  are the singular values squared of  $R\tilde{S}$  (see, e.g., [8]), which are also the eigenvalues of  $(R\tilde{S})(R\tilde{S})^T = R\tilde{S}\tilde{S}^T R$ . So, the largest eigenvalue of  $A^{1/2} V A^{-1/2}$  is less than 1 if and only if the largest eigenvalue of  $R\tilde{S}\tilde{S}^T R$  is less than 1. This happens if and only if

$$(3.7) \quad \frac{u^T R(\tilde{S}\tilde{S}^T)Ru}{u^T u} < 1 \quad \forall u \neq 0.$$

To maximize this ratio, we write  $u = A^{1/2}Zy_1 + Ry_2$  and note that  $R$  is the  $L_2$ -orthogonal projection onto the orthogonal complement of the range of  $A^{1/2}Z$ . Then,

$$u^T R(\tilde{S}\tilde{S}^T)Ru = y_2^T R(\tilde{S}\tilde{S}^T)Ry_2, \quad u^T u = y_1^T Z^T AZy_1 + y_2^T R^2 y_2.$$

So, maximizing the ratio over all choices of  $y_1$  means choosing  $y_1 = 0$  so that the denominator of (3.7) is as small as possible. Therefore,

$$(3.8) \quad \frac{u^T R\tilde{S}\tilde{S}^T Ru}{u^T u} < 1 \quad \forall u \neq 0 \quad \Leftrightarrow \quad \frac{y_2^T R\tilde{S}\tilde{S}^T Ry_2}{y_2^T R^2 y_2} < 1 \quad \forall y_2 \neq 0.$$

Thus, if the ratio on the right of (3.8) is bounded below 1 for all  $y_2$ , so must be the ratio in (3.7). But, if the ratio in (3.7) is bounded below 1 for all  $u$ , then it is bounded for  $u = Ry_2$ , since  $R$  is a projection, and thus  $R^2 = R$ . This gives the bound at the right-hand side of (3.8).

Equivalently, we can maximize the ratio over  $\mathcal{R}(R) = \mathcal{R}(A^{1/2}Z)^\perp$  (i.e., the orthogonal complement of the range of  $A^{1/2}Z$ ). So, the largest eigenvalue of  $R\tilde{S}\tilde{S}^T R$  is less than 1 if and only if

$$(3.9) \quad \max_{\substack{x: x \perp A^{1/2}Zy \forall y \\ x \neq 0}} \frac{x^T \tilde{S}\tilde{S}^T x}{x^T x} < 1.$$

By computation, we have  $\tilde{S}\tilde{S}^T = I - A^{1/2}(M + M^T - MAM^T)A^{1/2}$ . Therefore, the bound (3.9) is equivalent to requiring

$$\min_{\substack{x: x \perp A^{1/2}Zy \forall y \\ x \neq 0}} \frac{x^T A^{1/2}(M + M^T - MAM^T)A^{1/2}x}{x^T x} > 0.$$

Taking  $w = A^{1/2}x$ , this is, in turn, equivalent to

$$\min_{\substack{w: w \perp AZy \forall y \\ w \neq 0}} w^T (M + M^T - MAM^T) w > 0,$$

because  $w^T A^{-1}w > 0$  for all  $w$ .  $\square$

Intuitively, we expect the spectral properties of  $\mathcal{P}_{\text{MG}}$  to reflect those of  $M$ , with some account for the effect of the coarse-grid correction. Inequality (3.6) is particularly interesting in comparison with Theorem 3.6. This theorem gives a necessary and sufficient condition for  $M$  to define a convergent smoother; see also [7, 30].

**THEOREM 3.6.** *Let  $M \in \mathbb{R}^{n \times n}$ , and let  $S$  be as given in (2.11). A necessary and sufficient condition for  $S$  to be convergent in the  $A$ -norm is*

$$(3.10) \quad \min_{w \neq 0} w^T (M + M^T - MAM^T) w > 0.$$

*Proof.* See [7, 30].  $\square$

Note that when  $M$  is invertible, Theorem 3.6 amounts to the condition

$$\|S\|_A < 1 \Leftrightarrow \lambda_{\min}(M^{-1} + M^{-T} - A) > 0,$$

which can also be found, for example, in [30, Thm. 5.3]. On the other hand, Theorem 3.5 gives

$$\min_{\substack{w: w \perp AZy \forall y \\ w \neq 0}} w^T \tilde{M} w > 0 \Leftrightarrow \min_{\substack{v: v = M^T w, w \perp AZy \forall y \\ v \neq 0}} v^T (M^{-1} + M^{-T} - A) v > 0,$$

where

$$(3.11) \quad \tilde{M} := M + M^T - MAM^T.$$

Necessarily,

$$\min_{\substack{w: w \perp AZy \forall y \\ w \neq 0}} w^T \tilde{M} w \geq \min_{w \neq 0} w^T \tilde{M} w,$$

so the condition for  $\mathcal{P}_{\text{MG}}$  to be SPD is weaker than the condition for a convergent  $S$  in the  $A$ -norm. In other words, the  $A$ -norm convergence of  $S$  implies both convergence of  $I - \mathcal{P}_{\text{MG}}A$  and that  $\mathcal{P}_{\text{MG}}$  is SPD. However,  $\mathcal{P}_{\text{MG}}$  can be SPD even if  $\|S\|_A \geq 1$ , as long as coarse-grid correction effectively treats amplified modes.

**4. Comparing MG and DEF for a particular choice of  $M$  and  $Z$ .** In this section, we show that abstract preconditioners in the MG framework do not always lead to better conditioned systems than DEF (although our example does not reflect a practical parameter choice, particularly from the point of view of MG). Such problems can even be found in the case of  $M = I$ ; see [20, sect. 4]. Here, we show that this can be generalized for an arbitrary SPD  $M$  used in both DEF and MG when  $Z$  consists of eigenvectors of  $MA$ . We start with some spectral bounds on MG and DEF under these assumptions. Thereafter, we perform a comparison between the condition numbers for MG and DEF. Note that there is no assumption on the ordering of eigenvalue and eigenvector sets presented below.

**THEOREM 4.1.** *Suppose that  $M$  is given and  $\{\lambda_i\}$  is the set of eigenvalues of  $MA$  with corresponding eigenvectors  $\{v_i\}$ . Let the columns of  $Z$  be given by  $v_1, \dots, v_k$ . Let  $\mathcal{P}_{\text{DEF}}$  and  $\mathcal{P}_{\text{MG}}$  be as given in (2.5) and (2.9), respectively. Then,*

(i)  $\mathcal{P}_{\text{MG}}A$  has the following eigenvalues:

$$(4.1) \quad \begin{cases} 1 & \text{for } i = 1, \dots, k, \\ \lambda_i(2 - \lambda_i) & \text{for } i = k + 1, \dots, n; \end{cases}$$

(ii)  $\mathcal{P}_{\text{DEF}}A$  has the following eigenvalues:

$$(4.2) \quad \begin{cases} 0 & \text{for } i = 1, \dots, k, \\ \lambda_i & \text{for } i = k + 1, \dots, n. \end{cases}$$

*Proof.* The proof follows from [4, Prop. 2] and [27, sect. 4]. □

If we make the added assumption that  $M$  is SPD (so that  $\mathcal{P}_{\text{DEF}}$  is also SPD) and that  $0 < \lambda_j < 2$  for  $k < j \leq n$  (so that  $\mathcal{P}_{\text{MG}}$  is SPD), then we can compare DEF and MG based on their condition numbers, denoted by  $\kappa_{\text{DEF}}$  and  $\kappa_{\text{MG}}$ , respectively. If we order the remaining eigenvalues so that  $\lambda_{k+1} \leq \lambda_j \leq \lambda_n$  for  $k < j \leq n$ , the resulting expressions for  $\kappa_{\text{MG}}$  and  $\kappa_{\text{DEF}}$  depend only on eigenvalues  $\lambda_{k+1}$  and  $\lambda_n$  of  $MA$ :

$$(4.3) \quad \kappa_{\text{MG}} = \frac{1}{\min\{\lambda_{k+1}(2 - \lambda_{k+1}), \lambda_n(2 - \lambda_n)\}}, \quad \kappa_{\text{DEF}} = \frac{\lambda_n}{\lambda_{k+1}}$$

(under the assumption that  $Z$  consists of eigenvectors of  $MA$ ). So, for some choices of  $Z$  and  $M$ , MG yields a larger condition number than DEF; see [20] for some concrete examples.

Using this result, we can graph the region of space parametrized by  $\lambda_{k+1}$  and  $\lambda_n$  where each method is preferable; see Figure 4.1. Note first that if  $\lambda_{k+1} = \lambda_n$ , then  $\sigma(\mathcal{P}_{\text{MG}}A)$  consists of at most two different eigenvalues, 1 and  $\lambda_n(2 - \lambda_n)$ . Also, if  $\lambda_{k+1} = 2 - \lambda_n$ , then  $\kappa_{\text{MG}} = [\lambda_{k+1}(2 - \lambda_{k+1})]^{-1} = [\lambda_n(2 - \lambda_n)]^{-1}$ , while  $\kappa_{\text{DEF}} = \frac{\lambda_n}{2 - \lambda_n}$ . In general, the region  $0 < \lambda_{k+1} \leq \lambda_n \leq 2$  is naturally partitioned into two subdomains along the line where  $\lambda_{k+1}(2 - \lambda_{k+1}) = \lambda_n(2 - \lambda_n)$ , which occurs when  $\lambda_{k+1} = 2 - \lambda_n$ :

- if  $\lambda_{k+1}(2 - \lambda_{k+1}) \leq \lambda_n(2 - \lambda_n)$ , then  $\kappa_{\text{MG}} = [\lambda_{k+1}(2 - \lambda_{k+1})]^{-1}$ . Thus,  $\kappa_{\text{MG}} < \kappa_{\text{DEF}}$  if and only if

$$\lambda_{k+1} < 2 - \frac{1}{\lambda_n};$$

- if  $\lambda_{k+1}(2 - \lambda_{k+1}) \geq \lambda_n(2 - \lambda_n)$ , then  $\kappa_{\text{MG}} = [\lambda_n(2 - \lambda_n)]^{-1}$ . Thus,  $\kappa_{\text{MG}} < \kappa_{\text{DEF}}$  if and only if

$$\lambda_{k+1} < \lambda_n^2(2 - \lambda_n).$$

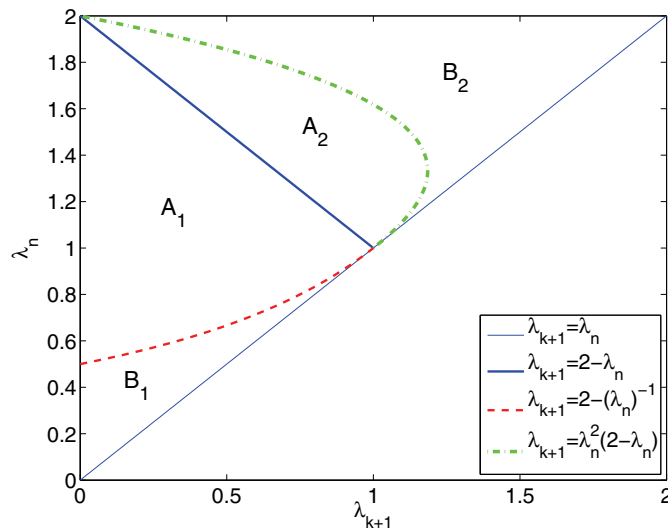


FIG. 4.1. Partitioning of region  $0 < \lambda_{k+1} \leq \lambda_n < 2$ . In regions  $A_1$  and  $A_2$ ,  $\kappa_{\text{MG}} < \kappa_{\text{DEF}}$ , while in regions  $B_1$  and  $B_2$ ,  $\kappa_{\text{DEF}} < \kappa_{\text{MG}}$ . Here, we assume that  $M$  is SPD and that  $Z$  consists of  $k$  eigenvectors of  $MA$ . The two condition numbers are equal (i.e.,  $\kappa_{\text{MG}} = \kappa_{\text{DEF}}$ ) along the red (dotted) and green (dotted-dashed) lines.

Figure 4.1 depicts these regions graphically. For any given  $\lambda_{k+1}$  and  $\lambda_n$ , the method with smallest condition number follows immediately from this figure. Example 4.2 gives some consequences of Figure 4.1.

*Example 4.2.*

(a) If  $\sigma(MA) \subseteq (0, 0.5]$ , then  $(\lambda_{k+1}, \lambda_n)$  falls within Region  $B_1$  and, hence,  $\kappa_{\text{DEF}} < \kappa_{\text{MG}}$ .

(b) If  $\sigma(MA) \subseteq (0, 2)$  with  $\lambda_{k+1} \approx 2 - \lambda_n$ , then  $(\lambda_{k+1}, \lambda_n)$  falls within either Region  $A_1$  or Region  $A_2$  and  $\kappa_{\text{DEF}} > \kappa_{\text{MG}}$  holds.

Case (a) says that if  $M$  is a “bad” smoother (no eigenvalues of  $S$  are less than  $\frac{1}{2}$ ), then MG is expected to require more iterations to converge than DEF does. On the other hand, case (b) implies that if  $M$  is a “good” smoother (all eigenvalues that need to be handled by relaxation are done so with eigenvalues of  $S$  bounded in a neighborhood of the origin), then MG should converge in fewer iterations than DEF.

**5. Effect of relaxation parameters.** While DEF may have a smaller condition number than MG for some choices of  $M$  and  $Z$ , MG methods often make use of an added, relaxation, parameter that can be very important in leading to an efficient algorithm. We illustrate this here by considering  $M = \alpha I$  for an optimized choice of  $\alpha$ . Such a choice of relaxation scheme within MG is commonly known as Richardson relaxation.

**5.1. Analysis of scaling relaxation.** Instead of considering the original linear system (1.1), we could consider preconditioning for the scaled linear system

$$(5.1) \quad \alpha Ax = \alpha b, \quad \alpha > 0,$$

for a given choice of  $M$ . A subscript,  $\alpha$ , will be added to the notation for operators and matrices, if they are for (5.1). So,  $\mathcal{P}_{\text{DEF}, \alpha}$  and  $\mathcal{P}_{\text{MG}, \alpha}$  denote the deflation matrix

and MG-preconditioner based on (5.1), respectively:

$$\begin{aligned}
 \mathcal{P}_{\text{DEF},\alpha} A &= M(I - \alpha AZ(Z^T(\alpha A)Z)^{-1}Z^T)(\alpha A) \\
 &= \alpha M(I - AZ(Z^T AZ)^{-1}Z^T)A; \\
 (5.2) \quad \mathcal{P}_{\text{MG},\alpha} A &= I - (I - M^T(\alpha A))(I - Z(Z^T(\alpha A)Z)^{-1}Z^T(\alpha A))(I - M(\alpha A)) \\
 &= I - (I - \alpha M^T A)(I - Z(Z^T AZ)^{-1}Z^T A)(I - \alpha MA).
 \end{aligned}$$

Solving the scaled linear system (5.1) for a given choice of  $M$  is, then, equivalent to solving the preconditioned linear system (1.2) with preconditioner  $\alpha M$ , as  $(I - \alpha AZ(Z^T(\alpha A)Z)^{-1}Z^T) = (I - AZ(Z^T AZ)^{-1}Z^T)$ . The parameter,  $\alpha$ , can then be regarded as a parameter of the relaxation only instead of the linear system (i.e., the relaxation processes are rescaled, but there is no net effect on coarse-grid correction). Therefore, DEF is scaling invariant:

$$\kappa_{\text{DEF},\alpha} = \frac{\lambda_n(MP\alpha A)}{\lambda_{k+1}(MP\alpha A)} = \frac{\lambda_n(MPA)}{\lambda_{k+1}(MPA)} = \kappa_{\text{DEF}}.$$

In contrast, MG is not scaling invariant, and the positive-definiteness property of  $\mathcal{P}_{\text{MG},\alpha}$  depends strongly on  $\alpha$ ; it is well known that Richardson relaxation (where  $M = \alpha I$ ) is convergent if

$$(5.3) \quad 0 < \alpha < \frac{2}{\|A\|_2};$$

see, e.g., [30]. For multigrid, a typical choice of  $\alpha$  is close to  $\frac{1}{\|A\|_2}$ , which guarantees that the slow-to-converge modes of relaxation are only those associated with the small eigenvalues of  $A$  [2, 11]. A better choice of  $\alpha$  is possible if we make further assumptions on how the eigenvectors of  $A$  associated with small eigenvalues are treated by coarse-grid correction. Indeed, it is possible to get an explicit expression for the optimal  $\alpha$  under such assumptions; see the next subsection.

**5.2. Optimal choice of  $\alpha$ .** The best value of  $\alpha$  depends on  $M$  and  $Z$ , so the optimal  $\alpha$ , denoted as  $\alpha_{\text{opt}}$ , can be determined only if these are fixed. In this case, the job of relaxation is specifically to reduce errors that are conjugate to the range of  $Z$ . In this section, we consider  $M = \alpha I$  and the columns of  $Z$  to be given by eigenvectors of  $A$ . The best choice of  $\alpha$  is then the one that minimizes the “spectral radius” of relaxation over the complement of the range of interpolation, i.e.,

$$\min_{\substack{x,y^T Z^T A x = 0 \ \forall y \\ x \neq 0}} \frac{|x^T (I - \alpha A)x|}{x^T x}.$$

This  $\alpha$  is determined explicitly in Theorem 5.1. In this theorem, we make only a weak assumption on the ordering of the eigenvalue and eigenvector sets; any combination of eigenvectors may be used to define the deflation-subspace matrix,  $Z$ , and the remaining eigenvalues and eigenvectors are assumed to be ordered so that  $\lambda_{k+1} \leq \lambda_j \leq \lambda_n$  for  $k < j \leq n$ .

**THEOREM 5.1.** *Suppose that  $M = \alpha I$  and  $Z = [v_1 \cdots v_k]$ , where  $\{v_i\}$  is the set of orthonormal eigenvectors corresponding to eigenvalues  $\{\lambda_i\}$  of  $A$ . We assume that all  $\lambda_i$  (and, correspondingly, the  $v_i$ ) are ordered so that  $\lambda_{k+1} \leq \lambda_j \leq \lambda_n$  for all  $j$  such*

that  $k < j \leq n$ . Moreover, let  $\mathcal{P}_{\text{MG},\alpha}$  be as given in (5.2) such that  $\mathcal{P}_{\text{MG},\alpha}A$  is SPD. Then,  $\kappa(\mathcal{P}_{\text{MG},\alpha}A)$  is minimized for

$$(5.4) \quad \alpha_{\text{opt}} = \frac{2}{\lambda_{k+1} + \lambda_n}.$$

*Proof.* Note first that, by choosing  $M = \alpha I$ , the error-propagation operator for MG,  $V$ , can be written as (cf. (3.4))

$$V = I - \mathcal{P}_{\text{MG}}A = (I - \alpha A)P^T(I - \alpha A) = 2\alpha I + Z\Lambda^{-1}Z^T - 2\alpha ZZ^T - \alpha^2 A + \alpha^2 Z\Lambda Z^T.$$

So, applying  $\mathcal{P}_{\text{MG}}$  to an eigenvector,  $v_i$ , of  $A$  gives (cf. (3.5))

$$\mathcal{P}_{\text{MG}}v_i = \begin{cases} \frac{1}{\lambda_i}v_i & \text{for } i = 1, \dots, k, \\ \alpha(2 - \alpha\lambda_i)v_i & \text{for } i = k + 1, \dots, n. \end{cases}$$

Thus,  $\mathcal{P}_{\text{MG}}A$  has eigenvalue 1 with algebraic multiplicity  $k$ , and  $n - k$  eigenvalues of the form  $\alpha\lambda_i(2 - \alpha\lambda_i)$  for  $i = k + 1, \dots, n$ .

Let  $\{\sigma_i\}$  be the set of eigenvalues of  $\mathcal{P}_{\text{MG}}A$ , which are positive and sorted increasingly, so that its condition number is given by  $\frac{\sigma_n}{\sigma_1}$ . By assumption,  $\alpha\lambda_i(2 - \alpha\lambda_i) > 0$  for all  $i = k + 1, \dots, n$ , and, by calculation,  $\alpha\lambda_i(2 - \alpha\lambda_i) \leq 1$  for all  $\alpha$  and  $\lambda_i$ . Thus,

$$\sigma_1 = \min_{i \in [k+1, n]} \{\alpha\lambda_i(2 - \alpha\lambda_i)\}, \quad \sigma_n = 1.$$

Since the function  $f(\lambda) := \alpha\lambda(2 - \alpha\lambda)$  is concave down,

$$(5.5) \quad \min_{i \in [k+1, n]} \{\alpha\lambda_i(2 - \alpha\lambda_i)\} = \min \{\alpha\lambda_{k+1}(2 - \alpha\lambda_{k+1}), \alpha\lambda_n(2 - \alpha\lambda_n)\}.$$

Subsequently, we choose  $\alpha_{\text{opt}}$  to maximize this minimum eigenvalue,

$$\max_{\alpha} \min \{\alpha\lambda_{k+1}(2 - \alpha\lambda_{k+1}), \alpha\lambda_n(2 - \alpha\lambda_n)\}.$$

This is achieved when

$$\alpha\lambda_{k+1}(2 - \alpha\lambda_{k+1}) = \alpha\lambda_n(2 - \alpha\lambda_n),$$

i.e., for  $\alpha = \frac{2}{\lambda_{k+1} + \lambda_n}$ .  $\square$

**COROLLARY 5.2.** *Let the conditions of Theorem 5.1 be satisfied. Then,  $\kappa_{\text{MG},\alpha_{\text{opt}}} \leq \kappa_{\text{DEF}}$ .*

*Proof.* If the optimal weighting parameter,  $\alpha_{\text{opt}}$ , is substituted into (5.5), then the smallest eigenvalue of  $\mathcal{P}_{\text{MG},\alpha_{\text{opt}}}A$  is equal to

$$(5.6) \quad \frac{4\lambda_{k+1}\lambda_n}{(\lambda_{k+1} + \lambda_n)^2}.$$

As a consequence, the condition number of  $\mathcal{P}_{\text{MG},\alpha_{\text{opt}}}A$  is then given by

$$(5.7) \quad \kappa_{\text{MG},\alpha_{\text{opt}}} = \frac{(\lambda_{k+1} + \lambda_n)^2}{4\lambda_{k+1}\lambda_n}.$$

Finally,  $\kappa_{\text{MG},\alpha_{\text{opt}}} \leq \kappa_{\text{DEF}}$  follows from the fact that

$$\frac{(\lambda_{k+1} + \lambda_n)^2}{4\lambda_{k+1}\lambda_n} \leq \frac{\lambda_n}{\lambda_{k+1}} \Leftrightarrow (\lambda_{k+1} + \lambda_n)^2 \leq (2\lambda_n)^2,$$

which is always true, since  $\lambda_{k+1} \leq \lambda_n$ .  $\square$

*Remark 5.3.*

- Using this weighting, the condition numbers corresponding to MG and DEF are the same if the spectrum of  $A$  is “flat” (i.e., if  $\lambda_{k+1} = \lambda_n$ ). In all other cases, using the optimized parameter,  $\alpha_{\text{opt}}$ , in MG leads to a more favorable condition number than DEF.
- In section 4, it has been shown that it is possible to have  $\kappa_{\text{MG}} > \kappa_{\text{DEF}}$ . However, Theorem 5.1 shows that these examples can never be constructed if an optimal relaxation parameter is used.
- In practice, approximations to  $\alpha$  are fairly easy to compute, although the exact eigenvalue distribution is usually unknown. Gershgorin’s theorem gives us estimates of both  $\lambda_1$  and  $\lambda_n$  which can be used to approximate  $\lambda_{k+1}$ . A naive approximation can be made by assuming that the spectrum is dense between  $\lambda_1$  and  $\lambda_n$  and that  $\lambda_{k+1}$  is simply a linear function of  $k$ :  $\lambda_{k+1} = \lambda_1 + \frac{k}{n-1}(\lambda_n - \lambda_1)$ .
- An optimal weighting parameter,  $\alpha_{\text{opt}}$ , can also be considered for general preconditioners,  $M$ ; however, it is often much more difficult to express  $\alpha_{\text{opt}}$  explicitly, as it depends on the spectral properties of  $MA$ , which may not be known. In general, the optimal choice of  $\alpha$  is such that relaxation converges as quickly as possible on the modes that are not being treated by the coarse-grid correction phase. Thus, if the spectral picture of  $MA$  is known well enough to approximate the eigenvalues corresponding to  $\lambda_{k+1}$  and  $\lambda_n$ , a choice of  $\alpha_{\text{opt}}$  similar to that in (5.4) may be possible.

**6. Symmetrizing the smoother.** In the previous section, we have seen that MG can be expected to converge in fewer iterations than DEF for specific choices of  $M$  and  $Z$ , when the same components are used in both MG and DEF. However, the fact that MG requires fewer iterations than DEF for many preconditioners does not necessarily mean that it is more efficient, since each iteration of MG is more expensive, due to the use of two smoothing steps. In order to make a fairer comparison between DEF and MG, we now consider DEF using the preconditioning version of the symmetrized smoother:

$$(6.1) \quad SS^* = (I - MA)(I - M^T A) = I - \widetilde{M}A,$$

with

$$(6.2) \quad \widetilde{M} := M + M^T - MAM^T.$$

Note that  $\widetilde{M}$ , as defined here, is the same as in (3.11). We then use  $\widetilde{M}$  as the preconditioner in DEF, since this choice allows implementation in such a way that each iteration of BNN, DEF, and MG has similar cost. In this section, the spectra of MG, BNN, and DEF are compared, using  $\widetilde{M}$  in DEF and BNN and  $M$  in MG. For general  $Z$  and  $\widetilde{M}$  such that  $\widetilde{M}$  is SPD, we show that BNN and DEF, both with preconditioner  $\widetilde{M}$ , and MG with smoother  $M$  yield the same eigenvalues for those modes that are not treated by the coarse-grid correction. This statement is completely independent of the choices of  $M$  and  $Z$  and, as such, is more general than similar results that have appeared previously (e.g., that of [29]).

**THEOREM 6.1.** *Let  $M \in \mathbb{R}^{n \times n}$  be as given in (2.1) such that  $\mathcal{P}_{\text{MG}}$  with smoother  $M$  is SPD. In addition, let  $\widetilde{M}$  be as defined in (6.2) such that  $\mathcal{P}_{\text{BNN}}$  with preconditioner  $\widetilde{M}$  is SPD. Then, the eigenvalues of  $\mathcal{P}_{\text{MG}}A$  and  $\mathcal{P}_{\text{BNN}}A$  are equal.*

*Proof.* We show the equivalence of MG and BNN by examining the eigenvalues of their error-propagation forms,

$$\begin{cases} I - \mathcal{P}_{\text{MG}}A &= S^*P^T S, \\ I - \mathcal{P}_{\text{BNN}}A &= P^T(I - \widetilde{M}A)P^T. \end{cases}$$

We examine both methods by making the same similarity transformation,

$$I - \mathcal{P}A \rightarrow A^{\frac{1}{2}}(I - \mathcal{P}A)A^{-\frac{1}{2}},$$

which allows us to make use of the fact that  $I - A^{\frac{1}{2}}QA^{\frac{1}{2}}$  is an orthogonal projection in the  $L_2$ -inner product. Computing the similarity transformed systems, we have

$$\begin{cases} A^{\frac{1}{2}}(I - \mathcal{P}_{\text{MG}}A)A^{-\frac{1}{2}} &= (I - A^{\frac{1}{2}}M^T A^{\frac{1}{2}})(I - A^{\frac{1}{2}}QA^{\frac{1}{2}})(I - A^{\frac{1}{2}}MA^{\frac{1}{2}}), \\ A^{\frac{1}{2}}(I - \mathcal{P}_{\text{BNN}}A)A^{-\frac{1}{2}} &= (I - A^{\frac{1}{2}}QA^{\frac{1}{2}})(I - A^{\frac{1}{2}}\widetilde{M}A^{\frac{1}{2}})(I - A^{\frac{1}{2}}QA^{\frac{1}{2}}). \end{cases}$$

By defining  $C := (I - A^{\frac{1}{2}}QA^{\frac{1}{2}})(I - A^{\frac{1}{2}}MA^{\frac{1}{2}})$ , we can rewrite the latter expressions as

$$\begin{cases} A^{\frac{1}{2}}(I - \mathcal{P}_{\text{MG}}A)A^{-\frac{1}{2}} &= C^T C, \\ A^{\frac{1}{2}}(I - \mathcal{P}_{\text{BNN}}A)A^{-\frac{1}{2}} &= C C^T, \end{cases}$$

where the following equalities are used:

$$\begin{cases} (I - A^{\frac{1}{2}}QA^{\frac{1}{2}})^2 &= I - A^{\frac{1}{2}}QA^{\frac{1}{2}}, \\ (I - A^{\frac{1}{2}}QA^{\frac{1}{2}})^T &= I - A^{\frac{1}{2}}QA^{\frac{1}{2}}, \\ (I - A^{\frac{1}{2}}MA^{\frac{1}{2}})^T &= I - A^{\frac{1}{2}}M^T A^{\frac{1}{2}}, \\ I - A^{\frac{1}{2}}\widetilde{M}A^{\frac{1}{2}} &= (I - A^{\frac{1}{2}}MA^{\frac{1}{2}})(I - A^{\frac{1}{2}}M^T A^{\frac{1}{2}}). \end{cases}$$

Since  $A^{\frac{1}{2}}(I - \mathcal{P}_{\text{MG}}A)A^{-\frac{1}{2}}$  and  $A^{\frac{1}{2}}(I - \mathcal{P}_{\text{BNN}}A)A^{-\frac{1}{2}}$  are similar to  $I - \mathcal{P}_{\text{MG}}A$  and  $I - \mathcal{P}_{\text{BNN}}A$ , respectively, and  $\sigma(C^T C) = \sigma(C C^T)$  (see, e.g., [21]), we obtain

$$\sigma(I - \mathcal{P}_{\text{MG}}A) = \sigma(C^T C) = \sigma(I - \mathcal{P}_{\text{BNN}}A),$$

and the theorem follows immediately.  $\square$

From Theorem 6.1, we obtain that MG with  $M$  and BNN with  $\widetilde{M}$  give exactly the same condition number. This also implies that the condition number of MG is not smaller than the condition number of DEF.

**COROLLARY 6.2.** *Let  $M$  be given and  $\widetilde{M}$  be as in Theorem 6.1. Then,*

- (i)  $\kappa_{\text{MG}} = \kappa_{\text{BNN}}$ , and
- (ii)  $\kappa_{\text{DEF}} \leq \kappa_{\text{MG}}$ ,

where  $\kappa_{\text{MG}}$ ,  $\kappa_{\text{BNN}}$ , and  $\kappa_{\text{DEF}}$  are the condition numbers corresponding to MG, BNN, and DEF, respectively.

*Proof.* The corollary follows from Theorem 6.1 and [16, Thm. 2.7].  $\square$

*Remark 6.3.*

- Ordering the smoothers in the opposite way would lead to a different definition of  $\widetilde{M}$ ; this, in turn, could change the eigenvalues of MG and BNN, although an analogous result to Theorem 6.1 still holds for the consistent choice of  $S$  and  $\widetilde{M}$ .
- Corollary 6.2 shows that BNN, DEF, and MG are expected to show comparable convergence behavior for special choices of the fine-level preconditioners. We note that this result is valid only in exact arithmetic. If coarse-grid systems are solved inaccurately, for example, DEF might have convergence difficulties, while BNN and MG are less sensitive; see, e.g., [21].



**7. Numerical experiments.** In this section, we present the results of some numerical experiments to demonstrate the theoretical results presented above, as well as to compare with the performance of a one-level preconditioned CG algorithm, where the preconditioner  $M$  is used alone, denoted by PREC. In these results, a random starting vector is chosen and used for each iterative method, and the iterative process is terminated when the norm of the relative residual,  $\frac{\|r_{j+1}\|_2}{\|r_0\|_2}$ , falls below a fixed tolerance,  $\delta = 10^{-8}$ . We start with a 1D Laplace-like problem to illustrate the theory obtained in section 4. Then, a two-dimensional (2D) bubbly flow problem is used to show the performance of DEF, BNN, and MG with the choices discussed in the previous section. Finally, a 2D finite-element discretization of Poisson’s equation is used to demonstrate the difference between the typical choices of  $M$  and  $Z$  for BNN, MG, and DEF. We stress that these examples are chosen to highlight the presented theory and not to present the efficiency of the solvers; in practice, very different choices of  $M$  and  $Z$  are used for each method (see [12, 25]), and much larger problem sizes are needed for these approaches to be more efficient than optimized direct solvers.

**7.1. 1D Laplace-like problem.** Several 1D Laplace-like problems are considered, with the matrices

$$(7.1) \quad A = \begin{bmatrix} \beta & \gamma & & \emptyset \\ \gamma & \beta & \ddots & \\ & \ddots & \ddots & \gamma \\ \emptyset & & \gamma & \beta \end{bmatrix}, \quad \beta, \gamma \in \mathbb{R},$$

where we vary the constants  $\beta$  and  $\gamma$  so that each test case corresponds to a different region within Figure 4.1; see Table 7.1. In addition, we choose  $M = I$  and  $Z$  consisting of eigenvectors corresponding to the smallest eigenvalues of  $A$ . In all examples, we use  $\alpha = 1$  in MG, so that we can see cases where DEF is faster to converge than MG. On the right-hand side,  $b$ , is chosen randomly. We take  $n = 100$  (other values of  $n$  lead to approximately the same results), and the number of projection vectors,  $k$ , is varied. The results of the experiment can be found in Table 7.2.

TABLE 7.1  
 Test cases corresponding to different regions as presented in Figure 4.1.

| Test problem | $\beta$ | $\gamma$ | Range of $\lambda_i$ | Region  | Expected fastest method |
|--------------|---------|----------|----------------------|---------|-------------------------|
| (T1)         | 1.5     | -0.125   | [1.25, 1.75]         | B2      | DEF                     |
| (T2)         | 1       | -0.05    | [0.9, 1.1]           | A1 / A2 | MG                      |
| (T3)         | 0.25    | -0.1     | [0.05, 0.45]         | B1      | DEF                     |
| (T4)         | 1.25    | -0.125   | [1.0, 1.5]           | A2 / B2 | MG/DEF                  |

Table 7.2(a) shows that DEF yields a smaller condition number and is faster than MG for the choices of  $\beta$  and  $\gamma$  in (T1). On the other hand, as observed in Table 7.2(b),  $\beta$  and  $\gamma$  can also be chosen such that MG yields a smaller condition number and is faster to converge than DEF.

Since the condition number associated with DEF is always below that of MG in the case presented in Table 7.2(c), DEF is expected to converge in fewer iterations than MG; however, that is not the case. The two methods converge at roughly the same rate for large  $k$ , but MG is faster than DEF for small  $k$ . This can be explained by the fact that the spectrum of eigenvalues of MG consists of two clusters; see Figure 7.1(c). If the first cluster of 1’s in the spectrum of MG is omitted (or is approximated by a

TABLE 7.2

Results of the experiment with test cases as presented for the Laplace-like problems where the coefficients are given in Table 7.1. The results are presented in terms of number of iterations, # It., and condition number,  $\kappa$ .

(a)  $\beta = 1.5$ ,  $\gamma = -0.125$ .

|        | $k = 2$ |          | $k = 20$ |          | $k = 60$ |          |
|--------|---------|----------|----------|----------|----------|----------|
| Method | # It.   | $\kappa$ | # It.    | $\kappa$ | # It.    | $\kappa$ |
| PREC   | 11      | 1.4      | 11       | 1.4      | 11       | 1.4      |
| DEF    | 11      | 1.4      | 10       | 1.3      | 8        | 1.1      |
| BNN    | 11      | 1.7      | 10       | 1.7      | 8        | 1.7      |
| MG     | 15      | 2.3      | 15       | 2.3      | 12       | 2.3      |

(b)  $\beta = 1$ ,  $\gamma = -0.05$ .

|        | $k = 2$ |          | $k = 20$ |          | $k = 60$ |          |
|--------|---------|----------|----------|----------|----------|----------|
| Method | # It.   | $\kappa$ | # It.    | $\kappa$ | # It.    | $\kappa$ |
| PREC   | 9       | 1.2      | 9        | 1.2      | 9        | 1.2      |
| DEF    | 9       | 1.2      | 9        | 1.2      | 7        | 1.1      |
| BNN    | 9       | 1.2      | 9        | 1.2      | 7        | 1.1      |
| MG     | 5       | 1.01     | 5        | 1.01     | 5        | 1.01     |

(c)  $\beta = 0.25$ ,  $\gamma = -0.1$ .

|        | $k = 2$ |          | $k = 20$ |          | $k = 60$ |          |
|--------|---------|----------|----------|----------|----------|----------|
| Method | # It.   | $\kappa$ | # It.    | $\kappa$ | # It.    | $\kappa$ |
| PREC   | 34      | 9.0      | 34       | 9.0      | 34       | 9.0      |
| DEF    | 34      | 8.8      | 24       | 4.9      | 11       | 1.4      |
| BNN    | 34      | 19.6     | 25       | 11.0     | 11       | 3.2      |
| MG     | 30      | 10.1     | 22       | 5.7      | 11       | 1.9      |

(d)  $\beta = 1.25$ ,  $\gamma = -0.125$ .

|        | $k = 2$ |          | $k = 20$ |          | $k = 60$ |          |
|--------|---------|----------|----------|----------|----------|----------|
| Method | # It.   | $\kappa$ | # It.    | $\kappa$ | # It.    | $\kappa$ |
| PREC   | 11      | 1.5      | 11       | 1.5      | 11       | 1.5      |
| DEF    | 12      | 1.5      | 11       | 1.4      | 8        | 1.1      |
| BNN    | 12      | 1.5      | 11       | 1.5      | 8        | 1.5      |
| MG     | 10      | 1.3      | 10       | 1.3      | 9        | 1.3      |

Ritz value), then the condition number of the remaining spectrum is smaller than that of DEF. When  $k = 20$ , the ratio of the largest to smallest nonunit eigenvalues of MG is approximately 3.5, while the ratio of the largest to smallest nonzero eigenvalues of DEF is 4.9. While the CG convergence bound predicts 20 iterations for DEF (which requires 24), it predicts only 16 iterations for MG when ignoring the unit eigenvalues (while MG requires 22). These “extra” iterations for MG are likely the result of resolving the Ritz value at 1.

Finally, MG has a smaller condition number and is faster than DEF for small  $k$  in the case presented in Table 7.2(d). On the other hand, for large  $k$ , DEF has a smaller condition number than MG and performs somewhat better than MG. Indeed, the best method depends on  $k$  (through  $\lambda_{k+1}$ ) for this case with  $\beta = 1.25$  and  $\gamma = -0.125$ .

**7.2. 2D bubbly flow problem.** Using pressure-correction techniques for solving the Navier–Stokes equations, the major computational bottleneck that arises in

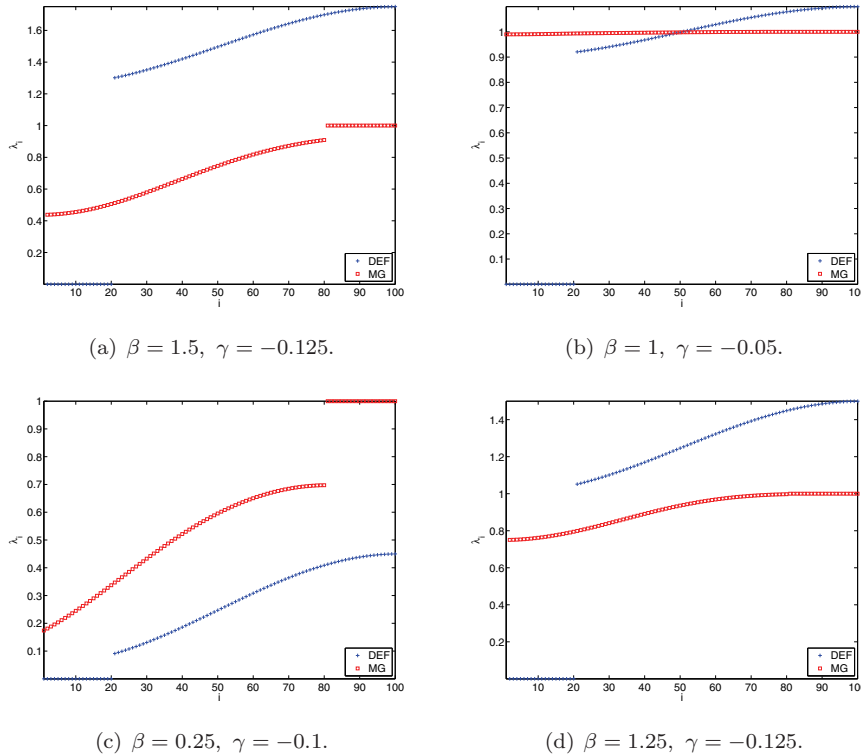


FIG. 7.1. Eigenvalues associated with DEF and MG for the test cases with  $k = 20$  as presented in Table 7.2.

modeling bubbly flows is the solution of the Poisson equation with a discontinuous coefficient,

$$(7.2) \quad -\nabla \cdot \left( \frac{1}{\rho(\mathbf{x})} \nabla p(\mathbf{x}) \right) = 0, \quad \mathbf{x} = (x, y) \in \Omega = (0, 1)^2,$$

where  $p$  denotes the pressure and  $\rho$  is a piecewise-constant density coefficient; see [12, 24] and the references therein. We consider circular bubbles of air in the domain,  $\Omega$ , that is otherwise filled with a fluid; see Figure 7.2(a) for the geometry. Here, a density contrast of  $10^3$  is considered. A standard second-order finite-difference scheme is applied to discretize (7.2), where we use a uniform Cartesian grid. Nonhomogeneous Neumann boundary conditions are imposed so that the resulting linear system (1.1) is still compatible. Moreover, we consider here the incomplete Cholesky decomposition without fill-in, IC(0), as the preconditioner,  $M$ . We refer the reader to [21] for more details about this experiment.

Let the open domain,  $\Omega$ , be divided into subdomains,  $\Omega_j, j = 1, 2, \dots, k + 1$ , such that  $\overline{\Omega} = \cup_{j=1}^{k+1} \overline{\Omega}_j$  and  $\Omega_i \cap \Omega_j = \emptyset$  for all  $i \neq j$ . The discretized domain and subdomains are denoted by  $\Omega_h$  and  $\Omega_{h_j}$ , respectively. Then, for each  $\Omega_{h_j}$  with  $j = 1, 2, \dots, k + 1$ , a projection vector,  $z_j$ , is defined as follows:

$$(7.3) \quad (z_j)_i := \begin{cases} 0, & x_i \in \Omega_h \setminus \overline{\Omega}_{h_j}, \\ 1, & x_i \in \Omega_{h_j}, \end{cases}$$

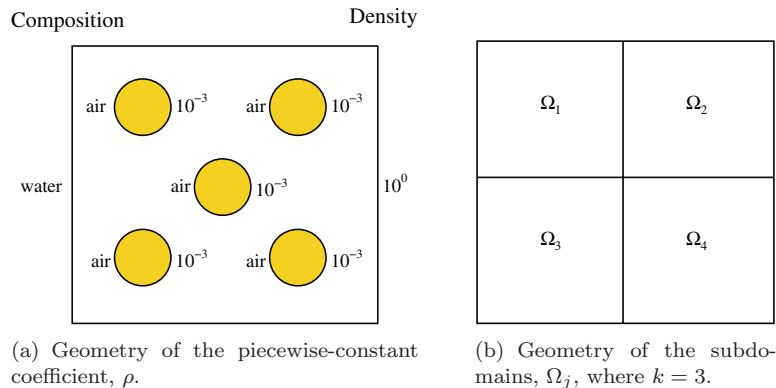


FIG. 7.2. Settings for the bubbly flow problem.

where  $x_i$  is a grid point of  $\Omega_h$ . The subdomains are identical square domains, which are chosen independently of the bubbles, and the number of them can be varied; see also Figure 7.2(b). It can be shown that these projection vectors accurately approximate the slowly varying eigenvectors corresponding to small eigenvalues when  $k$  is sufficiently large; see [12] and the references therein. Then, we take  $Z := [z_1 \ z_2 \ \cdots \ z_k]$ . Hence,  $Z$  consists of orthogonal, disjoint, and piecewise-constant vectors. We remark that the projection vectors are not restricted to choices that are common in domain decomposition and deflation. Typical MG projection vectors could also be taken, as will be seen in the next subsection.

Results for DEF, BNN, and MG using the same choice for  $M$  (IC(0)) are presented in Table 7.3. From the table, it can be observed that, for all  $k$ , DEF and BNN require the same number of iterations, whereas MG requires the fewest number of iterations, as expected from the discussion in section 2. Recall that this does not necessarily mean that MG is the fastest method with respect to computing time, since each iteration of MG is more expensive than an iteration of DEF. Moreover, note that the difference in performance between the cases with  $k = 3$  and  $k = 15$  is small. In these cases, the subspace spanned by the corresponding projection vectors allows only poor approximations to the eigenvectors associated with the smallest eigenvalues of  $A$ . With  $k = 63$ , the eigenvectors associated with the smallest eigenvalues of  $A$  are well approximated by the coarse-grid correction, and so we see fast convergence for all methods.

TABLE 7.3

Number of iterations required for convergence and the 2-norm of the relative errors of 2L-PCG methods for the bubbly flow problem with  $n = 64^2$  and using the same  $M$  in all three methods. PREC requires 137 iterations and leads to a relative error of  $4.6 \times 10^{-7}$ .

|        | $k = 2^2 - 1$ |                                    | $k = 4^2 - 1$ |                                    | $k = 8^2 - 1$ |                                    |
|--------|---------------|------------------------------------|---------------|------------------------------------|---------------|------------------------------------|
| Method | # It.         | $\frac{\ x_{it} - x\ _2}{\ x\ _2}$ | # It.         | $\frac{\ x_{it} - x\ _2}{\ x\ _2}$ | # It.         | $\frac{\ x_{it} - x\ _2}{\ x\ _2}$ |
| DEF    | 149           | $1.5 \times 10^{-8}$               | 144           | $3.1 \times 10^{-8}$               | 42            | $1.8 \times 10^{-8}$               |
| BNN    | 149           | $1.5 \times 10^{-8}$               | 144           | $3.1 \times 10^{-8}$               | 42            | $1.1 \times 10^{-8}$               |
| MG     | 86            | $1.0 \times 10^{-7}$               | 93            | $6.5 \times 10^{-8}$               | 32            | $1.9 \times 10^{-8}$               |

We now perform the same experiment as above, but using the symmetrized IC(0) preconditioner,  $M + M^T - MAM^T$ , in DEF and BNN. In contrast to the previous

TABLE 7.4

Number of iterations required for convergence and the 2-norm of the relative errors of 2L-PCG methods for the bubbly flow problem with  $n = 64^2$  and using the symmetrized  $\tilde{M}$  in DEF and BNN. PREC requires 137 iterations and leads to a relative error of  $4.6 \times 10^{-7}$ .

|        | $k = 2^2 - 1$ |                                  | $k = 4^2 - 1$ |                                  | $k = 8^2 - 1$ |                                  |
|--------|---------------|----------------------------------|---------------|----------------------------------|---------------|----------------------------------|
| Method | # It.         | $\frac{\ x_{it}-x\ _2}{\ x\ _2}$ | # It.         | $\frac{\ x_{it}-x\ _2}{\ x\ _2}$ | # It.         | $\frac{\ x_{it}-x\ _2}{\ x\ _2}$ |
| DEF    | 87            | $7.2 \times 10^{-8}$             | 94            | $1.3 \times 10^{-8}$             | 34            | $7.6 \times 10^{-9}$             |
| BNN    | 87            | $7.2 \times 10^{-8}$             | 94            | $1.3 \times 10^{-8}$             | 34            | $7.6 \times 10^{-9}$             |
| MG     | 86            | $1.0 \times 10^{-7}$             | 93            | $6.5 \times 10^{-8}$             | 32            | $1.9 \times 10^{-8}$             |

experiment, the amount of work for each iteration of BNN, MG, and DEF is now approximately the same and Theorem 6.1 holds. The results of this experiment are presented in Table 7.4. As can be observed in the table, the performance of all three methods is now nearly identical, as expected from the theory of section 6.

**7.3. Comparing choices of  $M$  and  $Z$ .** In this section, we compare the preconditioners that result from making choices for  $M$  and  $Z$  that are typical for each of the families of preconditioners considered here. For simplicity, we will base our comparison on a 2D finite-element discretization of the constant-coefficient Laplace operator,  $-\Delta u = f$ , on the unit square with Dirichlet boundary conditions.

For DEF, we consider a similar combination of results as those presented in the previous subsection, with  $M$  chosen as the inverse of the IC(0) approximation to  $A$ , and  $Z$  chosen based on subdomain vectors. Note that since we consider the problem with Dirichlet boundary conditions, we include all subdomain vectors,  $z_j$ , in  $Z$ , in contrast to the Neumann boundary condition case considered in section 7.2.

For BNN, we consider a similar partitioning of the computational domain into subdomains (now with overlap along subdomain boundaries), with  $Z$  chosen based on the overlapping subdomain vectors, weighted so that the columns of  $Z$  remain a partition of unity. To choose  $M$ , however, we use the matrix given by assembling local exact inverses of the assembled element-stiffness matrices over each subdomain, weighted to account for the subdomain boundary overlap. In these tests, we ignore the elimination of the interior degrees of freedom on each subdomain that is normally used in BNN [13]; while important for practical reasons, our goal here is to present a simpler comparison within the framework considered above. For the interior subdomains, the locally assembled element-stiffness matrices are singular; we address this singularity when computing the inverse by first computing the exact  $LU$  factorization of the (singular) matrix and then adding a small constant to the lower-right diagonal entry of  $U$  before computing  $U^{-1}L^{-1}$ . This gives a low-cost approximation of the pseudo-inverse of the singular subdomain matrix but has the effect that  $MA$  has  $(k - 2)^2$  large eigenvalues, one for each singular subdomain matrix whose  $LU$  factorization has been perturbed. Each of the eigenvectors corresponding to these eigenvalues, however, is in the range of  $Z$ , and, thus, the errors associated with these modes are exactly eliminated by the coarse-grid correction process.

For MG, we consider a weighted Jacobi smoother, with  $M = \frac{4}{5}D^{-1}$ , where  $D$  is the diagonal part of the discretization matrix. The subdomain vectors in  $Z$  are chosen based on linear interpolation from a coarse grid that is coarsened by a factor of 2, 4, or 8 in each direction.

Table 7.5 gives iteration counts for 2L-PCG using DEF, BNN, and MG as preconditioners with  $M$  and  $Z$  chosen as for deflation. In these tests, we always take the

TABLE 7.5

2L-PCG iteration counts for preconditioners with  $M$  and  $Z$  chosen as in subdomain-based deflation preconditioners.

|                  | 2 <sup>2</sup> pts per subdmn |     |    | 4 <sup>2</sup> pts per subdmn |     |    | 8 <sup>2</sup> pts per subdmn |     |    |
|------------------|-------------------------------|-----|----|-------------------------------|-----|----|-------------------------------|-----|----|
| $n$              | DEF                           | BNN | MG | DEF                           | BNN | MG | DEF                           | BNN | MG |
| 32 <sup>2</sup>  | 12                            | 11  | 8  | 15                            | 14  | 10 | 18                            | 17  | 12 |
| 64 <sup>2</sup>  | 12                            | 11  | 9  | 17                            | 16  | 13 | 22                            | 21  | 17 |
| 128 <sup>2</sup> | 13                            | 12  | 9  | 17                            | 16  | 14 | 24                            | 23  | 19 |

grid to be evenly divisible by the number of subdomains, so that each subdomain is of the same size. For each problem size, we consider the performance of these algorithms when the subdomains have 2<sup>2</sup>, 4<sup>2</sup>, and 8<sup>2</sup> points each. For this choice of  $M$  and  $Z$ , we see that the iteration counts for all methods are similar when the subdomains are of the same size. As expected, as the size of the subdomains grows (and, thus, the total number of subdomains shrinks), the iteration counts grow somewhat.

Similarly, Table 7.6 gives iteration counts for 2L-PCG using DEF, BNN, and MG as preconditioners with  $M$  and  $Z$  chosen as for BNN. Here, we assume that the subdomains are overlapping, and so the fine-grid domain size is of one point fewer in each direction than would evenly divide the domain into the required number of subdomains. For each problem size, we consider the performance of these algorithms when the subdomains have 2<sup>2</sup>, 4<sup>2</sup>, and 8<sup>2</sup> points each. For this choice of  $M$  and  $Z$ , we see that the performance of the DEF and BNN preconditioners is nearly identical (as expected) and grows roughly with the logarithm of the subdomain size. The preconditioner based on MG with this choice of  $M$  and  $Z$  performs quite poorly, however. This is, perhaps, not surprising, since the MG preconditioner applies the smoother based on  $M$  twice per iteration and measures reduction in the residual between these steps and not after the coarse-grid correction step that damps the errors magnified by  $M$ . While Theorem 4.1 does not directly apply here (as the columns of  $Z$  are not eigenvectors of  $MA$ ), we see similar results; the spectrum of  $MA$  is not tightly clustered around 1, and, thus, the convergence of MG is poorer than that of DEF and BNN.

TABLE 7.6

2L-PCG iteration counts for preconditioners with  $M$  and  $Z$  chosen as in balancing Neumann-Neumann preconditioners.

|                  | 2 <sup>2</sup> pts per subdmn |     |       | 4 <sup>2</sup> pts per subdmn |     |       | 8 <sup>2</sup> pts per subdmn |     |       |
|------------------|-------------------------------|-----|-------|-------------------------------|-----|-------|-------------------------------|-----|-------|
| $n$              | DEF                           | BNN | MG    | DEF                           | BNN | MG    | DEF                           | BNN | MG    |
| 31 <sup>2</sup>  | 6                             | 5   | 295   | 12                            | 11  | 158   | 18                            | 17  | 170   |
| 63 <sup>2</sup>  | 6                             | 5   | > 500 | 12                            | 11  | 404   | 18                            | 17  | > 500 |
| 127 <sup>2</sup> | 6                             | 5   | > 500 | 12                            | 11  | > 500 | 18                            | 17  | > 500 |

While geometric multigrid is almost always used with coarsening by a factor of only 2 in each direction, this is not a requirement of the algorithm. However, because of the overlapping nature of bilinear interpolation on a quadrilateral grid, the coarsening process in multigrid cannot easily be thought of in terms of subdomains. In Table 7.7, we give iteration counts for conjugate gradient using DEF, BNN, and MG as preconditioners using  $M$  and  $Z$  chosen as for MG, with coarsening by factors

TABLE 7.7

*2L-PCG iteration counts for preconditioners with  $M$  and  $Z$  chosen as in geometric multigrid.*

| $n$     | Coarsening by $2^2$ |     |    | Coarsening by $4^2$ |     |    | Coarsening by $8^2$ |     |    |
|---------|---------------------|-----|----|---------------------|-----|----|---------------------|-----|----|
|         | DEF                 | BNN | MG | DEF                 | BNN | MG | DEF                 | BNN | MG |
| $31^2$  | 12                  | 11  | 6  | 23                  | 22  | 14 | 41                  | 40  | 26 |
| $63^2$  | 12                  | 11  | 6  | 23                  | 22  | 14 | 43                  | 42  | 27 |
| $127^2$ | 12                  | 11  | 6  | 23                  | 22  | 14 | 43                  | 42  | 27 |

of  $2^2$ ,  $4^2$ , and  $8^2$ , resulting in a coarse-grid matrix  $E = Z^T AZ$  that is of the same size as those for DEF or BNN with  $2^2$ ,  $4^2$ , and  $8^2$  points per subdomain, respectively. Here, we again see the nearly equivalent performance of DEF and BNN, with slightly smaller iteration counts for MG.

The results in this section emphasize that, to a large degree, the particular choice of a DEF, BNN, or MG-style preconditioner is much less important than the choice of  $M$  and  $Z$  (with the exception of the poorly conditioned choice of  $M$  that arises from the local Neumann conditions used within BNN). We emphasize, however, that this comparison is made only in terms of the number of 2L-PCG iterations for each choice of  $M$  and  $Z$ . As discussed earlier, for the algorithms compared here, the cost per iteration of the MG-style preconditioner is larger than that for the DEF or (reduced) BNN preconditioners. From Theorem 6.1 and its corollary, we know that by choosing  $M$  so that the cost per iteration of these approaches is the same, their performance will also be equivalent, as shown in the example given in Table 7.4. The results in this section show that when  $M$  and  $Z$  are held constant, we still often see consistent results for all methods.

**8. Conclusions.** We compare two-level PCG methods based on deflation (DEF), balancing Neumann–Neumann (BNN), and multigrid V(1,1)-cycle (MG) preconditioners in their abstract forms, which all consist of combinations of one-level (traditional) and projection-type preconditioners. When specific choices are made for the algorithmic components, each MG iteration is more expensive than a DEF or BNN iteration, due to the more sophisticated form of the two-level preconditioner. At first glance, we would expect MG to be the most effective method; however, we have shown that there exist some choices for the algorithmic components for which DEF is expected to converge faster than MG in exact arithmetic.

When Richardson relaxation is used with an optimal weighting as the one-level preconditioner, the MG preconditioner is shown to always yield a smaller condition number than those for DEF or BNN. For more sophisticated and effective one-level preconditioners, we still expect MG to be superior to DEF and BNN, although the work per iteration of MG remains more than that for the other methods.

If, on the other hand, the one-level preconditioners are chosen carefully (as prescribed above), there are variants of the BNN, DEF, and MG preconditioners that all require the same amount of work per iteration. In this case, their spectra differ only in one cluster of eigenvalues around 0 or 1. Hence, these methods are expected to show a comparable convergence behavior, assuming that coarse-grid systems are solved accurately. This is observed in the given numerical results. In other words, the performance of the BNN, DEF, and MG preconditioners depends strongly on the choice of the parameters rather than on the choice of a specific two-level preconditioner.

## REFERENCES

- [1] J. H. BRAMBLE, J. E. PASCIAK, J. WANG, AND J. XU, *Convergence estimates for multigrid algorithms without regularity assumptions*, Math. Comp., 57 (1991), pp. 23–45.
- [2] M. BREZINA, A. J. CLEARY, R. D. FALGOUT, V. E. HENSON, J. E. JONES, T. A. MANTEUFFEL, S. F. MCCORMICK, AND J. W. RUGE, *Algebraic multigrid based on element interpolation (AMGe)*, SIAM J. Sci. Comput., 22 (2000), pp. 1570–1592.
- [3] W. L. BRIGGS, V. E. HENSON, AND S. F. MCCORMICK, *A Multigrid Tutorial*, 2nd ed., SIAM, Philadelphia, 2000.
- [4] B. CARPENTIERI, L. GIRAUD, AND S. GRATTON, *Additive and multiplicative two-level spectral preconditioning for general linear systems*, SIAM J. Sci. Comput., 29 (2007), pp. 1593–1612.
- [5] V. FABER AND T. MANTEUFFEL, *Necessary and sufficient conditions for the existence of a conjugate gradient method*, SIAM J. Numer. Anal., 21 (1984), pp. 352–362.
- [6] R. D. FALGOUT, P. S. VASSILEVSKI, AND L. T. ZIKATANOV, *On two-grid convergence estimates*, Numer. Linear Algebra Appl., 12 (2005), pp. 471–494.
- [7] A. FROMMER, R. NABBEN, AND D. B. SZYLD, *Convergence of stationary iterative methods for Hermitian semidefinite linear systems and applications to Schwarz methods*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 925–938.
- [8] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, MD, 1996.
- [9] W. HACKBUSCH, *Convergence of multi-grid iterations applied to difference equations*, Math. Comp., 34 (1980), pp. 425–440.
- [10] W. HACKBUSCH, *Multigrid Methods and Applications*, Springer-Verlag, Berlin, 1985.
- [11] S. MACLACHLAN, T. MANTEUFFEL, AND S. MCCORMICK, *Adaptive reduction-based AMG*, Numer. Linear Algebra Appl., 13 (2006), pp. 599–620.
- [12] S. P. MACLACHLAN, J. M. TANG, AND C. VUIK, *Fast and robust solvers for pressure-correction in bubbly flow problems*, J. Comput. Phys., 227 (2008), pp. 9742–9761.
- [13] J. MANDEL, *Balancing domain decomposition*, Comm. Numer. Methods Engrg., 9 (1993), pp. 233–241.
- [14] S. F. MCCORMICK AND J. W. RUGE, *Multigrid methods for variational problems*, SIAM J. Numer. Anal., 19 (1982), pp. 924–929.
- [15] R. NABBEN AND C. VUIK, *A comparison of deflation and coarse grid correction applied to porous media flow*, SIAM J. Numer. Anal., 42 (2004), pp. 1631–1647.
- [16] R. NABBEN AND C. VUIK, *A comparison of deflation and the balancing preconditioner*, SIAM J. Sci. Comput., 27 (2006), pp. 1742–1759.
- [17] R. A. NICOLAIDES, *On some theoretical and practical aspects of multigrid methods*, Math. Comp., 33 (1979), pp. 933–952.
- [18] R. A. NICOLAIDES, *Deflation of conjugate gradients with applications to boundary value problems*, SIAM J. Numer. Anal., 24 (1987), pp. 355–365.
- [19] G. STRANG, *Introduction to Linear Algebra*, 4th ed., SIAM, Philadelphia, 2009.
- [20] J. M. TANG, S. P. MACLACHLAN, R. NABBEN, AND C. VUIK, *Theoretical comparison of two-level preconditioners based on multigrid and deflation*, DIAM Report 08-05, Delft University of Technology, Delft, The Netherlands, 2008.
- [21] J. M. TANG, R. NABBEN, C. VUIK, AND Y. A. ERLANGGA, *Comparison of two-level preconditioners derived from deflation, domain decomposition and multigrid methods*, J. Sci. Comput., 39 (2009), pp. 340–370.
- [22] A. TOSELLI AND O. WIDLUND, *Domain Decomposition Methods—Algorithms and Theory*, Springer Ser. Comput. Math. 34, Springer, Berlin, 2005.
- [23] U. TROTTERBERG, C. W. OOSTERLEE, AND A. SCHÜLLER, *Multigrid*, Academic Press, London, 2000.
- [24] S. P. VAN DER PIJL, *Computation of Bubbly Flows with a Mass-Conserving Level-Set Method*, Ph.D. thesis, Delft University of Technology, Delft, The Netherlands, 2005.
- [25] S. P. VAN DER PIJL, A. SEGAL, C. VUIK, AND P. WESSELING, *A mass-conserving level-set method for modelling of multi-phase flows*, Internat. J. Numer. Methods Fluids, 47 (2005), pp. 339–361.
- [26] P. VASSILEVSKI, *Multilevel Block Factorization Preconditioners: Matrix-Based Analysis and Algorithms for Solving Finite Element Equations*, Springer, New York, 2008.
- [27] C. VUIK, A. SEGAL, AND J. A. MELJERINK, *An efficient preconditioned CG method for the solution of a class of layered problems with extreme contrasts in the coefficients*, J. Comput. Phys., 152 (1999), pp. 385–403.



- [28] P. WESSELING, *An Introduction to Multigrid Methods*, John Wiley & Sons, Chichester, UK, 1992; corrected reprint, R.T. Edwards, Philadelphia, 2004.
- [29] J. XU, *Iterative methods by space decomposition and subspace correction*, SIAM Rev., 34 (1992), pp. 581–613.
- [30] D. M. YOUNG, *Iterative Solutions of Large Linear Systems*, Academic Press, New York, 1971.