SIAM J. SCI. COMPUT.
Vol. 19, No. 1, pp. 87–110, January 1998

007

AN EVALUATION OF PARALLEL MULTIGRID AS A SOLVER AND A PRECONDITIONER FOR SINGULARLY PERTURBED PROBLEMS*

C. W. OOSTERLEE† AND T. WASHIO‡

Abstract. In this paper we try to achieve h-independent convergence with preconditioned GMRES ([Y. Saad and M. H. Schultz, *SIAM J. Sci. Comput.*, 7 (1986), pp. 856–869]) and BiCGSTAB ([H. A. Van der Vorst, *SIAM J. Sci. Comput.*, 13 (1992), pp. 63–644]) for two-dimensional (2D) singularly perturbed equations. Three recently developed multigrid methods are adopted as a preconditioner. They are also used as solution methods in order to compare the performance of the methods as solvers and as preconditioners.

Two of the multigrid methods differ only in the transfer operators. One uses standard matrix-dependent prolongation operators from [J. E. Dendy Jr., *J. Comput. Phys.*, 48 (1982), pp. 366–386], [W. Hackbusch, *Multi-grid Methods and Applications*, Springer, Berlin, 1985]. The second uses "upwind" prolongation operators, developed in [P. M. de Zeeuw, *J. Comput. Appl. Math.*, 33 (1990), pp. 1–27]. Both employ the Galerkin coarse grid approximation and an alternating zebra line Gauss–Seidel smoother. The third method is based on the block LU decomposition of a matrix and on an approximate Schur complement. This multigrid variant is presented in [A. Reusken, *A Multigrid Method Based on Incomplete Gaussian Elimination*, University of Eindhoven, the Netherlands, 1995]. All three multigrid algorithms are algebraic methods.

The eigenvalue spectra of the three multigrid iteration matrices are analyzed for the equations solved in order to understand the convergence of the three algorithms. Furthermore, the construction of the search directions for the multigrid preconditioned GMRES solvers is investigated by the calculation and solution of the minimal residual polynomials.

For Poisson and convection-diffusion problems all solution methods are investigated and evaluated for finite volume discretizations on fine grids. The methods have been parallelized with a grid partitioning technique and are compared on an MIMD machine.

Key words. Krylov subspace methods, multigrid, robustness, parallel computing, grid partitioning

AMS subject classifications. 65N55, 65F10, 65Y05

PII. S1064827596302825

1. Introduction.

In the search for robust and efficient Krylov subspace methods, multigrid is being considered as a preconditioner. With preconditioners based on multigrid it is expected that grid–size-independent convergence can be achieved for a large class of problems. BiCGSTAB ([19]) and GMRES ([14]) will be used as Krylov subspace solvers. Several 2D singularly perturbed problems are considered, for which the design of optimal standard multigrid is not easy. For these problems the adopted multigrid methods are being compared as solvers and as preconditioners. The purpose of this work is not to make optimal multigrid methods for specific problems but merely to construct a robust well-parallelizable solver, in which the smoother as well as the coarse grid correction is fixed.

Three different state-of-the-art multigrid methods are being compared. The first two algorithms differ only in the transfer operators. Matrix-dependent transfer operators are employed so that problems with convection dominance, as well as problems

*Received by the editors April 29, 1996; accepted for publication (in revised form) August 19, 1996.

http://www.siam.org/journals/sisc/19-1/30282.html

†GMD, Institute for Algorithms and Scientific Computing, D-53754 Sankt Augustin, Germany (oosterlee@gmd.de).

‡C&C Research Laboratories, NEC Europe Ltd., D-53757 Sankt Augustin, Germany (washio@ccrl-nece.technopark.gmd.de).

with jumping coefficients, can be solved efficiently. The first algorithm includes well-known matrix-dependent operators from the literature ([3], [6]). In [3] the operators have been designed so that problems on grids with arbitrary mesh sizes, not just powers of two (+1), can be solved with similar efficiency. Although in [3] these operators are mainly used for symmetric interface problems, we will also consider them here for unsymmetric problems.

The second algorithm uses the prolongation operators introduced by de Zeeuw ([25]), which are specially designed for solving unsymmetric problems. Both algorithms employ Galerkin coarsening ([6], [24]) for building the matrices on coarser grids. A robust smoother is a necessary requirement in standard multigrid for achieving grid–size-independent convergence for many problems. The alternating zebra line Gauss–Seidel relaxation method was found to be a robust smoother for 2D model equations ([15], [24]). This smoother is used since it is possible to parallelize a line solver efficiently on a parallel machine ([23]). Constructing robust parallel solvers is an important purpose of our work.

The third multigrid algorithm in our comparison is based on the block Gaussian elimination of a matrix and an approximate Schur complement. This method has been recently introduced by Reusken ([12]). An interesting aspect is that the M-matrix ([20]) properties of a fine grid matrix are preserved on coarse grid levels. The three multigrid methods are described in section 2.

Another robust multigrid variant for solving scalar partial differential equations is algebraic multigrid (AMG) by Ruge and Stüben ([13]), in which the smoother is fixed but the transfer operators depend on the connections in a matrix. Efficient parallelization of AMG is, however, not trivial. Other multigrid preconditioners have been used for problems on structured grids, for example, in [1], [16], [17], and for unstructured grids in [2]. In [9] an early comparison of multigrid and multigrid preconditioned CG for symmetric model equations showed the promising robustness of the latter method. A comparison between multigrid as a solver and as a preconditioner for incompressible Navier–Stokes equations is presented in [26].

The solution methods are analyzed in order to understand the convergence behavior of the *MG* methods used as solvers and as preconditioners. The eigenvalue spectrum of the multigrid iteration matrices for the singularly perturbed problems is calculated in section 3. Interesting subjects for the convergence behavior are the spectral radius and the eigenvalue distribution. Another paper in which eigenvalues of multigrid preconditioned Krylov methods are determined for convergence analysis is [16]. Furthermore, the polynomials that give a representation of the GMRES residual in terms of the preconditioned matrix and the initial residual are explicitly computed and solved. The evolution of the subspace in which GMRES minimizes the residual can be determined from the solution of these polynomials.

Numerical results are also presented in section 3. The benefits of the constructed multigrid preconditioned Krylov methods are shown for 2D Poisson and convection-diffusion problems on fine grids solved on the NEC Cenju–3 MIMD machine ([8]). The message passing is done with the message-passing interface (MPI). Furthermore, the methods are compared with MILU ([18]) preconditioned Krylov methods.

In [23] we compared several standard and nonstandard multilevel preconditioners for BiCGSTAB. The changes in the multigrid algorithm presented in this paper can also be applied to the nonstandard multilevel approach MG-S from [23]. Further research on parallel nonstandard multilevel preconditioners particularly for 3D problems (in order to avoid alternating plane smoothers) will be described in part II of this paper.

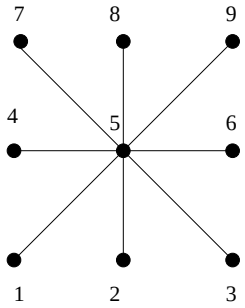

FIG. 1. *The nine-point stencil with numbering.*

2. The multigrid preconditioned Krylov methods. We concentrate on linear matrix systems with nine diagonals

$$A\phi = b \tag{1}$$

or

$$\begin{aligned} &a^1_{i,j}\phi_{i-1,j-1} + a^2_{i,j}\phi_{i,j-1} + a^3_{i,j}\phi_{i+1,j-1} + a^4_{i,j}\phi_{i-1,j} + a^5_{i,j}\phi_{i,j} + a^6_{i,j}\phi_{i+1,j} \\ &\quad + a^7_{i,j}\phi_{i-1,j+1} + a^8_{i,j}\phi_{i,j+1} + a^9_{i,j}\phi_{i+1,j+1} = b_{i,j}\ \forall (i,j) \in G. \end{aligned} \tag{2}$$

Here, G is a subset of $P^{n_x,n_y} = \{(i,j)|1 \le i \le n_x,\ 1 \le j \le n_y\}$.

The stencil is presented in Figure 1 for convenience.

Matrix A has right preconditioning as follows:

$$AK^{-1}(K\phi) = b. \tag{3}$$

The Krylov subspace methods that are used for solving (3) are BiCGSTAB ([19]), as described in detail in [23], and GMRES(m) ([14]).

2.1. GMRES. The GMRES(m) algorithm with a right preconditioner appears as follows:

GMRES (m,A, b, ϕ) {
Choose $\phi^{(0)}$, dimension m, matrix $\tilde{\mathbf{H}} = \mathbf{0}$ with dim: $(m+1) \times m$
$r^{(0)} = b - A\phi^{(0)}$; $\beta = ||r^{(0)}||_2$; $f_1 = r^{(0)}/\beta$;
for $j = 1, \ldots, m$ {
$u_j := K^{-1} f_j$;
$w := Au_j$;
for $i = 1, \ldots, j$ {
$h_{i,j} := (w, f_i)$;
$w := w - h_{i,j} f_j$;
$h_{j+1,j} := ||w||_2$, $f_{j+1} = w/h_{j+1,j}$;
}
}
Define $\mathbf{F}_m := [f_1, \ldots, f_m]$;
$\phi^{(m)} := \phi^{(0)} + K^{-1}\mathbf{F}_m y_m$; with $y_m = \min_y ||\beta e_1 - \tilde{\mathbf{H}} y||_2$,
$(e_1 = [1, 0, \ldots, 0]^T)$;
Compute $r^{(m)} = b - A\phi^{(m)}$;
If satisfied **stop**, else restart $\phi^{(0)} \leftarrow \phi^{(m)}$;
}

$K^{-1}f_j$ is the preconditioning step, which is one iteration of a multigrid cycle.

The algorithms are described in a C-type metalanguage. The software is implemented as a combination of Fortran 77 and C. C is used for the driver routines of the preconditioner in order to exploit the recursiveness of C and so to simplify the multigrid implementation.

Storage. The parameter m represents the number of vectors stored after which the GMRES(m) algorithm will restart. It is set to 20 in our work. Restart vectors are not necessary in BiCGSTAB, where six intermediate vectors have to be stored. The work for one GMRES iteration consists of one matrix–vector product and one preconditioning step. In BiCGSTAB two matrix–vector products and two preconditioning steps are necessary in each iteration. The computation of one iteration of BiCGSTAB is twice as expensive as an iteration of GMRES(m), when, as in our case, the computation is dominated by the preconditioning and the matrix–vector product. When the number of iterations needed for convergence is much larger than the restart parameter m, it might be better to choose BiCGSTAB.

Parallelism is straightforward in Krylov methods, except for the multigrid preconditioner, the matrix–vector, and inner products, which need communication among neighboring processors.

Using a preconditioner as solver. A preconditioner, like the multigrid preconditioner, can easily be used as a solver. An iteration of a multigrid solver is equivalent to a Richardson iteration on the preconditioned matrix. With K being the iteration matrix, multigrid can be written as follows:

$$K\phi^{(k+1)} + (A - K)\phi^{(k)} = b. \tag{4}$$

This formulation is equivalent to

$$\phi^{(k+1)} = \phi^{(k)} + K^{-1}(b - A\phi^{(k)}) = \phi^{(k)} + K^{-1}r^{(k)};\ \ r^{(k+1)} = (I - AK^{-1})r^{(k)}. \tag{5}$$

The multigrid solver is implemented as a Richardson iteration with a left multigrid preconditioner for A. The convergence of (5) can be investigated by analyzing the spectrum of the iteration matrix. The spectral radius of $I - AK^{-1}$ determines the convergence. This spectrum is analyzed in section 3 for the different multigrid methods for the problems tested on 33^2 and 65^2 grids. With this spectrum we can also investigate the convergence of GMRES, since the spectra of left and right preconditioned matrices are the same.

Nested iteration ([6]), or full multigrid (FMG), which often reduces the number of multigrid iterations, is closely related to the multigrid algorithms considered here. Our starting point is the sparse matrix on the finest grid, but an F-cycle without presmoothing is always used.

The convergence of GMRES. Here, we give an estimation of the convergence of GMRES(m) in cases where most of the eigenvalues (the last $n - l$ eigenvalues in the theorem below) of the preconditioned matrix $\tilde{A}$ are close to $1 \in \mathbf{C}$.

THEOREM 2.1. *Let $\tilde{A}$ be an $n \times n$ nonsingular matrix with eigenvalues $\{\lambda_i \in \mathbf{C} \mid 1 \le i \le n\}$, and let $\bar{V}_l$ be the subspace spanned by the vectors $\{v \in V | \prod_{k>l}^{n}(\lambda_k I - \tilde{A})v = 0\}$, let K_l be the Krylov subspace $K(l, r^{(0)}, \tilde{A}) = span[r^{(0)}, \tilde{A}r^{(0)}, \ldots, \tilde{A}^{l-1}r^{(0)}]$, and let P_k be a set of kth-order complex polynomials $p_k(\lambda)$ that satisfy $p_k(0) = 1$, and $r^{(0)} = b - \tilde{A}\phi^{(0)}$ the initial residual. Let radius Γ_i be defined as*

$$\Gamma_i := \max\{||(\lambda_i I - \tilde{A})v||_2 \ \ | v \in \bar{V}_{i-1} \cap K_i;\ \|v\|_2 = 1\}. \tag{6}$$

Then a vector $\bar{r}^{(l)}$ *defined by*

$$\bar{r}^{(l)} := \left(\prod_{i=1}^{l} \frac{1}{\lambda_i} (\lambda_i I - \tilde{A}) \right) r^{(0)} \tag{7}$$

is included in $\bar{V}_l \cap K_{l+1}$.

Furthermore,

$$\|\bar{r}^{(l)}\|_2 \leq \left(\prod_{i=1}^{l} \frac{\Gamma_i}{|\lambda_i|} \right) \|r^{(0)}\|_2. \tag{8}$$

Assuming $l < k \leq m$, *then the norm of the residual of* k*th GMRES(*m*) iteration can be estimated as follows:*

$$\|r^{(k)}\|_2 \leq \min\{\|p_{k-l}(\tilde{A})\bar{r}^{(l)}\|_2 \mid p_{k-l}(\lambda) \in P_{k-l}\} \tag{9}$$

$$\leq \|(I - \tilde{A})^{k-l}\bar{r}^{(l)}\|_2. \tag{10}$$

Proof. Inequality (8) is found by induction: $\bar{r}^{(l)} = \frac{1}{\lambda_l}(\lambda_l \bar{r}^{(l-1)} - \tilde{A}\bar{r}^{(l-1)}) \in \bar{V}_l \cap K_{l+1}$, so

$$\|\bar{r}^{(l)}\|_2 = \left\| \frac{1}{\lambda_l}(\lambda_l I - \tilde{A})\bar{r}^{(l-1)} \right\|_2 \leq \frac{\Gamma_l}{|\lambda_l|}\|\bar{r}^{(l-1)}\|_2. \tag{11}$$

GMRES(m) is one of the algorithms which searches an update q in the Krylov subspace $K(k, r^{(0)}, \tilde{A})$ since it minimizes residual $r^{(k)} = b - \tilde{A}(\phi^{(0)} + q) = r^{(0)} - \tilde{A}q$ at the kth iteration ([14]). Hence, from this fact,

$$\|r^{(k)}\|_2 = \min\{\|p_k(\tilde{A})\bar{r}^{(0)}\|_2 \mid p_k(\lambda) \in P_k\} \tag{12}$$

is satisfied for $k \leq m$. From (12) and the fact that

$$s_k(\lambda) = \left(\prod_{i=1}^{l} \frac{\lambda_i - \lambda}{\lambda_i} \right) p_{k-l}(\lambda) \tag{13}$$

is included in P_k for an arbitrary $p_{k-l}(\lambda) \in P_{k-l}$, inequality (9) is obvious.

Inequality (10) is found with $(1 - \lambda)^{k-l}$ as a choice for $p_{k-l}(\lambda)$ in (9). ∎

Although $\tilde{A}$ is not symmetric in our case, it might be helpful for understanding (8) to consider the symmetric case. As upperbound for Γ_i it then follows that

$$\begin{aligned} \Gamma_i \leq \bar{\Gamma}_i &=: \max\{||(\lambda_i I - \tilde{A})v||_2 \ \ |v \in \bar{V}_{i-1};\ \|v\|_2 = 1\} \\ &= \max\{|\lambda_i - \lambda_j| \ \mid j \geq i\}. \end{aligned} \tag{14}$$

Figure 2 shows $\bar{\Gamma}_i$ in this case.

When $\tilde{A}$ is nonsymmetric, we see that

$$\bar{\Gamma}_i \geq \max\{|\lambda_i - \lambda_j| \ \mid j \geq i\}. \tag{15}$$

Here are some remarks concerning the theorem.

Remark 2.1.1. If some eigenvalues λ_i ($i \leq l$) are close to zero, the norm of $\bar{r}^{(l)}$ becomes large. This suggests that we may need a relatively large k to reduce the residual by a certain order of magnitude even if l is not large.

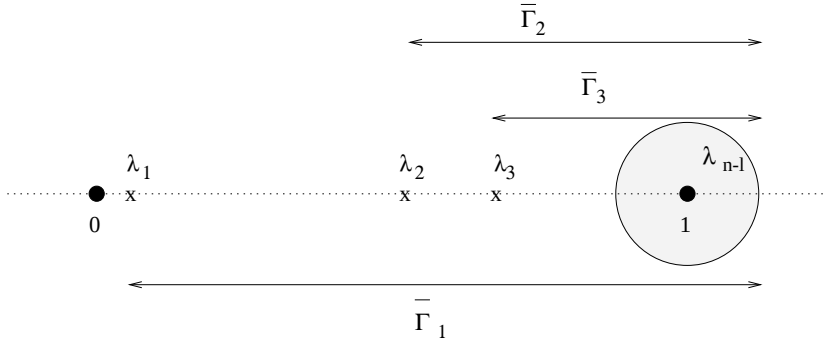

FIG. 2. $\bar{\Gamma_i}$ *for a case where most eigenvalues are near one.*

Remark 2.1.2. Inequality (9) shows that $\|r^{(k)}\|_2$ is not larger than the norm of the $(k-l)$th residual of GMRES(m) with initial residual $\bar{r}^{(l)}$ that is included in the subspace corresponding to the eigenvalues close to one $\{\lambda_i \mid i > l\}$.

Remark 2.1.3. As already mentioned, $(I - AK^{-1})$ gives the residual transformation for the Richardson iterative process based on the matrix splitting $A = K + (A - K)$. Hence, from (10), the asymptotic convergence of GMRES(m) is at least faster than Richardson with an initial residual; that does not include components corresponding to the large eigenvalues $\{\lambda_i \mid i \leq l\}$ of AK^{-1}.

GMRES polynomial. We construct the GMRES polynomial p_k to understand the way the GMRES search directions are constructed. Here, the GMRES polynomial means $p_k \in P_k$, which gives the residual vector $r^{(k)}$ of the kth GMRES iterative process as $r^{(k)} = p_k(\tilde{A})r^{(0)}$. As mentioned in the proof of Theorem 2.1, p_k gives the minimal L_2-norm of the residual under the condition $p_k(0) = 1$. The minimal residual polynomial p_k looks like

$$p_k(\tilde{A})r^{(0)} = b_k\tilde{A}^k r^{(0)} + \cdots + b_1\tilde{A}^1 r^{(0)} + r^{(0)}. \tag{16}$$

In order to find the coefficients b_k the following function f should be minimized:

$$\begin{aligned} f(b_1, \ldots, b_k) &:= (p_k(\tilde{A})r^{(0)}, p_k(\tilde{A})r^{(0)}) \\ &:= \sum_{i,j=1}^{k} b_i b_j (\tilde{A}^i r^{(0)}, \tilde{A}^j r^{(0)}) + 2\sum_{i=1}^{k} b_i (\tilde{A}^i r^{(0)}, r^{(0)}). \end{aligned} \tag{17}$$

Therefore,

$$\frac{\partial f}{\partial b_i} = 2\sum_{j=1}^{k} b_j (\tilde{A}^i r^{(0)}, \tilde{A}^j r^{(0)}) + 2(\tilde{A}^i r^{(0)}, r^{(0)}) = 0. \tag{18}$$

The coefficients b_k are obtained by solving matrix equation (18).

The solutions of $p_k(\lambda^*) = 0$ will suggest how GMRES optimizes $p_k(\tilde{A})r^{(0)}$; these solutions are calculated in section 3. Then it can be seen whether the polynomial (13) assumed in Theorem 2.1 is appropriate for the problems investigated.

2.2. Three multigrid methods. For a given positive number M, a standard sequence of grids $G^1,\ G^2, \ldots, G^M$ for multigrid is defined as follows:

$$G^M = G, \tag{19}$$

$$G^L = \{(i,j) | (2i-1, 2j-1) \in G^{L+1}\},\ 1 \leq L \leq M-1. \tag{20}$$

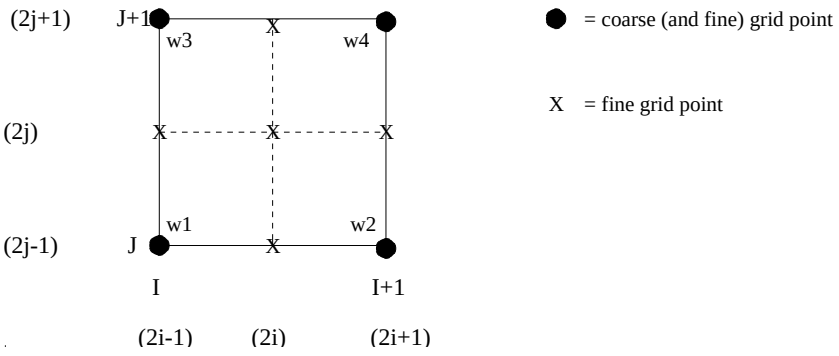

FIG. 3. *One coarse grid cell with the prolongation weights and four fine grid cells for standard multigrid (for coarse grid indices capital letters are used and the fine grid indices are in brackets).*

The three multigrid preconditioners that are implemented and whose performance is compared are now discussed.

2.2.1. Prolongation operators. In the first two algorithms different matrix-dependent prolongation operators are implemented. Figure 3 shows one coarse and four fine grid cells with indices for the explanation of the different prolongation weights. The first operator is well known in the literature ([3], [6]) and has been used before by the authors in [23]. Problems on grids with arbitrary grid sizes can be solved with a similar efficiency. So it is not necessary to choose powers of two (+1) for the grid sizes. The prolongation mapping $P^{L+1}: \ u^L \in Q^{G^L} \mapsto u^{L+1} \in Q^{G^{L+1}}$ with weights $w1$, $w2$, and $w3$ appears as follows:

- for $(2i, 2j-1) \in Q^{G^{L+1}}$

$$d_1 = a^1_{2i,2j-1} + a^4_{2i,2j-1} + a^7_{2i,2j-1},$$
$$d_2 = a^3_{2i,2j-1} + a^6_{2i,2j-1} + a^9_{2i,2j-1},$$
$$d = -(a^2_{2i,2j-1} + a^5_{2i,2j-1} + a^8_{2i,2j-1}),$$
$$w1_{2i,2j-1} = \tfrac{d_1}{d};\ w2_{2i,2j-1} = \tfrac{d_2}{d};$$

- for $(2i-1, 2j) \in Q^{G^{L+1}}$

$$d_1 = a^1_{2i-1,2j} + a^2_{2i-1,2j} + a^3_{2i-1,2j},$$
$$d_2 = a^7_{2i-1,2j} + a^8_{2i-1,2j} + a^9_{2i-1,2j},$$
$$d = -(a^4_{2i-1,2j} + a^5_{2i-1,2j} + a^6_{2i-1,2j}),$$
$$w1_{2i-1,2j} = \tfrac{d_1}{d};\ \ w3_{2i-1,2j} = \tfrac{d_2}{d}.$$

The construction of d differs from the operators in [23] and results in an accurate prolongation, for example, near Dirichlet boundaries, when the grid size is not ideal for multigrid.

On the remaining points the prolongation is found as follows:

(21) on points (2i-1,2j-1) $\qquad u^{L+1}_{2i-1,2j-1} = u^L_{i,j};$

(22) on points (2i,2j) $u^{L+1}_{2i,2j}$ is determined so that $A^{L+1}P^{L+1}u^L = 0$ at (2i,2j).

Some matrix components in the algorithm disappear at boundaries. Therefore, the prolongation is also valid at a boundary. This first algorithm is denoted by $MG1$.

The second algorithm evaluated, denoted by $MG2$, adopts the "upwind" prolongation operator by de Zeeuw ([25]). This operator is also presented in [24] and is specially designed for problems with jumping coefficients and for second-order differential equations with a dominating first-order derivative. As already indicated in [4], it is appropriate for the construction of transfer operators for unsymmetric matrices to split a matrix A into a symmetric and an antisymmetric part:

$$S = \frac{1}{2}(A + A^T), \quad T = A - S = \frac{1}{2}(A - A^T). \tag{23}$$

The investigated transfer operators for $MG2$ are also based on this splitting. The diagonal elements of matrices S and T are numbered similarly as the elements of A in (2). One now defines the following:

$$d_w = \max(|s^1_{2i,2j-1} + s^4_{2i,2j-1} + s^7_{2i,2j-1}|, |s^1_{2i,2j-1}|, |s^7_{2i,2j-1}|), \tag{24}$$

$$d_e = \max(|s^3_{2i,2j-1} + s^6_{2i,2j-1} + s^9_{2i,2j-1}|, |s^3_{2i,2j-1}|, |s^9_{2i,2j-1}|), \tag{25}$$

$$d_n = \max(|s^7_{2i-1,2j} + s^8_{2i-1,2j} + s^9_{2i-1,2j}|, |s^7_{2i-1,2j}|, |s^9_{2i-1,2j}|), \tag{26}$$

$$d_s = \max(|s^1_{2i-1,2j} + s^2_{2i-1,2j} + s^3_{2i-1,2j}|, |s^1_{2i-1,2j}|, |s^3_{2i-1,2j}|), \tag{27}$$

$$\sigma_1 = \frac{1}{2}\min\left(1, \left|1 - \frac{\sum_{k=1}^9 s^k_{2i,2j-1}}{a^5_{2i,2j-1}}\right|\right), \tag{28}$$

$$\sigma_2 = \frac{1}{2}\min\left(1, \left|1 - \frac{\sum_{k=1}^9 s^k_{2i-1,2j}}{a^5_{2i-1,2j}}\right|\right), \tag{29}$$

$$c_1 = t^3_{2i,2j-1} + t^6_{2i,2j-1} + t^9_{2i,2j-1} - (t^1_{2i,2j-1} + t^4_{2i,2j-1} + t^7_{2i,2j-1}), \tag{30}$$

$$c_2 = t^7_{2i-1,2j} + t^8_{2i-1,2j} + t^9_{2i-1,2j} - (t^1_{2i-1,2j} + t^2_{2i-1,2j} + t^3_{2i-1,2j}), \tag{31}$$

$$w_w = \sigma_1\left[1 + \frac{d_w - d_e}{d_w + d_e} + \frac{c_1}{d_w + d_e + d_n + d_s}\right], \tag{32}$$

$$w_e = 2\sigma_1 - w_w, \tag{33}$$

$$w_n = \sigma_2\left[1 + \frac{d_s - d_n}{d_s + d_n} + \frac{c_2}{d_w + d_e + d_n + d_s}\right], \tag{34}$$

$$w_s = 2\sigma_2 - w_n. \tag{35}$$

Weights $w1$ and $w2$ are now found as follows:

- for $(2i, 2j-1) \in Q^{G^{L+1}}$
  $w1_{2i,2j-1} = \min(2\sigma_1, \max(0, w_w))$; $w2_{2i,2j-1} = \min(2\sigma_1, \max(0, w_e))$;
- for $(2i-1, 2j) \in Q^{G^{L+1}}$
  $w1_{2i-1,2j} = \min(2\sigma_2, \max(0, w_s))$; $w3_{2i-1,2j} = \min(2\sigma_2, \max(0, w_n))$.

Equations (21) and (22) are used for the remaining grid points.

Analysis of this operator and the numerical experiments by de Zeeuw [25] show the very interesting behavior of $MG2$ as a solver for several problems.

Restriction. As a restriction operator in $MG1$ and $MG2$, the transpose of the prolongation is used. Restriction $R^L : u^{L+1} \in Q^{G^{L+1}} \mapsto u^L \in Q^{G^L}$ is defined as

$$R^L = \left(P^{L+1}\right)^T, \ 1 \le L \le M - 1. \tag{36}$$

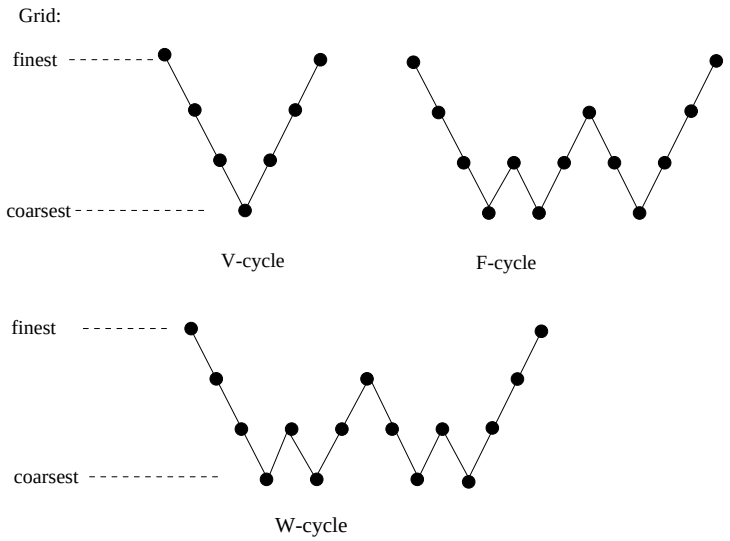

FIG. 4. *Different ways of cycling through fine and coarse grids.*

Coarse grid discretization. Coarse grid matrix A^L on grid G^L is defined with Galerkin coarsening ([6], [24], [25])

$$A^M = A, \tag{37}$$

$$A^L = R^L A^{L+1} P^{L+1},\ 1 \leq L \leq M-1. \tag{38}$$

A drawback of the multigrid preconditioners, compared with, for example, ILU preconditioners, is that additional storage for the matrices on coarse grids as well as for the matrix-dependent transfer operators is needed.

Smoother. The smoother in $MG1$ and $MG2$ is the alternating zebra line Gauss–Seidel smoother. First, all odd (white) lines are processed in one direction, after which all even (black) lines are processed. This procedure takes place in the x- and y-directions. Fourier smoothing analysis for model equations, presented in [24], indicates the robustness of this smoother. The parallel implementation by means of grid partitioning is explained in detail in [23]. Care has been taken that the single block and the multiblock convergences are identical.

Multigrid cycles. The multigrid correction scheme is used for solving the linear equations. In [23] we chose the V-cycle as iteration cycle. Here, the robustness and efficiency of more expensive cycles, like the W- and F-cycles, are also evaluated. The cycles are shown in Figure 4.

The multigrid F-cycle is a cycle between the cheap V-cycle and the expensive W-cycle. The algorithm is written down for convenience in a recursive way.

MG F-cycle

MGF$(A^L, f^L, u^L, \nu_1, \nu_2, \nu_3)$ {
  if $(L = 1)$ {
    $u^1 :=$ smoother(A^1, f^1, u^1, ν_3);
    return;
  }
  $u^L :=$ smoother(A^L, f^L, u^L, ν_1);

$r^L := f^L - A^L u^L$;
$f^{L-1} := R^{L-1} r^L$;
$u^{L-1} := 0$;
$u^{L-1} := \mathrm{MGF}(A^{L-1}, f^{L-1}, u^{L-1}, \nu_1, \nu_2, \nu_3)$;
$u^L := u^L + P^L u^{L-1}$;
$u^L := \mathrm{smoother}(A^L, f^L, u^L, \nu_2)$;
$r^L := f^L - A^L u^L$;
$f^{L-1} := R^{L-1} r^L$;
$u^{L-1} := 0$;
$u^{L-1} := \mathrm{MGV}(A^{L-1}, f^{L-1}, u^{L-1}, \nu_1, \nu_2, \nu_3)$;
$u^L := u^L + P^L u^{L-1}$;
$u^L := \mathrm{smoother}(A^L, f^L, u^L, \nu_2)$;
}

All components in the code have been explained in this section.

2.2.2. Multigrid based on the block LU decomposition. The third algorithm, *MG*3, evaluated as a solver and a preconditioner, was recently proposed by Reusken ([12]). It originates from several publications in which the fine grid matrix A is decomposed ([10], [11]). In matrix A the fine grid points (excluding the coarse grid points) are ordered in the first part, the coarse grid points in the second part. The coefficient matrix on the fine grid is now described as

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}. \tag{39}$$

An approximation M of the above matrix based on the block LU decomposition is made as follows:

$$M = \begin{pmatrix} I_1 & 0 \\ A_{21}{A_{11}}^{-1} & I_2 \end{pmatrix} \begin{pmatrix} A_{11} & A_{12} \\ 0 & S \end{pmatrix}. \tag{40}$$

If S equals the Schur complement S_A $(= A_{22} - A_{21}{A_{11}}^{-1}A_{12})$, M is equal to A. Multigrid methods based on the formulation and splitting from (40) lead to interesting properties of coarse grid matrices. If the fine grid matrix is an M-matrix ([20]), S will also be an M-matrix. In [12], S is an approximate Schur complement constructed from the Schur complement of a modified matrix $\overline{A}$. With the modification based on linear interpolation, A_{11} and A_{12} reduce to $\overline{A}_{11}$ and $\overline{A}_{12}$, where $\overline{A}_{11}$ is diagonal:

$$\overline{A} = \begin{pmatrix} \overline{A}_{11} & \overline{A}_{12} \\ A_{21} & A_{22} \end{pmatrix};\ \overline{A}_{11} = \mathrm{diag}(A_{11}),\ \ \overline{A}_{12} = A_{12} + \mathrm{offdiag}(A_{11})P_{12}. \tag{41}$$

In (41), P_{12} is an operator that interpolates fine grid elements bilinearly to adjacent coarse grid points.

A damping parameter ω is used to correct an incorrect scaling of S compared to S_A. So

$$S = \omega^{-1}(A_{22} - A_{21}\overline{A}_{11}^{-1}\overline{A}_{12}). \tag{42}$$

Another choice for S based on lumping is found in [11].

We construct multigrid cycles based on the Richardson iteration of the splitting of $A = M + (A - M)$ as in (4).

An iteration with M in (40) is described as follows:

1. $r_1 = f_1 - A_{11}u_1^{(k)} - A_{12}u_2^{(k)},\ r_2 = f_2 - A_{21}u_1^{(k)} - A_{22}u_2^{(k)}$.
2. Solve $A_{11}\Delta u_1^* = r_1$.
3. $r_2^* = r_2 - A_{21}\Delta u_1^* = f_2 - A_{21}(u_1^{(k)} + \Delta u_1^*) - A_{22}u_2^{(k)}$.
4. Solve $S\Delta u_2^{(k)} = r_2^*$.
5. $u_2^{(k+1)} = u_2^{(k)} + \Delta u_2^{(k)}$.
6. Solve $A_{11}\Delta u_1^{(k)} = r_1 - A_{12}\Delta u_2^{(k)}$.
7. $u_1^{(k+1)} = u_1^{(k)} + \Delta u_1^{(k)}$.

Vectors u, f, and r are also split according to (39); u_1^* and r_2^* are intermediate variables. Steps 6 and 7 are rewritten as follows:

$$A_{11}(\Delta u_1^{(k)} - \Delta u_1^*) = f_1 - A_{11}(u_1^{(k)} + \Delta u_1^*) - A_{12}(u_2^{(k)} + \Delta u_2^{(k)}),$$

$$u_1^{(k+1)} = (u_1^{(k)} + \Delta u_1^*) + (\Delta u_1^{(k)} - \Delta u_1^*).$$

The algorithm from [12] is obtained if we put

$$u_1^* = u_1^{(k)} + \Delta u_1^*,$$

$$\Delta u_1^{**} = \Delta u_1^{(k)} - \Delta u_1^*.$$

The following equivalent process is obtained:

1. $r_1 = f_1 - A_{11}u_1^{(k)} - A_{12}u_2^{(k)}$.
2. Solve $A_{11}\Delta u_1^* = r_1, u_1^* = u_1^{(k)} + \Delta u_1^*$.
3. $r_2^* = f_2 - A_{21}u_1^* - A_{22}u_2^{(k)}$.
4. Solve $S\Delta u_2^{(k)} = r_2^*$.
5. $u_2^{(k+1)} = u_2^{(k)} + \Delta u_2^{(k)}$.
6. $r_1^* = f_1 - A_{11}u_1^* - A_{12}u_2^{(k+1)}$.
7. Solve $A_{11}\Delta u_1^{**} = r_1^*,\ u_1^{(k+1)} = u_1^* + \Delta u_1^{**}$.

For solving the linear system A_{11}, a certain number of iterations (ν) of an alternating line Jacobi relaxation is performed on the fine grid points excluding the coarse grid points. This operation with right-hand side vector f_1 is described as

$$u_1 := J_{11}(A_{11}, f_1, u_1, \nu). \tag{43}$$

Rewriting the algorithm leads to the algorithm that has been implemented. The multigrid V-, F-, and W-cycles are now constructed with recursive calls replacing step 4 in the algorithms.

The robustness of this algorithm has been presented in [12] by means of two-grid Fourier analysis for the convection-diffusion equation. In the numerical results in [12] parameter ω was set to 0.7. For convenience the MGLU F-cycle is written out in metalanguage.

MGLU F-cycle

```
MGLUF(A^L, f^L, u^L, ν1, ν2, ν3) {
    if (L = 1) {
        u^1 := smoother(A^1, f^1, u^1, ν3);
```

```
        return;
      }
      r_1^L := f_1^L − A_12^L u_2^L;
      u_1^L := J_11(A_11, r_1^L, u_1^L, ν_1);
      r_2^L := f_2^L − A_21^L u_1^L − A_22 u_2^L;
      f^{L−1} := r_2^L ; u^{L−1} := 0;
      u_2^L := u_2^L + MGLUF(S, f^{L−1}, u^{L−1}, ν_1, ν_2, ν_3);
      r_1^L := f_1^L − A_12^L u_2^L;
      u_1^L := J_11(A_11, r_1^L, u_1^L, ν_2);
      r_2^L := f_2^L − A_21^L u_1^L − A_22 u_2^L;
      f^{L−1} := r_2^L; u^{L−1} := 0;
      u_2^L := u_2^L + MGLUV(S, f^{L−1}, u^{L−1}, ν_1, ν_2, ν_3);
      r_1^L := f_1^L − A_12^L u_2^L;
      u_1^L := J_11(A_11, r_1^L, u_1^L, ν_2);
}
```

2.3. MILU preconditioner. We also present results with an ILU preconditioner for the comparison of convergence with the multigrid approaches. Though efficient parallelizations can be achieved by employing parallel ordering of grid points (like red–black ordering), it is well known that the convergence degradation is large compared to the natural order ([5]). In [22] a parallelization of the ILU preconditioner by ignoring interface components between subdomains and by applying the natural ordering in each subdomain is described. However, for our problems, a severe convergence degradation was observed, compared to the single block convergence, because strong couplings of unknowns across grid partitioning interfaces exist. Therefore, in section 3 only single block convergence results of five-point MILU(α_1) preconditioners with an acceleration parameter α_1 ([18]) are given. Elapsed times are not presented.

3. Numerical results. The partial differential equations investigated are 2D singularly perturbed problems. Poisson-type equations and a convection-diffusion equation with a dominating convection term are solved. Except for the first equation, we concentrate on "difficult" problems for multigrid.

The results presented are the number of iterations (n) to reduce the L_2-norm of the initial residual with eight orders of magnitude ($||r^{(n)}||_2/||r^{(0)}||_2 \leq 10^{-8}$). If this criterion is not satisfied in 70 iterations, the asymptotic convergence ρ is presented. Furthermore, the elapsed time $T_{elapsed}$ for this number of iterations obtained on the NEC Cenju–3 MIMD machine ([8]) is presented. For all problems 32 processors are used in a 4×8 configuration.

As the initial guess $\phi^{(0)} = 0$ is used for all problems. After some efficiency tests, we choose no presmoothing and two postsmoothing iterations in $MG1$ and $MG2$. On the coarsest grid two smoothing iterations are performed ($\nu_3 = 2$) in order to keep the method as cheap as possible. For the problems investigated this did not influence convergence negatively, since the coarsest grid is always a 3^2 grid. As the number of J_{11} iterations in $MG3$ varies, it is explicitly mentioned.

Problem I: *An anisotropic diffusion equation.* The first problem investigated is an anisotropic diffusion problem from [12]:

$$-e^{\alpha(1-\frac{1}{x})}\frac{\partial^2\phi}{\partial x^2} - \frac{\partial^2\phi}{\partial y^2} = 1 \text{ on } \Omega = (0,1)\times(0,1). \tag{44}$$

Boundary conditions are chosen as

$$\begin{aligned} \frac{\partial \phi}{\partial n} &= 0 \quad \text{on } \{x = 0,\ 0 \le y \le 1\},\ \{0 \le x \le 1,\ y = 0\}, \\ \phi &= 0 \quad \text{on } \{x = 1,\ 0 \le y \le 1\},\ \{0 \le x \le 1,\ y = 1\}. \end{aligned} \tag{45}$$

This problem with varying anisotropy along x-direction is relatively easy for multigrid methods with an alternating line smoother. Also, Poisson problems on severely stretched grids, which are known to be difficult for ILU preconditioners, are handled very well with our $MG1$ and $MG2$ methods. It is, however, interesting to see the difference in convergence between the three MG methods. Parameter α in (44) was set to one here; other parameter choices lead to similar convergence results.

MG as solver. In Table 1 results are presented for the multigrid methods $MG1$, $MG2$, and $MG3$ used as solvers. The convergence results are obtained with the V-, F- and W-cycles on three consecutive grid sizes. Convergence on a 514^2 grid is also presented to show that for $MG1$ and $MG2$ similar convergence is found on grid sizes that are not ideal for multigrid. For $MG3$ convergence was not found on this grid size because the transfer operators were not suited for irregular sizes. $MG3$ also performs well for this problem on the 514^2 grid, if the linear interpolation P_{12} for the modification of A in (41) is changed to a matrix-dependent interpolation with weights from $MG1$ or $MG2$.

Grid-independent convergence is found for all cycles with $MG1$ and $MG2$. Their performance is similar for this test problem. The convergence of $MG3$ is slower than that of the other algorithms but is in accordance with the performance in [12].

It was found that $MG3$ as a solver is fairly sensitive in the variation of parameter ω from (42); optimal for this problem is $\omega = 0.75$. In order to obtain grid-independent convergence the number of J_{11} steps grows for $MG3$ as the problem grows: on a 129^2 grid $J_{11}(2,4)$ (two pre- and four post-line Jacobi steps) was sufficient; on the 257^2 grid $J_{11}(3,5)$ steps were necessary, and on a 513^2 grid $J_{11}(4,6)$ steps were necessary. The condition number of A_{11} grows as the grid size becomes larger.

Furthermore, it can be seen in Table 1 that a W-cycle is more expensive than an F-cycle on a parallel machine. This is due to the fact that coarse grids are processed more often and that an agglomeration strategy ([7]) is not incorporated. On coarse grid levels some processors are idle, and the other processors contain a very small number of grid points. Communication is then dominant. On one processor the difference between the F- and W-cycle is less pronounced.

MG as preconditioner. In Tables 2 and 3 the results are presented with $MG1$, $MG2$, and $MG3$ as preconditioners. In Table 2 BiCGSTAB is the solution method, in Table 3 GMRES(20) is the solution method. Comparison of these tables shows that the elapsed times for the same preconditioning cycle are in the same range; there is not much difference between the performance of BiCGSTAB and GMRES(20). The number of iterations as well as the elapsed time are reduced compared to Table 1 for many cases. Also, in Tables 2 and 3 grid-independent convergence is found with all cycles as preconditioners. It is interesting to see that the performance of $MG3$ as a preconditioner is much better and more robust than that of $MG3$ as a solver. For this problem it was also possible to choose a V-cycle for $MG3$ as a preconditioner, and the number of pre- and post-Jacobi steps could be reduced by two for each grid size. With $MG3$ as a preconditioner the methods also converge on the 514^2 grid with linear interpolation P_{12} in (41), i.e., the original method from [12].

MILU preconditioner. Problem (44) is also solved with the MILU(α_1) preconditioned Krylov methods. The convergence appeared to be very sensitive with respect

TABLE 1
Multigrid as solver: number of solution iterations (n) and elapsed time $T_{elapsed}$ in seconds for the anisotropic diffusion equation ($\alpha = 1$) on several grid sizes.

Grid:		129^2	257^2	513^2	514^2
Method:	Cycle:				
*MG*1	V	(11) 1.4	(11) 2.0	(11) 4.1	(11) 4.1
	F	(8) 2.7	(8) 3.9	(8) 6.7	(5) 4.2
	W	(8) 7.3	(8) 15.1	(8) 31.9	(5) 20.2
*MG*2	V	(9) 1.1	(9) 1.8	(9) 4.2	(11) 4.3
	F	(7) 2.3	(7) 3.7	(7) 6.8	(9) 8.0
	W	(7) 6.5	(7) 13.4	(7) 28.9	(9) 36.6
*MG*3	V	$\rho = 0.82$	DIV	DIV	DIV
	F	(29) 13.5	(34) 29.3	(40) 57.9	DIV
	W	(19) 23.6	(17) 55.4	(16) 133.5	DIV

TABLE 2
Multigrid as preconditioner for BiCGSTAB: number of solution iterations (n) and elapsed time $T_{elapsed}$ in seconds for the anisotropic diffusion equation ($\alpha = 1$) on several grid sizes.

Grid:		129^2	257^2	513^2	514^2
Method:	Cycle:				
*MG*1	V	(5) 1.1	(4) 1.4	(4) 2.9	(4) 3.0
	F	(4) 2.6	(4) 4.0	(4) 6.7	(3) 4.3
	W	(4) 7.3	(4) 15.1	(4) 32.0	(3) 24.2
*MG*2	V	(4) 0.9	(4) 1.4	(4) 2.9	(4) 2.9
	F	(3) 1.9	(3) 2.9	(3) 5.1	(3) 5.1
	W	(3) 5.4	(3) 11.3	(3) 24.0	(3) 24.2
*MG*3	V	(16) 3.8	(23) 7.8	(30) 28.9	(33) 40.7
	F	(10) 6.5	(10) 13.4	(10) 28.5	(16) 46.1
	W	(6) 14.9	(7) 45.7	(7) 96.2	(12) 167.7

TABLE 3
Multigrid as preconditioner for GMRES(20): number of solution iterations (n) and elapsed time $T_{elapsed}$ in seconds for the anisotropic diffusion equation ($\alpha = 1$) on several grid sizes.

Grid:		129^2	257^2	513^2	514^2
Method:	Cycle:				
*MG*1	V	(7) 0.8	(8) 1.5	(8) 3.1	(7) 2.8
	F	(6) 2.0	(6) 3.0	(6) 5.2	(5) 4.3
	W	(6) 5.5	(6) 11.5	(6) 24.1	(5) 20.3
*MG*2	V	(7) 0.9	(7) 1.3	(7) 2.7	(7) 2.7
	F	(6) 2.0	(6) 3.0	(6) 5.2	(6) 5.2
	W	(6) 5.5	(6) 11.4	(6) 24.1	(6) 24.3
*MG*3	V	(29) 4.1	(39) 7.7	(58) 31.9	(69) 47.5
	F	(18) 6.0	(18) 12.4	(18) 26.7	(31) 46.7
	W	(12) 15.0	(12) 40.0	(13) 90.2	(22) 155.7

to parameter α_1. For this problem the best results were obtained with $\alpha_1 = 1$ and BiCGSTAB. On a 129^2 grid 54 iterations were needed; on a 257^2 grid, 91 iterations, on a 513^2 grid, 126 iterations, and on a 514^2 grid, 130 iterations. With GMRES(50)

the following results were obtained: on a 129^2 grid, 62 iterations; on a 257^2 grid, 139 iterations, on a 513^2 grid, 444 iterations, and on a 514^2 grid, 446 iterations. The number of iterations grows rapidly for larger grid sizes.

Problem II: *Rotating convection-diffusion equation.* The next problem is a rotating convection-diffusion problem, presented in [21] and also tested in [23],

$$-\epsilon\Delta\phi + a(x,y)\frac{\partial\phi}{\partial x} + b(x,y)\frac{\partial\phi}{\partial y} = 1 \text{ on } \Omega = (0,1)\times(0,1). \tag{46}$$

Here, $a(x,y) = -\sin(\pi x).\cos(\pi y)$, $b(x,y) = \sin(\pi y).\cos(\pi x)$.

Dirichlet boundary conditions are prescribed: $\phi|_\Gamma = \sin(\pi x) + \sin(13\pi x) + \sin(\pi y) + \sin(13\pi y)$.

A convection dominated test case is chosen: $\epsilon = 10^{-5}$. The convection terms are discretized with a standard upwind discretization, due to the fact that the multigrid methods with the chosen smoothers are not convergent methods for matrices resulting from a second-order discretization. In a forthcoming paper, possibly part III, we will concentrate on second-order accuracy for convection-diffusion problems.

Spectrum analysis. For this test problem we investigate the eigenvalue spectrum of the Richardson iteration matrix $I - K^{-1}A$ (5). For K^{-1} in (5) all MG methods are compared on a 33^2 grid. The spectra are presented in Figure 5. As can be seen, most eigenvalues are clustered around zero for $MG1$ and $MG2$ but not for $MG3$. For $MG3$ this has also been observed in [12]. Furthermore, a difference between the distribution of $MG1$ and $MG2$ can be seen; the spectral radius of $MG1$ is larger. The actual convergence with these methods as solvers and preconditioners on the 33^2 grid is presented in Figure 6. Also, if it was obtained, the observed asymptotic convergence ρ is presented. If an asymptotic convergence was not found, the average convergence $\rho^{(n)} = (r^{(n)}/r^{(0)})^{\frac{1}{n}}$ is shown. Comparing Figures 5 and 6 we see that the spectral radius determines the convergence of the MG methods as solvers, as expected.

For this problem we study the development of the solutions λ_k^* of the minimal residual polynomial $p_k(\lambda_k^*) = 0$ from (16). Solutions $1-\lambda_k^*$ of the last five polynomials until convergence are presented in Table 4 for $MG1$ and $MG2$. Due to the fact that $1-\lambda^*$ is presented, it is possible to compare Table 4 and the eigenvalues in Figure 5. It can be clearly seen that the vectors belonging to the most distant eigenvalues of $I - AK^{-1}$ from zero are first captured in the Krylov space. The development of this space gives a clear view of the search directions of GMRES. For $MG1$ six eigenvalues with largest distance from zero are obtained from the ninth polynomial, for $MG2$ four are found after seven iterations. So the choice of $s_k(\lambda)$ in (13) may give a nearly optimal upper bound.

MG as solver and preconditioner. Tables 5, 6, and 7 present convergence results obtained on three different grid sizes, 129^2, 257^2, and 513^2. The MG methods are used as solvers in Table 5, as preconditioners for BiCGSTAB in Table 6, and for GMRES(20) in Table 7. For $MG3$ the number of J_{11} steps is as in Problem I, with less steps as a preconditioner than as a solver. Here, matrix-dependent interpolations in (41) did not result in a robust $MG3$ method. It can be seen that $MG2$ handles this test case with dominating convection better than the other methods. With the expensive W-cycle almost grid-independent convergence is found. Very satisfactory convergence associated with small elapsed times is found in most cases with the F-cycle. With the V-cycle used as a preconditioner the best elapsed times are found, but the number of iterations is increasing for large grid sizes. Again, the convergence is fastest when the MG methods are used as preconditioners.

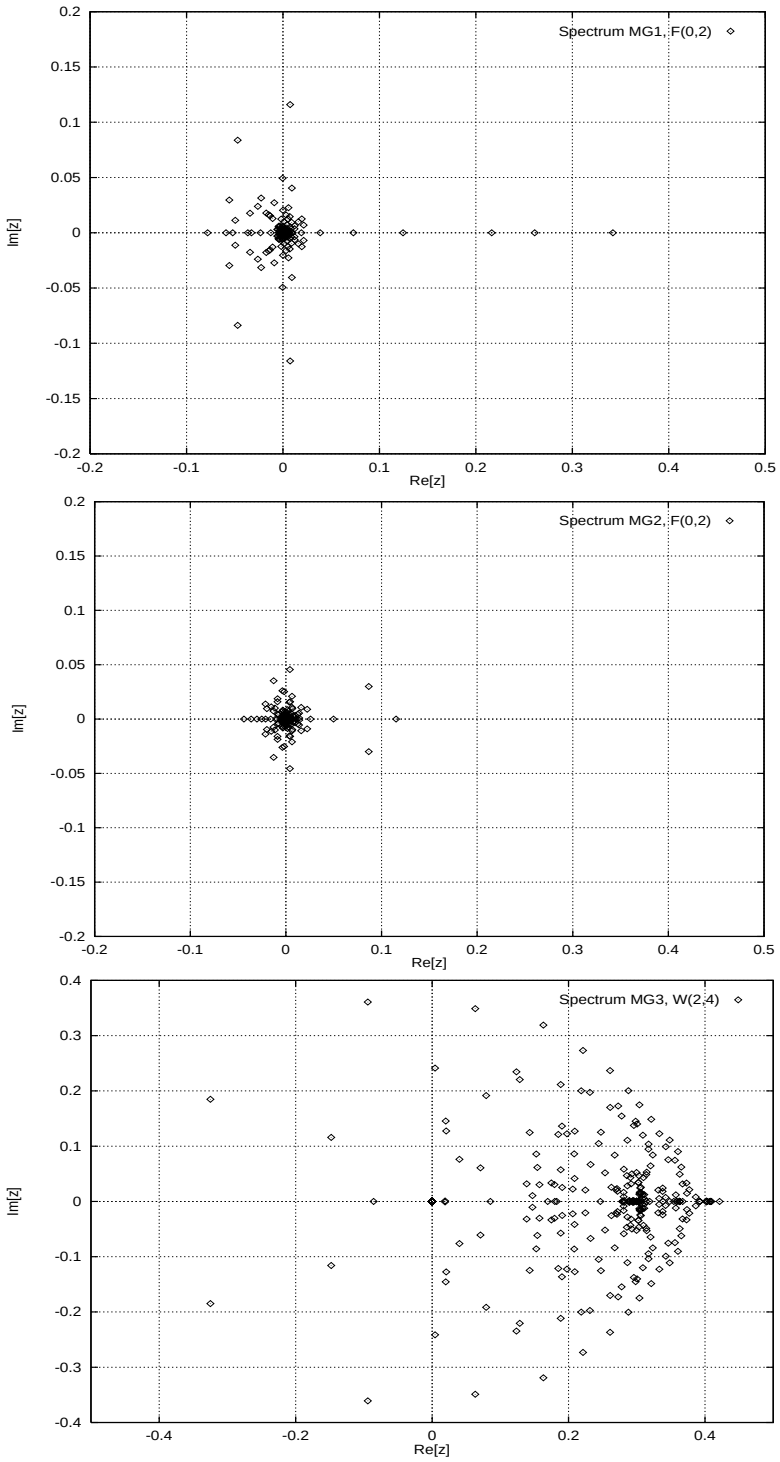

FIG. 5. *The eigenvalue spectra for the rotating convection-diffusion problem,* $\epsilon = 10^{-5}$ *on a* 33^2 *grid, with all MG methods. Note the different scaling.*

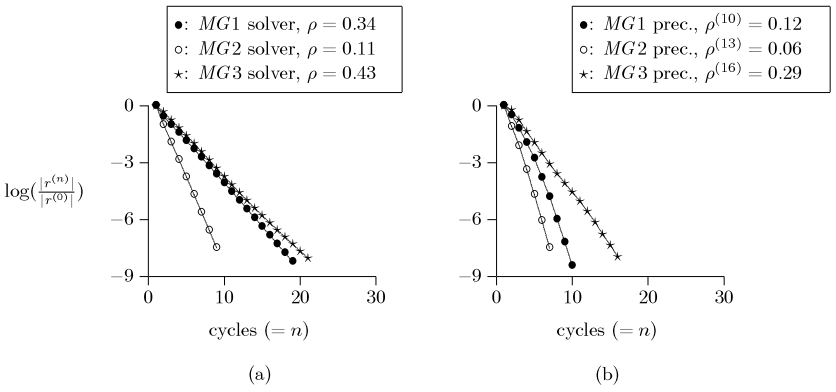

FIG. 6. *The convergence of the MG solvers* (a), *and GMRES*(20) *with MG preconditioners* (b), *for the rotating convection-diffusion equation on a* 33^2 *grid.*

TABLE 4

Solutions $1-\lambda_k^*$ *related to the polynomials* p_k *chosen by GMRES with MG*1 *and MG*2 *for the convection-diffusion equation* ($\epsilon = 10^{-5}$).

k:	5	6	7	8	9
	0.32	0.33	0.34	0.34	0.34
*MG*1	0.25	0.28	$0.24 + 1.9.10^{-2}i$	0.27	0.26
	0.14	0.18	$0.24 - 1.9.10^{-2}i$	0.21	0.21
$1-\lambda_k^*$	$2.4.10^{-2}$	$-4.3.10^{-2}$	$-5.4.10^{-2}$	$-6.4.10^{-2}$	$-8.5.10^{-3}$
	$-1.7.10^{-2}$	$6.6.10^{-2}$	0.12	0.12	0.13
		$1.6.10^{-2}$	$2.5.10^{-2}$	$1.7.10^{-2}$	$1.7.10^{-2}$
			$-1.1.10^{-2}$	$5.9.10^{-2}$	$6.4.10^{-2}$
				$-5.3.10^{-2}$	$-5.9.10^{-2} + 2.3.10^{-2}i$
					$-5.9.10^{-2} - 2.3.10^{-2}i$

k:	4	5	6	7
	$8.5.10^{-2} + 1.8.10^{-2}i$	$9.0.10^{-2} + 1.6.10^{-2}i$	$9.1.10^{-2}$	0.14
*MG*2	$8.5.10^{-2} - 1.8.10^{-2}i$	$9.0.10^{-2} - 1.6.10^{-2}i$	$9.1.10^{-2} + 2.2.10^{-2}i$	$9.5.10^{-2} + 3.5.10^{-2}i$
	$-2.8.10^{-2}$	$-2.4.10^{-2}$	$9.1.10^{-2} - 2.2.10^{-2}i$	$9.5.10^{-2} - 3.5.10^{-2}i$
$1-\lambda_k^*$	$9.0.10^{-3}$	$4.4.10^{-2}$	$-1.0.10^{-2} + 1.6.10^{-2}i$	$-1.3.10^{-2} + 1.7.10^{-2}i$
		$7.9.10^{-3}$	$-1.0.10^{-2} - 1.6.10^{-2}i$	$-1.3.10^{-2} - 1.7.10^{-2}i$
			$3.3.10^{-3}$	$6.6.10^{-2}$
				$8.4.10^{-3}$

MILU preconditioner. The MILU(α_1) preconditioner converges satisfactorily with $\alpha_1 = 0.95$. Here, GMRES(50) did not lead to convergence. For BiCGSTAB, the number of iterations to reduce the initial residual with eight orders of magnitude were as follows: for 129^2 grid, 32 iterations; for 257^2 grid, 58 iterations; for 513^2 grid, 133 iterations.

Problem III: *The rotated anisotropic diffusion equation.* The third equation investigated is the well-known rotated anisotropic diffusion equation

(47)
$$-(\cos^2\beta + \epsilon\sin^2\beta)\frac{\partial^2\phi}{\partial x^2} - 2(\epsilon-1)\cos\beta\sin\beta\frac{\partial^2\phi}{\partial x\partial y} - (\epsilon\cos^2\beta + \sin^2\beta)\frac{\partial^2\phi}{\partial y^2} = 1 \quad \text{on } (0,1)\times(0,1).$$

TABLE 5

Multigrid as solver: number of solution iterations (n) and elapsed time $T_{elapsed}$ in seconds for the rotating convection-diffusion equation on several grid sizes.

Grid:		129^2	257^2	513^2
Method:	Cycle:			
*MG*1	V	$\rho = 0.78$	$\rho = 0.88$	$\rho = 0.94$
	F	(41) 14.2	(58) 29.1	$\rho = 0.78$
	W	(36) 33.8	(43) 83.0	(50) 204.8
*MG*2	V	(29) 3.5	(54) 9.7	$\rho = 0.81$
	F	(15) 5.1	(20) 10.1	(29) 24.8
	W	(13) 12.1	(15) 28.9	(16) 65.4
*MG*3	V	$\rho = 0.90$	DIV	DIV
	F	(49) 20.6	(58) 39.2	(64) 66.4
	W	(35) 36.3	(42) 108.0	(46) 202.9

TABLE 6

Multigrid as preconditioner for BiCGSTAB: number of solution iterations (n) and elapsed time $T_{elapsed}$ in seconds for the rotating convection-diffusion equation on several grid sizes.

Grid:		129^2	257^2	513^2
Method:	Cycle:			
*MG*1	V	(14) 3.4	(22) 7.8	(36) 26.3
	F	(11) 7.3	(12) 12.1	(16) 27.2
	W	(10) 18.9	(12) 46.5	(13) 106.7
*MG*2	V	(8) 1.9	(12) 4.2	(19) 13.9
	F	(6) 3.5	(7) 7.0	(9) 15.3
	W	(5) 9.3	(6) 23.2	(7) 57.4
*MG*3	V	(36) 13.3	(54) 19.1	>70
	F	(15) 11.1	(19) 18.3	(21) 43.1
	W	(12) 24.2	(13) 66.9	(14) 97.2

TABLE 7

Multigrid as preconditioner for GMRES(20): number of solution iterations (n) and elapsed time $T_{elapsed}$ in seconds for the rotating convection-diffusion equation on several grid sizes.

Grid:		129^2	257^2	513^2
Method:	Cycle:			
*MG*1	V	(23) 3.0	(40) 7.9	$\rho = 0.80$
	F	(19) 6.5	(23) 12.1	(30) 27.0
	W	(18) 17.1	(22) 43.3	(25) 104.1
*MG*2	V	(14) 1.8	(20) 4.1	(40) 17.0
	F	(10) 3.5	(12) 6.1	(16) 14.3
	W	(10) 9.5	(11) 21.6	(12) 49.8
*MG*3	V	$\rho = 0.7$	DIV	DIV
	F	(29) 13.7	(34) 16.9	(39) 42.4
	W	(23) 24.1	(24) 63.0	(27) 96.6

Boundary conditions used are as in Problem I (45). Parameters ϵ and β are to be varied. The mixed derivative is approximated with four grid points. A nine-point stencil results from the discretization of all terms in (48). Multigrid convergence is slow, for example, for parameters $\epsilon = 10^{-5}$ and $\beta = 135^o$.

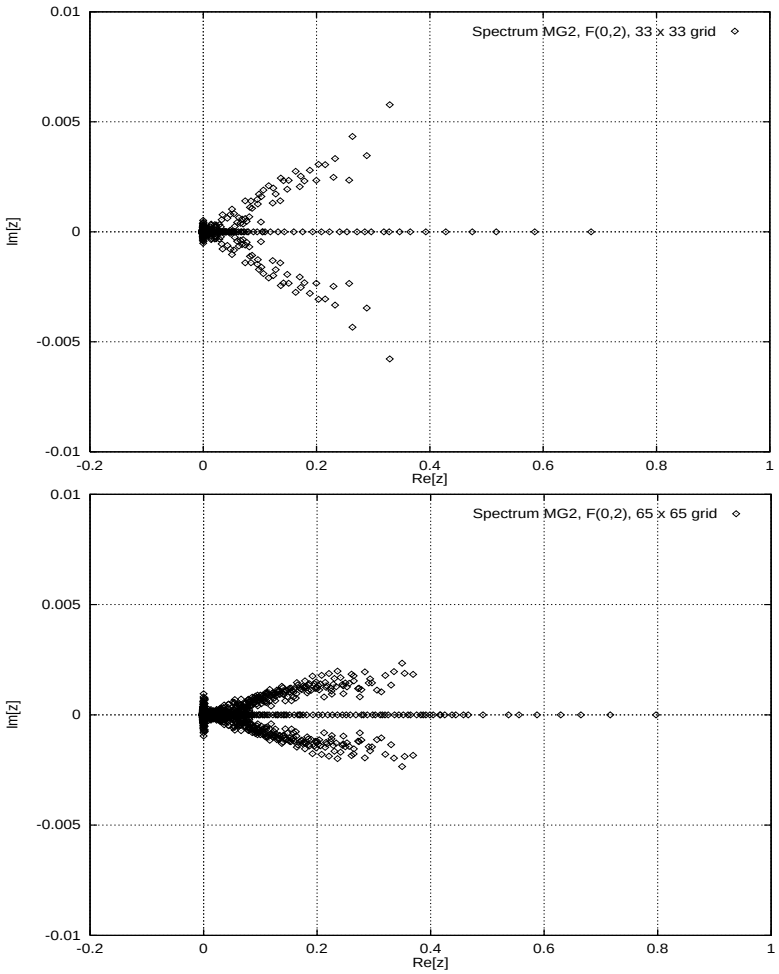

FIG. 7. *The eigenvalue spectra for the rotated anisotropic diffusion problem,* $\epsilon = 10^{-5}$, $\beta = 135^o$ *on a* 33^2 *and a* 65^2 *grid, with* $MG2$ $F(0,2)$.

Spectrum analysis. For these parameters we also calculated the eigenvalue spectrum of the Richardson iteration matrix (5) for a 33^2 and a 65^2 problem. The spectra presented in Figure 7 are obtained with $MG2$ and the F(0,2)-cycle. They are almost identical to the spectra obtained with $MG1$. With the two eigenvalue pictures in Figure 7 it is possible to observe the mesh dependency of the multigrid convergence. It can be seen, for example, that the spectral radius is increasing as the grid gets finer. The spectral radius is, for these coarse grid problems, already larger than 0.6. Therefore, the multigrid convergence slows down more dramatically than the convergence of the preconditioned Krylov methods. Many eigenvalues are clustered around zero and only a limited number of eigenvalues are larger than 0.4 for both grid sizes, so eigenvalues of the preconditioned matrix (AK^{-1}) are clustered around one, which is advantageous for the Krylov methods. The convergence of $MG1$, $MG2$, and $MG3$ as solution methods (Figure 8(a)) and as preconditioners for GMRES(20) (Figure 8(b)) is presented for the 33^2 grid. Also, for this problem, it is interesting to examine the way GMRES minimizes the residual to find the solutions of the different

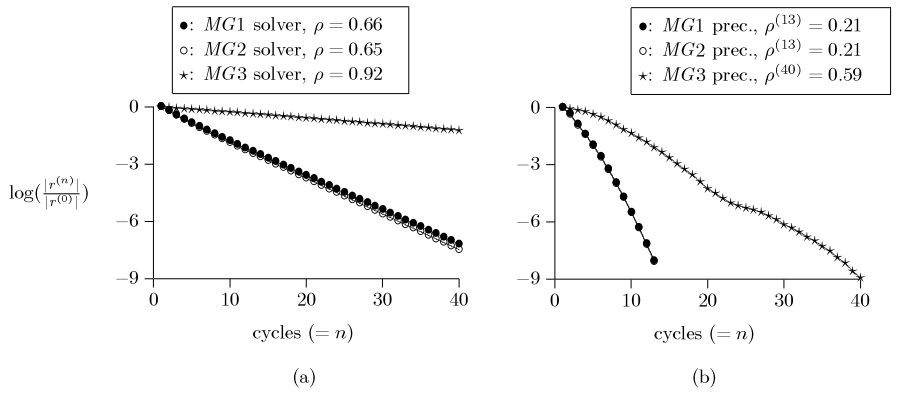

FIG. 8. *The convergence of the MG solvers* (a), *and GMRES*(20) *with MG preconditioners* (b), *for the rotated anisotropic diffusion equation on a* 33^2 *grid.*

TABLE 8
Number of BiCGSTAB and GMRES(20) *iterations* (n) *and elapsed time* $T_{elapsed}$ *in seconds for the rotated anisotropic diffusion equation* ($\epsilon = 10^{-5}$, $\beta = 135^o$).

Grid:		257^2		513^2		769^2	
Method:	Cycle:	BiCGSTAB	GMRES	BiCGSTAB	GMRES	BiCGSTAB	GMRES
*MG*2	F	(17) 17.0	(31) 13.1	(21) 35.6	(43) 39.0	(25) 69.2	(48) 71.2
	W	(10) 38.9	(19) 37.2	(12) 98.4	(20) 83.3	(13) 196.4	(22) 168.8

polynomials (16) of GMRES. It was found that, indeed, the first eigenvalues identified corresponded to the eigenvalues with the largest distance from zero in Figure 7. Six of these eigenvalues were solutions of the tenth polynomial for the 33^2 problem.

MG as solver and preconditioner. On finer grids the multigrid convergence rates further increased towards one, while the convergence with the W-cycle preconditioners of *MG*1 and *MG*2 was level independent.

Table 8 presents the convergence of BiCGSTAB and GMRES(20) with *MG*2 F- and W-cycles as preconditioners on three very fine grid sizes. Results with *MG*1 were similar to those from Table 8; with *MG*3 convergence was more slow.

MILU preconditioner. Satisfactory convergence was not found with the MILU(α_1) preconditioners for this test problem on these grid sizes.

Problem IV: *An interface problem.* Next, an interface problem is considered. This type of problem has been investigated with multigrid in [1], [3], [16], and [25]. The problem to be solved is presented in [23] and appears as follows:

$$\frac{\partial}{\partial x} D_1 \frac{\partial \phi}{\partial x} + \frac{\partial}{\partial y} D_2 \frac{\partial \phi}{\partial y} = b \ \text{ on } \Omega = (0,1) \times (0,1). \tag{48}$$

Right-hand side b is chosen as follows: $b(\frac{1}{4}, \frac{1}{4}) = b(\frac{1}{2}, \frac{1}{2}) = b(\frac{3}{4}, \frac{3}{4}) = 10$, elsewhere $b(x, y) = 0$.

Dirichlet conditions are used:

$$\phi = 1 \text{ on } \left\{ x \leq \frac{1}{2} \wedge y = 0 \right\} \text{ and on } \left\{ x = 0 \wedge y \leq \frac{1}{2} \right\}; \text{ elsewhere } \phi = 0. \tag{49}$$

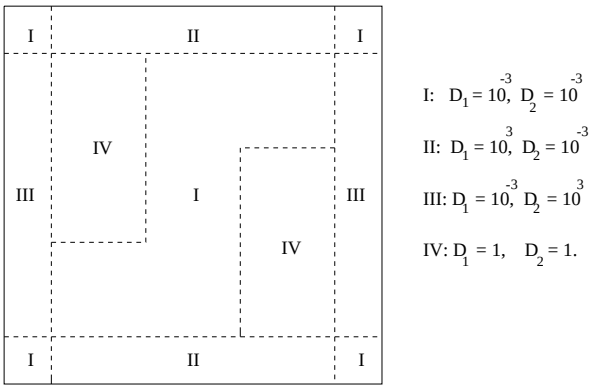

FIG. 9. *The domain with the jumping diffusion coefficients.*

The computational domain with the values of the jumping diffusion coefficients D_1 and D_2 is presented in Figure 9. The discretization is vertex centered and all diffusion coefficients are assumed in the vertices. For the discretized diffusion coefficient between two vertices the harmonic average of the neighboring coefficients is then taken.

Clearly our multigrid methods can solve many interface problems, presented, for example, in [3] or [25]. Here, we constructed a difficult problem, where the robust components of $MG1$, $MG2$, and $MG3$ are not satisfactory. The Krylov acceleration is really needed for convergence. AMG ([13]), however, solves this problem in approximately 15 iterations on different grid sizes.

Spectrum analysis. The eigenvalue spectrum obtained with $MG2$ is presented in Figure 10. Also, for this problem the performance of $MG1$ and $MG2$ was almost identical. In Figure 10 we see two eigenvalues close to one, so multigrid convergence is already very slow on this coarse grid. With two eigenvalues of the preconditioned matrix close to zero, this is an example mentioned in Remark 2.1.1. The convergence of GMRES(20) with $MG1$, $MG2$, and $MG3$ as preconditioners on the 33^2 grid and of GMRES(30) on a 129^2 grid is shown in Figure 11. On the finer grid the restart parameter is raised so that the convergence for this example is not disturbed by restarts. Multigrid as a preconditioner is converging well. In Figure 11(a) it can be seen that for the problem on a 33^2 grid the first nine GMRES iterations are not reducing the residual very much, but after iteration nine fast convergence is obtained. This can also be observed after the calculation of the solutions $1 - \lambda_i^*$ related to the GMRES minimal residual polynomial. These solutions of the eighth and ninth polynomial are presented in Table 9. Here, it can be seen that the vector belonging to a second eigenvalue of $I - AK^{-1}$ around one is obtained in the Krylov space in the ninth iteration, and then GMRES starts converging very rapidly. In the polynomial for the interface problem discretized on a 129^2 grid, a second eigenvalue near one was seen first as solution of polynomial 15, and it was developing even more towards one in polynomial 16. In Figure 11(b) it can be seen that the GMRES convergence for the 129^2 problem starts after 15 iterations.

MG as solver and preconditioner. For this test problem the convergence of the preconditioned Krylov methods with the $MG2$ preconditioner on three very fine grids is presented. The number of BiCGSTAB and GMRES(20) iterations (n) and the elapsed time $T_{elapsed}$ are presented in Table 10. The GMRES convergence is influenced

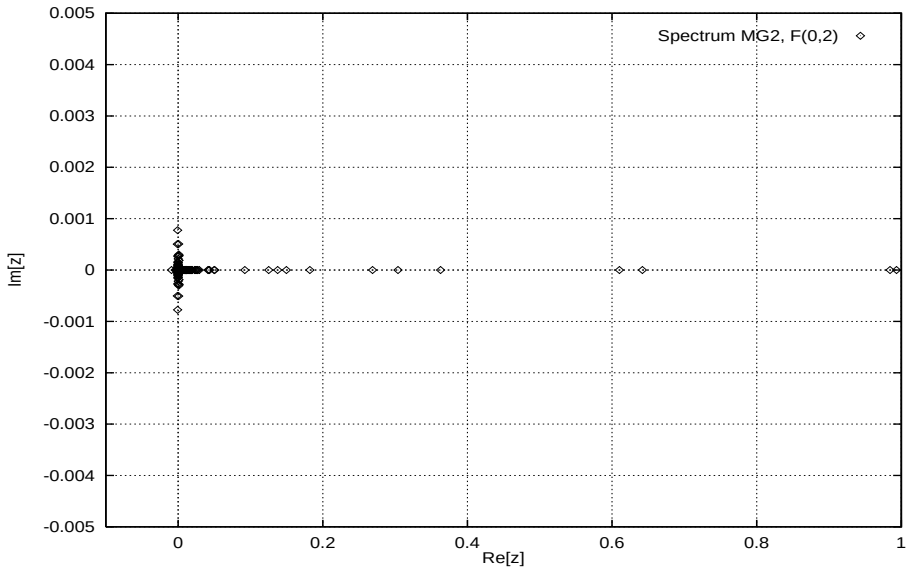

FIG. 10. *The eigenvalue spectrum for the interface problem on a* 33^2 *grid, with* $MG2$ $F(0,2)$.

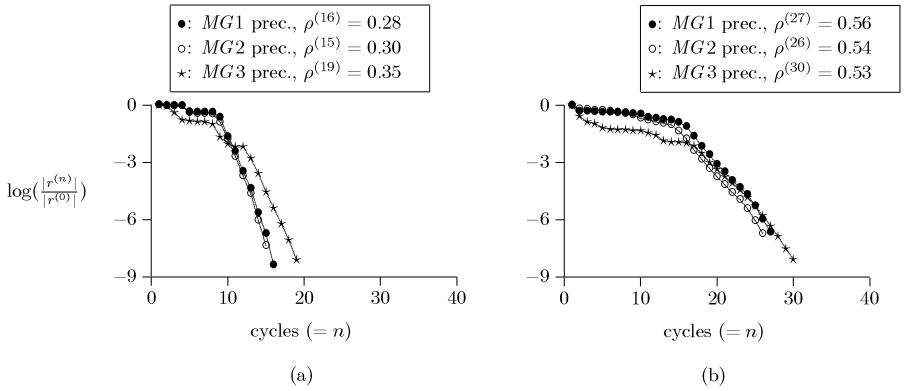

FIG. 11. *The convergence of GMRES*(20) *with MG preconditioners* (a), 33^2 *grid,* (b) 129^2 *grid.*

by the fact that the restart parameter is 20; a larger parameter results in faster convergence.

Again, the F-cycle is preferred for its robustness and efficiency. From Table 10 it can be seen that with a cheap cycle, like the V-cycle, BiCGSTAB is more favorable than GMRES(20), due to the restarts. On finer grids the number of solution iterations decreases, which is due to a better initial residual.

MILU preconditioner. For interface problem (48) MILU(α_1) performed best with $\alpha_1 = 1$. On fine grids the number of iterations found with BiCGSTAB were as follows: 129^2 grid, 89 iterations; 257^2 grid, 124 iterations; 513^2 grid, 210 iterations; 769^2 grid, 295 iterations. The increase in the number of iterations on fine grids is large.

4. Conclusion. In the present work three multigrid methods have been evaluated as solvers and as preconditioners for BiCGSTAB and GMRES. The problems investigated were singularly perturbed. The behavior of the multigrid methods was

TABLE 9
Solutions $1 - \lambda_k^*$ related to the eighth and ninth polynomial constructed in GMRES for the interface problem, 33^2 grid.

k:	8	9
$1 - \lambda_k^*$	0.99	0.99
	0.73	0.98
	$0.74 + 0.19i$	0.63
	$0.74 - 0.19i$	0.62
	0.29	0.33
	$8.3.10^{-2}$	0.15
	$3.8.10^{-2}$	0.11
	$5.5.10^{-4}$	$7.9.10^{-3}$
		$-3.9.10^{-4}$

TABLE 10
Number of BiCGSTAB and GMRES(20) iterations (n) and elapsed time $T_{elapsed}$ in seconds for the interface problem.

Grid:		257^2		513^2		769^2	
Method:	Cycle:	BiCGSTAB	GMRES	BiCGSTAB	GMRES	BiCGSTAB	GMRES
*MG*2	V	(28) 9.8	(62) 12.1	(27) 19.8	$\rho^{(70)} = 0.84$	(24) 35.0	$\rho^{(70)} = 0.83$
	F	(23) 22.8	(40) 20.8	(19) 32.4	(38) 34.6	(18) 49.9	(35) 51.8
	W	(18) 70.0	(32) 62.7	(15) 122.9	(29) 120.8	(16) 241.1	(22) 168.7

much more robust when they were used as preconditioners, since then the convergence was not sensitive to parameter changes. Problems that could not be solved with the multigrid methods as solvers could be solved with the preconditioned Krylov methods. The most efficient results were also obtained when the methods were used as preconditioners. For several problems level-independent convergence rates were found with the W-cycle and best elapsed times were found with the V-cycle. The F-cycle is preferred, however, since it is robust and efficient. For good efficiency with the W-cycle on an MIMD machine, an agglomeration strategy is a necessary requirement.

The convergence behavior can be well understood by investigating the eigenvalue spectrum of the iteration matrix and the solutions of the GMRES minimal residual polynomials. From these investigations and the convergence behavior on fine grids, it was found that methods based on standard multigrid, *MG*1 and *MG*2, are more robust for the problems investigated than *MG*3 based on an approximate Schur complement. In particular, *MG*2 performed very satisfactorily with transfer operators based on the splitting of a matrix into symmetric and antisymmetric parts.

Acknowledgment. Correction of the English text by Mr. R. Calkin is gratefully acknowledged.

REFERENCES

[1] S.F. ASHBY, R.D. FALGOUT, S.G. SMITH, AND T.W. FOGWELL, *Multigrid Preconditioned Conjugate Gradients for the Numerical Simulation of Groundwater Flow on the Cray T3D*, Tech. report UCRL-JC-118622, Lawrence Livermore National Laboratory, Livermore, CA, 1994.

[2] P. BASTIAN, *Parallele adaptive Mehrgitterverfahren*, Ph.D. thesis, University of Heidelberg, Bericht n94/1 Heidelberg, Germany, 1994 (in German).

[3] J.E. Dendy Jr., *Black box multigrid*, J. Comput. Phys., 48 (1982), pp. 366–386.
[4] J.E. Dendy Jr., *Black box multigrid for nonsymmetric problems*, Appl. Math. Comput., 13 (1983), pp. 261–283.
[5] I.S. Duff and G.A. Meurant, *The effect of ordering on preconditioned conjugate gradients*, BIT, 29 (1989), pp. 635–657.
[6] W. Hackbusch, *Multi-grid Methods and Applications*, Springer, Berlin, 1985.
[7] R. Hempel and A. Schüller, *Experiments with Parallel Multigrid Algorithms Using the SUPRENUM Communications Subroutine Library*, GMD Arbeitspapier 141, GMD St. Augustin, Germany, 1988.
[8] R. Hempel, R. Calkin, R. Hess, W. Joppich, U. Keller, N. Koike, C.W. Oosterlee, H. Ritzdorf, T. Washio, P. Wypior, and W. Ziegler, *Real applications on the new parallel system NEC Cenju–3*, Parallel Comput., 22 (1996), pp. 131–148.
[9] R. Kettler, *Analysis and comparison of relaxation schemes in robust multigrid and preconditioned conjugate gradient methods*, in Multigrid Methods, Lecture Notes in Math. 960, W. Hackbusch and U. Trottenberg, eds., Springer, Berlin, 1982, pp. 502–534.
[10] A. Reusken, *Multigrid with matrix-dependent transfer operators for a singular perturbation problem*, Computing, 50 (1993), pp. 199–211.
[11] A. Reusken, *Multigrid with matrix-dependent transfer operators for convection-diffusion problems*, in Multigrid Methods IV, P. Hemker and P. Wesseling, eds., Internat. Ser. of Numer. Math. 116, Birkhäuser, Basel, Switzerland, 1991, pp. 269–280.
[12] A. Reusken, *A Multigrid Method Based on Incomplete Gaussian Elimination*, Tech. report RANA 95–13, University of Eindhoven, the Netherlands, 1995.
[13] J. Ruge and K. Stüben, *Algebraic multigrid (AMG)*, in Multigrid Methods, S. McCormick, ed., Frontiers in Appl. Math. 5, SIAM, Philadelphia, PA, 1987, pp. 73–130.
[14] Y. Saad and M.H. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Comput., 7 (1986), pp. 856–869.
[15] K. Stüben and U. Trottenberg, *Multigrid methods: fundamental algorithms, model problem analysis and applications*, in Multigrid Methods, W. Hackbusch and U. Trottenberg, eds., Lecture Notes in Math. 960, Springer, Berlin, 1982, pp. 1–17.
[16] O. Tatebe, *The multigrid preconditioned conjugate gradient method*, in Proc. Sixth Copper Mountain Conference on Multigrid Methods, NASA Conference Publication 3224, Hampton, VA, 1993, pp. 621–634.
[17] O. Tatebe and Y. Oyanagi, *Efficient implementation of the multigrid preconditioned conjugate gradient method on distributed memory machines*, in Proc. of Supercomputing '94, IEEE Computer Society, Los Alamitos, CA, 1994, pp. 194-203.
[18] H.A. van der Vorst, *High performance preconditioning*, SIAM J. Sci. Comput., 10 (1989), pp. 1174–1185.
[19] H.A. Van der Vorst, *BICGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of non-symmetric linear systems*, SIAM J. Sci. Comput., 13 (1992), pp. 631–644.
[20] R.S. Varga, *Matrix Iterative Analysis*, Prentice–Hall, Englewood Cliffs, NJ, 1962.
[21] Ch. Wagner, *Frequenzfilternde Zerlegungen für unsymmetrische Matrizen und Matrizen mit stark variierenden Koeffizienten*, Ph.D. thesis, ICA University of Stuttgart, Stuttgart Germany, 1995 (in German).
[22] T. Washio and K. Hayami, *Parallel block preconditioning based on SSOR and MILU*, Numer. Linear Alg. Appl., 1 (1994), pp. 533-553.
[23] T. Washio and C.W. Oosterlee, *Experiences with Robust Parallel Multilevel Preconditioners for BiCGSTAB*, GMD Arbeitspapier 949, GMD St. Augustin, Germany, 1995.
[24] P. Wesseling, *An Introduction to Multigrid Methods*, John Wiley, Chichester, 1992.
[25] P.M. de Zeeuw, *Matrix-dependent prolongations and restrictions in a blackbox multigrid solver*, J. Comput. Appl. Math., 33 (1990), pp. 1–27.
[26] S. Zeng, C. Vuik, and P. Wesseling, *Numerical solution of the incompressible Navier–Stokes equations by Krylov subspace and multigrid methods*, Adv. Comput. Math., 4 (1995), pp. 27–49.