ADDITIVE MULTILEVEL-PRECONDITIONERS BASED ON BILINEAR INTERPOLATION, MATRIX-DEPENDENT GEOMETRIC COARSENING AND ALGEBRAIC-MULTIGRID COARSENING FOR SECOND ORDER ELLIPTIC PDES

T. GRAUSCHOPF, M. GRIEBEL, H. REGLER
INSTITUT FÜR INFORMATIK
TECHNISCHE UNIVERSITÄT MÜNCHEN
D-80290 MÜNCHEN, FRG

Abstract. In this paper, we study additive multilevel preconditioners based on bilinear interpolation, matrix-dependent interpolations and the algebraic multigrid approach. We consider 2nd order elliptic problems, i.e. strong elliptic ones, singular perturbation problems and problems with locally strongly varying or discontinuous coefficient functions. We report on the results of our numerical experiments which show that especially the algebraic multigrid based method is mostly robust also in the additive case.

1. Introduction. For the solution of elliptic PDEs, iterative methods based on the multilevel principle are known to be fast solvers. They are often optimal in the sense that the number of iterations necessary to reduce the error by a prescribed value is independent of the mesh size used in the discretization or the number of unknowns, respectively. Here, following [59], multigrid algorithms [12, 30] can be interpreted as multiplicative Schwarz methods, whereas multilevel preconditioners [11] can be identified as their additive counterparts.

For strongly elliptic symmetric PDEs of second order (i.e. basically the Laplacian) optimality of the convergence rate under moderate regularity assumptions was shown in [5, 8, 29, 30] using smoothing and approximation properties for the multigrid method. Without regularity assumptions, optimality was further shown in [4], [11], [34, 35, 36, 37, 63], [18] for the additive case and [9, 10], [63] for the multiplicative case.

However, for these methods in their standard version, the convergence rates are still dependent on the coefficient functions of the operator under consideration, both theoretically (via the Lemma of Sobolev, i.e. by the ellipticity constants) and practically in numerical experiments. Thus, for singularly perturbed problems, for problems with locally strong variation or even a jump in the coefficient functions, the convergence rates become worse with stronger deviation from the ellipticity of the operator and this renders the methods useless for practical applications. They are not robust in the sense of [57].

Now, at least for the multigrid method, i.e. the multiplicative variant, there exist basically two types of modifications of the standard method to remedy this situation.

First, there are a number of approaches where the underlying sequence of grids is still coarsened the usual standard way but the interpolation and restriction matrices and by means of the Galerkin approach also the discrete operators on the coarser grids are chosen dependently of the fine grid matrix. However, a simple point-wise iterative method used as smoother often fails to give robust convergence rates and ILU-type smoothers must be employed instead. Then, at least for 2D problems, a robust multigrid method results, see [56, 57]. The algorithms of Dendy [20], Wesseling [54, 55], de Zeeuw [19], Reusken [43], Wagner [52], Kuznetsov [32], Fuhrmann [23], and also, to some extent, the more preconditioning-oriented algorithms of Axelsson and Vassilevski [2, 3], van den Ploeg [48], Botta et al. [49] belong to this type of method. All these approaches are more or less attempts to approximate the Block Gaussian elimination (Schur complement process) of the unknowns belonging to grid points of the fine grid but not belonging to the next coarser grid (recursively over all levels). For a general framework to that, see [31]. Some of these methods are called algebraic due to the algebraic, i.e. matrix-dependent nature of the construction of interpolation, restriction and coarse grid operators. Nevertheless, they belong to the geometric type of multilevel methods due to the geometric coarsening process of the involved sequence of grids.

Apart from that, there exists the algebraic multigrid method due to Ruge and Stüben [44, 45] where not only the interpolation, restriction and coarse grid operators are chosen in a matrix-dependent manner, but also the sequence of coarsened grids itself. In this way, an algebraic principle is used for both, the construction of coarser grids *and* the construction of the inter-grid transfer operators and the discrete operators on these coarser grids. Thus, a method is gained where, in contrast to the previous mentioned ones, a simple point-wise smoother (e.g. Gauss-Seidel or Jacobi) is sufficient for a robust convergence behaviour of the overall method for a wide class of problems. Beside the fact that theory is missing for this method, maybe also its quite difficult implementation, the difficulties this method gives in the adaptive refinement case and its slightly higher operation count were obstacles to its wider use. The latter is due to the so-called setup step which must be programmed very carefully for maintaining an $O(N)$ operation count, where N is the number of unknowns, see [15]. Nevertheless, in the recent years, there is now more interest in the algebraic multigrid method, since it provides a simple to use black box type solver which can be applied by not so experienced users in many situations where conventional multigrid fails or must be modified in a skilled way. Beside the developments in [16, 17], [61, 62] and [41, 42] there can nowadays also block-wise variants of AMG be found, especially for the treatment of structural analysis, elasticity problems ([6, 51]) and fluid dynamics applications ([7, 33, 40, 53]) where more than one unknown is assigned to one grid point.

In this paper, we use the two previously mentioned modifications of multigrid methods within a multilevel-type preconditioner to gain a hopefully robust additive Schwarz method. We are interested in the question whether and for what problem classes such additive methods behave as robust as their multiplicative (i.e. multigrid) counterparts.

Our motivation to this study was twofold: First, we aim for a method which can be applied to any given stable fine grid discretization (e.g. by a nine point stencil) of an elliptic PDE of second order (with singular perturbations) and results in a robust convergence rate. Thus, in contrast to AMG, we are not interested in a general black box solver for any (M-Matrix) input matrix but study standard discretizations of PDEs instead. And second, for parallelization reasons, we are interested in an additive method. To this end the subproblems to be solved in the additive method should be as small as possible. In a level-like setting this corresponds to point-wise smoothers involving only local operations. Thus, smoothers like ILU are less favorable in this context since they are difficult to parallelize efficiently.

In the first section we repeat the framework of additive Schwarz methods and give it in both, the functional representation by means of subspace splittings and the corresponding algebraic representation. Then, we demonstrate how, by means of a generating system, a corresponding semidefinite linear system can be achieved. We further specialize the additive Schwarz schemes to the case of multilevel preconditioners where the involved subspaces are one-dimensional but form a multilevel basis similar to the one used within the BPX-preconditioner. In contrast to the BPX-method we allow for more general subspaces on coarser levels and corresponding general mappings (interpolations and restrictions) between these subspaces that only have to fulfill a general (local) refinement relation. Then, we state the three general requirements such kind of methods have to fulfill, i.e. an $O(N)$ operation count per iteration cycle, a resulting condition number which is independent of N and the number of levels involved and, most important, a condition number which is independent of the coefficient functions of the operator under consideration. We further give explanations how such a general scheme can be implemented.

In the next section, we consider three specific examples for the construction of the refinement masks and, consequently, of the underlying subspace splitting. They correspond to

- standard coarsening, involving bilinear interpolation and its transposed as restriction, i.e. the BPX-method,
- the coarsening scheme of Dendy as an example of matrix-dependent geometric coarsening and
- the purely algebraic coarsening process of our variant of the algebraic multigrid method.

Finally we give the results of our numerical experiments where we compare the condition numbers and the convergence behaviour of the corresponding additive preconditioning schemes for various types of problem classes of (possibly singularly perturbed) elliptic PDEs of second order with constant and varying coefficient functions.

2. Additive Schwarz preconditioners - functional and algebraic representation. First, consider a stationary iterative method for solving a linear system

$$Au = f, \tag{1}$$

in the Hilbert space V, $A \in \mathcal{L}(V,V)$, $u \in V$, $f \in V$. One iteration step is given by

$$u^{it+1} = u^{it} + B(f - Au^{it}) \ , \tag{2}$$

where B is an approximate inverse of A.

If V is finite-dimensional, we also find appropriate algebraic formulations equivalent to (1) and (2). Let $n := \dim(V)$. Then the corresponding algebraic analogs are

$$\mathcal{A}\mu = \eta \tag{3}$$

with $\mathcal{A} \in \mathbb{R}^{n\times n}, \eta \in \mathbb{R}^n$, the unknowns $\mu \in \mathbb{R}^n$ and

$$\mu^{it+1} = \mu^{it} + \mathcal{B}(\eta - \mathcal{A}\mu^{it}). \tag{4}$$

with $\mathcal{B} \in \mathbb{R}^{n\times n}$. Clearly, for numerical treatment only the algebraic formulations are of interest. $\mathcal{B}$ should have two properties: Multiplication with $\mathcal{B}$ can be implemented efficiently and $\mathcal{B}$ should approximate the inverse of $\mathcal{A}$ somehow. If $\mathcal{B}$ is a good approximation to the inverse of $\mathcal{A}$, one may expect that

$$\kappa(\mathcal{B}\mathcal{A}) << \kappa(\mathcal{A}),$$

where κ denotes the condition number of a matrix. Therefore, $\mathcal{B}$ acts as a preconditioner to $\mathcal{A}$. Indeed, many popular preconditioning techniques correspond to a linear iteration. For example, if we split the matrix $\mathcal{A}$ into its lower triangular, diagonal and upper tridiagonal parts, $\mathcal{A} = \mathcal{X} + \mathcal{Y} + \mathcal{Z}$, preconditioning with $\mathcal{B} := \mathcal{Y}^{-1}$ corresponds to the Jacobi iterative method. Preconditioning with $\mathcal{B} := (\mathcal{Z}+\mathcal{Y})^{-1}\mathcal{Y}(\mathcal{X}+\mathcal{Y})^{-1}$ corresponds to the symmetric Gauß-Seidel method. In this way, a better but not yet optimal condition number is usually achieved. More efficient preconditioning schemes are obtained via multilevel subspace splittings, which we will consider in the next section. We first give the functional representation according to (2) and then derive the algebraic representation according to (4) from it. Note that there exist condition number estimates for the resulting functional representation [34, 35, 36, 37], [58, 59], [4, 60], which translate directly to the algebraic representation.

2.1. Functional representation. We use the following Hilbert space setting. Let V be some fixed, finite-dimensional Hilbert space. The scalar product in V is denoted by $(\cdot,\cdot)$. We consider a positive definite, symmetric bilinear form $a(u,v) = (Au,v)$, $u,v \in V$, with $A : V \to V$ denoting the corresponding s.p.d. operator acting on V.

Now, consider an arbitrary additive representation of V by the sum of a finite number of subspaces $V_j \subset V$:

$$V = \sum_{k=1}^{s} V_k\ . \tag{5}$$

More precisely, this means that any $u \in V$ has at least one representation $u = \sum_{k=1}^{s} u_k$ where $u_k \in V_k$ for $k = 1,\ldots,s$. The operator A possesses restrictions A_k to V_k given by

$$(A_k u_k, v_k) = (Au_k, v_k)\ , \qquad u_k, v_k \in V_k. \tag{6}$$

Suppose further that the V_k are equipped with auxiliary continuous s.p.d. forms

$$r_k(u_k, v_k) = (R_k u_k, v_k)$$

given by the s.p.d. operators $R_k : V_k \to V_k$. These forms might model approximative solvers used on the subspaces, i.e. R_k is an approximative inverse of A_k.

Now, we define two projection operators $E_k, Q_k : V \longrightarrow V_k$ by

$$a(u, v_k) = a(E_k u, v_k) \qquad \text{(energy projection)}$$

$$(u, v_k) = (Q_k u, v_k) \qquad (L_2\text{-Projection})$$

where $v_k \in V_k, u \in V$. Obviously one then has

$$A_k E_k = Q_k A. \tag{7}$$

If we multiply $Au = f$ by Q_k we obtain with (7) the corresponding subspace problem

$$A_k u_k = f_k \ , \tag{8}$$

where $u_k := E_k u$, $f_k := Q_k f$. With the previous defined operators R_k we get approximate solutions of (8):

$$\tilde{u}_k = R_k f_k.$$

Following Xu [58, 59], we can use this framework to define an iterative scheme for solving $Au = f$. Assume some approximation u^{it} to the solution u. If we could exactly solve the residual equation $Ae = r^{it} := f - Au^{it}$, we would have the solution, because $u = u^{it} + e$. But this is usually as complicated as the original problem. We can instead solve the restricted residual equation

$$A_k e_k^{it} = Q_k r^{it}$$

approximately by

$$\tilde{e}_k^{it} = R_k Q_k r^{it}$$

and, for $k = 1, \ldots, s$, add this subspace corrections to our iterate u^{it}. Then we have

$$u^{it+1} = u^{it} + Br^{it} \tag{9}$$

with the s.p.d. operator

$$B := \sum_{k=1}^{s} R_k Q_k. \tag{10}$$

This is an additive Schwarz iteration, because the update of the values in (9) takes place additively for all subspaces V_k , $k = 1, \ldots, s$. Now, B can also be used as preconditioner.

In the next section we derive the algebraic representation of B.

2.2. Algebraic representation. Assume a basis $\{\phi_{k1}, \ldots, \phi_{kn_k}\}$ is given on a subspace V_k. Then, every $v_k \in V_k$ may be uniquely represented as

$$v_k = \sum_{i=1}^{n_k} \chi_{ki}\phi_{ki} \ , \qquad n_k := \dim(V_k),$$

and the vector $\chi_k := (\chi_{k1}, \ldots, \chi_{k,n_k})^T$ will be referred to as coordinate vector of v_k.

The associated mapping $\xi_k : \mathbb{R}^{n_k} \longrightarrow V_k$ will be defined by

$$\xi_k \chi_k := \sum_{i=1}^{n_k} \chi_{ki} \phi_{ki}. \tag{11}$$

Obviously

$$\xi_k^T v_k = \left((v_k, \phi_{k1}), \ldots, (v_k, \phi_{kn_k})\right)^T .$$

Now, we assume $A_k \in \mathcal{L}(V_k, V_k)$. The *stiffness matrix* $\mathcal{A}_k \in \mathbb{R}^{n_k \times n_k}$, as used in the finite element method, is defined by

$$(\mathcal{A}_k)_{ij} := (A_k \phi_{ki}, \phi_{kj}).$$

One easily verifies

$$\mathcal{A}_k = \xi_k^T A_k \xi_k. \tag{12}$$

The stiffness matrix $\mathcal{A}_k$ is just the algebraic representation of A_k. We also need the *mass matrix* $\mathcal{M}_k \in \mathbb{R}^{n_k \times n_k}$. It is defined by

$$(\mathcal{M}_k)_{ij} := (\phi_{ki}, \phi_{kj}),$$

which is equivalent to

$$\mathcal{M}_k = \xi_k^T \mathrm{id}_k \xi_k = \xi_k^T \xi_k \ , \tag{13}$$

where id_k denotes the identity operator on V_k.

In the following we reconsider the algebraic representation (4) of the iterative method given by (2) in a subspace V_k. We first start with the functional formulation of a single step of an iterative method on the subspace V_k:

$$u_k^{it+1} = u_k^{it} + R_k(f_k - A_k u_k^{it}).$$

With

$$u_k^{it+1} = \xi_k \mu_k^{it+1}, \ u_k^{it} = \xi_k \mu_k^{it}, \ f_k = \xi_k \eta_k,$$

we obtain

$$\xi_k \mu_k^{it+1} = \xi_k \mu_k^{it} + R_k(\xi_k \eta_k - A_k \xi_k \mu_k^{it}). \tag{14}$$

From this we conclude that an appropriate algebraic representation $\mathcal{R}_k$ of R_k is given implicitly by

$$\xi_k \mathcal{R}_k \xi_k^T = R_k. \tag{15}$$

Using (12), (13) and (15), relation (14) can be transformed to

$$\xi_k \mu_k^{it+1} = \xi_k \mu_k^{it} + \xi_k \mathcal{R}_k (\mathcal{M}_k \eta_k - \mathcal{A}_k \mu_k^{it}). \tag{16}$$

Since ξ_k is invertible on V_k, (16) is equivalent to

$$\mu_k^{it+1} = \mu_k^{it} + \mathcal{R}_k(\mathcal{M}_k\eta_k - \mathcal{A}_k\mu_k^{it}).$$

Now, given a basis $\{\phi_1, \ldots, \phi_n\}$ of V and $A \in \mathcal{L}(V,V)$, we can define the mappings ξ and ξ^T and the matrices $\mathcal{A}$ and $\mathcal{M}$ for V analogously as we did for the subspace V_k. Then, an appropriate algebraic representation $\mathcal{B}$ of B is given implicitly by

$$\xi\mathcal{B}\xi^T = B. \tag{17}$$

Note that, in contrast to ξ and ξ^T, ξ_k and ξ_k^T are not invertible on V (but they are invertible on V_k).

Furthermore, since $V_k \subset V$, every basis function ϕ_{ki} of the subspace V_k possesses an unique representation in terms of the basis functions ϕ_l of V :

$$\phi_{ki} = \sum_{l=1}^{n} \pi_l^{ki}\phi_l \ , \qquad \pi_l^{ki} \in \mathbb{R}. \tag{18}$$

For the following it is practical to exploit this fact to define the matrix $\mathcal{P}_k \in \mathbb{R}^{n\times n_k}$ by

$$(\mathcal{P}_k)_{ij} = \pi_i^{kj}.$$

Now, using $\mathcal{P}_k$, we can give an explicit representation of ξ_k, i.e.

$$\xi_k = \xi\mathcal{P}_k. \tag{19}$$

Furthermore, the L_2-Projector Q_k onto V_k can be stated as follows:

$$Q_k = \xi_k\mathcal{M}_k^{-1}\xi_k^T. \tag{20}$$

At last, we can deduce the algebraic representation $\mathcal{B} \in \mathbb{R}^{n\times n}$ of the additive Schwarz operator $B = \sum_{k=1}^{s} R_kQ_k$ in the following way:

From (17) we get

$$\mathcal{B} = \xi^{-1}B\xi^{-T} = \sum_{k=1}^{s} \xi^{-1}R_kQ_k\xi^{-T}. \tag{21}$$

Plugging the identities (15) and (20) into (21) and then using (13) and (19), we obtain

$$\begin{aligned} \mathcal{B} &= \sum_{k=1}^{s} \xi^{-1}R_kQ_k\xi^{-T} = \\ &= \sum_{k=1}^{s} \xi^{-1}\xi_k\mathcal{R}_k\xi_k^T\xi_k\mathcal{M}_k^{-1}\xi_k^T\xi^{-T} = \\ &= \sum_{k=1}^{s} \mathcal{P}_k\mathcal{R}_k\mathcal{P}_k^T. \end{aligned} \tag{22}$$

This is the algebraic representation of the additive Schwarz preconditioner we looked for.

Now, it is easy to proof, that

$$\kappa(BA) = \kappa(\mathcal{B}\mathcal{A}),$$

so that if we can show condition number estimates for the functional representation, they immediately translate to the algebraic representation, and vice versa.

The often-needed Galerkin identity may be shown as follows:

$$\begin{aligned}
\mathcal{A}_k &= \xi_k^T \text{ id } A \text{ id } \xi_k = \xi_k^T(\xi\mathcal{M}^{-1}\xi^T)A(\xi\mathcal{M}^{-1}\xi^T)\xi_k = \\
&= (\xi_k^T\xi\mathcal{M}^{-1})(\xi^T A\ \xi)(\mathcal{M}^{-1}\xi^T\xi_k)
\end{aligned}$$

and from this we obtain with (19)

$$\mathcal{A}_k = \mathcal{P}_k^T\mathcal{A}\mathcal{P}_k. \tag{23}$$

Thus, if either a discretization or a subspace operator should be consistent with this theory, it must be constructed via the Galerkin identity.

2.3. From a definite to a semidefinite system. In the previous section, we gave the algebraic representation of the additive Schwarz operator $B = \sum_{k=1}^s R_k Q_k$ in terms of the basis $\{\phi_1, \ldots, \phi_n\}$ of V, which was

$$\mathcal{B} = \sum_{k=1}^{s} \mathcal{P}_k\mathcal{R}_k\mathcal{P}_k^T. \tag{24}$$

Now it is an interesting task to deduce an algebraic representation of it with respect not to a basis of V but to the generating system

$$\bigcup_{k=1}^{s}\{\phi_{k1}, \ldots, \phi_{kn_k}\}$$

of V which contains *all* the bases of the subspaces arising in the splitting (5). Then, elements of V will in general have a non-unique representation only. Let us denote

$$n^E := \sum_{k=1}^{s} n_k.$$

Analogously to the previous section, we can define the operator $\xi^E : \mathbb{R}^{n^E} \longrightarrow V$ by

$$\xi^E\chi^E = \sum_{k=1}^{s}\sum_{l=1}^{n_k}\chi_{kl}^E\phi_{kl}\ ,$$

where

$$\chi^E = (\chi_{11}, \ldots, \chi_{1n_1}, \ldots, \chi_{s1}, \ldots, \chi_{sn_s})^T \in \mathbb{R}^{n^E}.$$

Furthermore, we define the matrix $\mathcal{P}^E \in \mathbb{R}^{n\times n^E}$ by

$$\mathcal{P}^E = (\mathcal{P}_1, \ldots, \mathcal{P}_s).$$

It is elementary to show, that

$$\xi^E = \xi \mathcal{P}^E \ . \tag{25}$$

One step of a stationary iterative method using the generating system is now given by

$$\mu^{E,it+1} = \mu^{E,it} + \mathcal{B}^E(\mathcal{M}^E \eta^{E,it} - \mathcal{A}^E \mu^{E,it}) \ , \tag{26}$$

where

$$\mathcal{A}^E = (\xi^E)^T A \ \xi^E = (\mathcal{P}^E)^T \mathcal{A} \ \mathcal{P}^E,$$

and

$$\mathcal{M}^E = (\xi^E)^T \xi^E = (\mathcal{P}^E)^T \mathcal{M} \ \mathcal{P}^E.$$

Note that the entries of $\mathcal{A}^E$ and $\mathcal{M}^E$ are now given by the values $a(\phi_{k_1,i_1}, \phi_{k_2,i_2})$ and $(\phi_{k_1,i_1}, \phi_{k_2,i_2})$, $k_1, k_2 = 1, \ldots, s$, $i_1 = 1, \ldots, n_{k_1}, i_2 = 1, \ldots, n_{k_2}$, respectively. Then, we obtain the solution u of (1) from any fix-point μ^E of (26) by

$$u = \xi^E \mu^E.$$

More generally speaking, the solutions u, μ, μ^E of $Au = f$, $\mathcal{A}\mu = \xi^T f$, $\mathcal{A}^E \mu^E = (\xi^E)^T f$ are connected via

$$u = \xi \mu = \xi^E \mu^E.$$

$\mathcal{B}^E$ in (26) is again implicitly defined by

$$B = \xi^E \mathcal{B}^E (\xi^E)^T.$$

If we plug in the multilevel preconditioner (24) for $\mathcal{B}$ we obtain with (25) and (17):

$$\mathcal{P}^E \mathcal{B}^E \mathcal{P}^E = \xi^{-1} B \xi^{-T} = \mathcal{B} = \sum_{k=1}^{s} \mathcal{P}_k \mathcal{R}_k \mathcal{P}_k^T .$$

This equation is fulfilled, if we take as algebraic representation for $\mathcal{B}^E$ for example

$$\mathcal{B}^E = \mathrm{diag}\,(\mathcal{R}_1, \ldots, \mathcal{R}_s) \, .$$

So if we look at the additive Schwarz operator from the view of our generating system instead of the basis of V, its algebraic representation simplifies to a block diagonal matrix.

Note also, that if we define a generalized condition number $\tilde{\kappa}$ as the quotient of the largest eigenvalue and the smallest non-vanishing eigenvalue, we have

$$\kappa(BA) = \kappa(\mathcal{B}\mathcal{A}) = \tilde{\kappa}(\mathcal{B}^E \mathcal{A}^E) \ .$$

Furthermore, modern multilevel and domain decomposition methods correspond to classical iterative methods applied to the enlarged representation of the generating system: Additive Schwarz methods are then equivalent to (block-)Jacobi, multiplicative Schwarz methods to (block-)Gauß-Seidel iterations.

2.4. Multilevel preconditioners: nested subspace splittings. In the following we concentrate our studies on preconditioners constructed from a multilevel splitting of V which we will now introduce in terms given before.

We therefore assume a splitting of V into *nested* subspaces $V_L, L = 1, \ldots, M$,

$$V_1 \subset V_2 \subset \ldots \subset V_{M-1} \subset V_M = V. \tag{27}$$

It should be emphasized that the nestedness of these subspaces is the only prerequisite we need. It implies an ordering to the subspaces, where the associated indices now denote the so-called levels. In finite element methods usually compactly supported basis functions are chosen where each of them is associated to a specific point of the discretization domain, such that the other basis functions vanish there. Then, these points set up an associated grid[1]. As a consequence of the nestedness (27), if $C < F$, then the grid belonging to a coarser space V_C is a subset of the grid belonging to V_F.

We could start with such subspaces in decomposition (5). But it comes handy to refine the V_L further into spans of single basis functions, i.e. one-dimensional subspaces:

$$V = \sum_{L=1}^{M} V_L = \sum_{L=1}^{M} \sum_{i=1}^{n_L} V_{Li} \ , \tag{28}$$

where $n_L := \dim(V_L)$ and $V_{Li} := \operatorname{span}\{\phi_{Li}\}$ are *one-dimensional* subspaces. In the notation above, a subspace V_{Li} corresponds to a subspace V_k in (5) (one might imagine an appropriate reordering of the indices). To avoid notational conflicts, we use uppercase indices in connection with levels, and lowercase indices for the single unknowns or basis functions.

Let us now look at the concrete formulations of the operators defined in section 2.1. By (6) we see that

$$A_{Li} u_{Li} = \frac{a(u_{Li}, \phi_{Li})}{(\phi_{Li}, \phi_{Li})} \phi_{Li} \ , \qquad u_{Li} \in V_{Li},$$

which holds because any $v_{Li} \in V_{Li}$ can be represented as $\lambda \phi_{Li}$, since $V_{Li} = \operatorname{span}\{\phi_{Li}\}$. Thus, we can set

$$R_{Li} u_{Li} = \frac{(u_{Li}, \phi_{Li})}{a(\phi_{Li}, \phi_{Li})} \phi_{Li}$$

and obtain an exact subspace solver. It can also easily be seen, that for $u_M \in V_M = V$

$$Q_{Li} u_M = \frac{(u_M, \phi_{Li})}{(\phi_{Li}, \phi_{Li})} \phi_{Li}$$

[1] Note that, at this point, we do not assume any uniformity of the grid nor that the basis functions associated to a grid point must be linear at all, especially on coarser levels. They can and will, as we will see in section 5.5, consist of functions which are piecewise linear only with respect to much finer grids.

and

$$E_{Li} u_M = \frac{a(u_M, \phi_{Li})}{a(\phi_{Li}, \phi_{Li})} \phi_{Li}$$

hold. Because of our choice $R_{Li} = A_{Li}^{-1}$ we now obtain the formulas

$$\begin{aligned} B_M v_M &= \sum_{L=1}^{M} \sum_{i=1}^{n_L} R_{Li} Q_{Li} v_M = \sum_{L=1}^{M} \sum_{i=1}^{n_L} \frac{(v_M, \phi_{Li})}{a(\phi_{Li}, \phi_{Li})} \phi_{li} \ , \\ B_M A_M v_M &= \sum_{L=1}^{M} \sum_{i=1}^{n_L} E_{Li} v_M = \sum_{L=1}^{M} \sum_{i=1}^{n_L} \frac{a(v_M, \phi_{Li})}{a(\phi_{Li}, \phi_{Li})} \phi_{Li} \ , \end{aligned}$$

which are the applications of diagonal scaled versions of the multilevel-preconditioner and the multilevel-preconditioned matrix to a vector.

We can also consider the algebraic representation. We have

$$\mathcal{B}_M = \sum_{L=1}^{M} \sum_{i=1}^{n_L} \mathcal{P}_{Li} \mathcal{R}_{Li} \mathcal{P}_{Li}^T.$$

Because of (28) it is clear, that $n_C \le n_F$ for $C < F$, and therefore every basis function ϕ_{Ci} of V_C may be represented as a linear combination of basis functions ϕ_{Fj} of V_F analogously to (18):

$$\phi_{Ci} = \sum_{j=1}^{n_F} \pi_{Fj}^{Ci} \phi_{Fj} \ , \qquad \pi_{Fj}^{Ci} \in \mathbb{R}. \tag{29}$$

If we collect the matrices $\mathcal{P}_{Li}$ corresponding to one level L in a single matrix by

$$\mathcal{P}_L^M := (\mathcal{P}_{L1}, \ldots, \mathcal{P}_{Ln_L})$$

and define the diagonal matrix

$$\mathcal{R}_L := \operatorname{diag}(\mathcal{R}_{L1}, \ldots, \mathcal{R}_{Ln_L}) \,,$$

we obtain

$$\mathcal{B}_M = \sum_{L=1}^{M} \mathcal{P}_L^M \mathcal{R}_L (\mathcal{P}_L^M)^T. \tag{30}$$

$\mathcal{R}_L$ is the inverse of the diagonal of the stiffness matrix $\mathcal{A}_L$, i.e. $\mathcal{R}_{Li} = a(\phi_{Li}, \phi_{Li})^{-1}$. Note also, that $\mathcal{A}_L$ is determined by $\mathcal{A}_M$, i.e. the stiffness matrix of the finest level of discretization, and $\mathcal{P}_L^M$. This is due to the Galerkin identity (23), which implies

$$\mathcal{A}_L = (\mathcal{P}_L^M)^T \mathcal{A}_M \mathcal{P}_L^M \ . \tag{31}$$

$\mathcal{P}_L^M$ can be defined by interpolation matrices between two successive levels. In our context, $\mathcal{P}_L^{L+1} \in \mathbb{R}^{n_{L+1} \times n_L}$ should be the interpolation matrix defined by the $\pi_{L+1,j}^{Li}$, compare also (29). Then we can show that

$$\mathcal{P}_L^M = \mathcal{P}_{M-1}^M \cdot \ldots \cdot \mathcal{P}_L^{L+1}. \tag{32}$$

These facts are essential for an efficient implementation of the corresponding preconditioner.

With this notation the generating system version of our preconditioner is simply the diagonal matrix

$$\mathcal{B}_M^E = \mathrm{diag}(\mathcal{R}_1, \ldots, \mathcal{R}_M) \ ,$$

which has to be applied to the enlarged matrix

$$\mathcal{A}_M^E = (\mathcal{P}_M^E)^T \mathcal{A}\, \mathcal{P}_M^E \ ,$$

where $\mathcal{P}_M^E$ can compactly be defined as

$$\mathcal{P}_M^E := (\mathcal{P}_1^M, \ldots, \mathcal{P}_{M-1}^M, \mathcal{P}_M^M) \tag{33}$$

($\mathcal{P}_M^M$ is the unity matrix). This gives now the alternative representation

$$\mathcal{B}_M = \mathcal{P}_M^E \mathcal{B}_M^E (\mathcal{P}_M^E)^T$$

of the preconditioner $\mathcal{B}_M$.

Note that the enlarged matrices $\mathcal{A}_M^E$ and $\mathcal{B}_M^E \mathcal{A}_M^E$ are in general semidefinite and possess zero eigenvalues as well. But the spectrum of $\mathcal{B}_M^E \mathcal{A}_M^E$ is up to the zero eigenvalues the same as that of $\mathcal{B}_M \mathcal{A}_M$. Furthermore, by using the generating system representation we are now able to write the preconditioned matrix by

$$(\mathcal{B}_M^E)^{1/2} (\mathcal{P}_M^E)^T \mathcal{A}_M\, \mathcal{P}_M^E (\mathcal{B}_M^E)^{1/2} \tag{34}$$

in a symmetric form whenever $\mathcal{A}_M$ is symmetric. This is in contrast to the definite but unsymmetric notation $\mathcal{B}_M \mathcal{A}_M$.

The approach with the generating system now allows to apply standard iterative methods to the enlarged problem, see [25, 28]. For our choice of $\mathcal{B}_M^E$ we just obtain the Jacobi iteration. If we would apply the symmetric Gauß-Seidel iteration to the enlarged linear system associated to (34) we obtain a method exactly equivalent to the multigrid V-cycle with one pre- and one post-smoothing step using the nested spaces $V_1, \ldots, V_M$. Beside these approaches other traditional iterative methods can be applied to (34) resulting in new multilevel algorithms. For example in [26, 27], point- instead of level-oriented methods are derived from just permuting the generating system properly.

3. Model problem and requirements for an efficient implementation of the multilevel-preconditioner. In this paper, we restrict A to be the two-dimensional elliptic differential operator

$$\left(\sum_{i,j=1}^{2} a_{ij}(x_1, x_2) \frac{\partial^2}{\partial x_i \partial x_j} + \sum_{i=1}^{2} b_i(x_1, x_2) \frac{\partial}{\partial x_i} + c(x_1, x_2) \right) \tag{35}$$

on the domain $[0, 1] \times [0, 1]$ with Dirichlet boundary conditions.

As starting point for the initial fine-level discretization of A serves the regular, equidistant grid

$$\Omega_h := \{(ih, jh) : i, j = 1, \ldots, h^{-1} - 1\} \ . \tag{36}$$

The space V is then defined as the span of functions $\phi_i \ , i = 1, \ldots, (h^{-1} - 1)^2$ of principally any type with local support. This gives us a finite-dimensional restriction A_h of A and via Galerkin discretization its algebraic equivalent $\mathcal{A}_h$.

For example, as basis for V, the well-known nodal basis can be used, which spans the space of bilinear functions over Ω_h. A nodal basis function centered at (a, b) is defined by

$$\phi^h_{(a,b)}(x_1, x_2) := \max\{0, (h - |x_1 - a|)/h\} \cdot \max\{0, (h - |x_2 - b|)/h\} \ .$$

The entries in the initial discretization matrix $\mathcal{A}_h$ are then given by the scalar products $a(\phi^h_{(ih,jh)}, \phi^h_{(kh,lh)})$, where $a(\cdot, \cdot)$ is the weak bilinear form induced by (35)[2].

With $\mathcal{A}_h$ given, we now have the freedom to choose the interpolation matrices $\mathcal{P}^M_L, L = M - 1, \ldots, 1$, i.e. the matrix $\mathcal{P}^E_M$ and depending on that the number of levels M, defining $\mathcal{A}_M := \mathcal{A}_h$. Note that, by choosing a specific interpolation scheme, not only the stiffness matrices on the coarser levels are determined by the Galerkin approach but also, at least implicitly, a sequence of bases $\phi_{Li} \ , L = 1, \ldots, M \ , i = 1, \ldots, n_L$ and associated spaces $V_L \ , L = 1, \ldots, M$ is defined. Except for the special case of the values $\pi^{L-1,j}_{Lj}$ given in subsection 4.1 where self-similarity of the basis functions of different levels under translation and dilation is achieved, the basis functions on the coarser levels are usually not more bilinear functions w.r.t. to that level of discretization but merely local linear combinations of the basis functions of the finest level, and thus are piecewise linear with respect to the *finest* grid Ω_h only.

Optimally, the choice of the interpolation matrices should be such that the resulting preconditioned fine grid system or equivalently the resulting enlarged generation system matrix fulfills three requirements:

1. *$h-$independent convergence rate*: The condition number $\kappa(\mathcal{B}_M \mathcal{A}_M)$ of the resulting preconditioned system should be bounded by a constant independent of the mesh size h.
2. *Robustness of the method:* The condition number $\kappa(\mathcal{B}_M \mathcal{A}_M)$ of the resulting preconditioned system should be bounded by a constant independent of the values of the coefficient functions $a_{ij}(x_1, x_2), b_i(x_1, x_2)$ and $c(x_1, x_2)$ in (35).
3. *Work count*: The number of operations necessary to perform the matrix vector multiplication by the resulting preconditioner should be proportional to the number of fine grid unknowns only.

If merely requirement 1 is fulfilled, the constants forming the condition number can still depend on the coefficient functions $a_{ij}(x_1, x_2), b_i(x_1, x_2)$ and $c(x_1, x_2)$. Now,

[2] The grid (36) also serves as starting point for the finite difference discretizations in our numerical experiments presented in a later section. Often finite difference discretizations can be viewed as result of a finite element discretization with different basis functions, i.e. linear instead of bilinear ones on a triangulation of Ω.

if one ore more of these functions possess large values in Ω, the condition number can be large as well and the corresponding solution procedure is practically useless, since its convergence is unacceptable slow. This is the case for singular perturbed problems, when one or more of the coefficient functions are dominating, and causes problems also in the case of rapidly varying or discontinuous coefficients as they appear in so-called interface problems. Thus, it is advantageous to choose the interpolations such that $\kappa(\mathcal{B}_M\mathcal{A}_M)$ is independent of the coefficient functions.

One step of the resulting iterative method should only involve a number of operations proportional to the unknowns on the finest level of discretization. Usually, f.e. in a CG-type iterative method, the multiplication of a vector with the preconditioned fine grid stiffness matrix is the crucial task. Beside the multiplication with the matrix $\mathcal{A}_M$, it also invokes the multiplication with $\mathcal{B}_M$. From (30) and (32) it is clear how a multiplication with $\mathcal{B}_M$ can be implemented.

Multiplication of a vector $\boldsymbol{v_M}\in\boldsymbol{V_M}$with $\boldsymbol{\mathcal{B}_M}$

$w_M = v_M$
for $L = M, \ldots, 2$:
    $w_{L-1} = (\mathcal{P}^L_{L-1})^T w_L$
for $L = 1, \ldots, M$:
    $w_L = \mathcal{R}_L w_L$
for $l = 2, \ldots, M$:
    $w_L = w_L + \mathcal{P}^L_{L-1} w_{L-1}$
$\mathcal{B}_M v_K = w_K$

We want the number of operations for a single matrix-vector multiplication[3] with $\mathcal{B}_M$ to have the same order as the number of unknowns, i.e. $\#operations \approx n_M$. Therefore, the interpolations $\mathcal{P}^L_{L-1}$ must be sparse matrices. Thus, only a bounded number of entries in each row and column may be non-zero. This means that, in the representation

$$\phi_{L-1,i} = \sum_{j=1}^{n_L} \pi^{L-1,i}_{Lj} \phi_{Lj} \ , \tag{37}$$

only a fixed number of $\pi^{L-1,i}_{Lj}$ may be non-zero, which is equivalent to locally supported basis functions on all levels.

Furthermore, the number of basis functions over all levels $\sum_{L=1}^M n_L$ must be in the order of n_M as well. In a practical approach this is fulfilled, if the number of unknowns decreases geometrically between the levels, $n_L \le \rho n_{L+1}$, $0 < \rho < 1$. This automatically leads to the relation $M \approx \log(h^{-1})$.

[3] Note that the multiplication with the generating system matrix can be implemented in just the same amount of work, see [26].

4. Considered interpolation schemes. In the previous section we already stated that the multilevel-preconditioning scheme is uniquely determined by the stiffness matrix $\mathcal{A}_M$ and the interpolation matrices $\mathcal{P}_{L-1}^{L}$, $l = 2, \ldots, M$. Because of the Galerkin identity (31), the inverse $\mathcal{R}_L$ of the diagonal of $\mathcal{A}_L$ is then determined, too. So if we have $\mathcal{A}_M$, our only freedom is to choose the interpolation weights $\pi_{L,j}^{L-1,i}$, both, with respect to locality, i.e. the non-zero pattern, and with respect to the specific values of the non-zero entries. In this section we consider three different interpolation schemes, i.e. bilinear interpolation which results in the standard BPX-preconditioning [11] or in its diagonal scaled variant MDS [63], respectively, matrix-dependent weighted interpolation with nine-point masks which results in a preconditioner that corresponds to the approaches of Dendy and deZeeuw [20], [19], and a more general interpolation which results in a preconditioner that corresponds to the AMG method of Ruge and Stüben [44, 45].

4.1. Standard coarsening with bilinear interpolation. Bilinear interpolation and its scaled transposed, the weighted restriction are the standard inter-grid transfer operators for multigrid methods. For many problems, especially strong-elliptic ones they work well.

We will use bilinear interpolation only in connection with regular coarsened grids $\Omega_h, \Omega_{2h}, \ldots, \Omega_{1/2}$, where h^{-1} should be the M^{th} power of 2. Then, the underlying basis for *each* subspace $V_L,\ L = M, \ldots, 1$ should be the nodal basis defined by the grid $\Omega_{2^{-L}}$. The nodal-basis functions are then self-similar and satisfy the recurrence relation

$$\phi_{(x_0,y_0)}^{2h} \equiv \sum_{i,j=-1}^{1} \pi_{i,j} \phi_{(x_0+ih,y_0+jh)}^{h} ,$$

where $\pi_{-1,-1} = \pi_{-1,1} = \pi_{1,-1} = \pi_{1,1} = 1/4$, $\pi_{0,-1} = \pi_{-1,0} = \pi_{0,1} = \pi_{1,0} = 1/2$, $\pi_{0,0} = 1$. This weights are, after reordering of the indices exactly the non-zero values $\pi_{L,j}^{L-1,i}$ in (37). They can be stated shortly by the well-known stencil notation

$$\frac{1}{4}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} .$$

Using bilinear interpolation in a multilevel preconditioning algorithm saves the setup phase for the computation of the interpolation weights. A further advantage is, that if interpolation and discretization can be expressed with nine-point stencils, the Galerkin coarsening leads to discretizations on coarser levels which can be expressed by nine-point stencils again. This means, that the sparsity pattern of the matrices is preserved in the coarsening process.

The major disadvantage of bilinear interpolation is its lack of flexibility which can result in bad convergence rates, as we will see in the numerical results for singularly perturbed problems.

For this kind of interpolation the associated multilevel preconditioner (30) results in the so-called MDS-method, see [63], which is just the variant of the BPX-preconditioner [11, 58] involving multilevel diagonal scaling.

4.2. Standard coarsening with matrix-dependent interpolation. In contrast to the use of a-priori fixed interpolation and restriction weights in $\mathcal{P}_{L-1}^{L}, L = M, \dots, 2$, as in the case of bilinear interpolation, one can take advantage of the information contained in the matrix $\mathcal{A}_L, L = M, \dots, 2$, respectively. Coarsening schemes based on this idea are called matrix-dependent, see [1, 19, 20]. In algorithmic notation:

Matrix-dependent coarsening scheme
Given initially $\mathcal{A}_M$.
for $L = M, \dots, 2$:
    Compute $\mathcal{P}_{L-1}^{L}$ from $\mathcal{A}_L$.
    Compute $\mathcal{A}_{L-1} = (\mathcal{P}_{L-1}^{L})^T \mathcal{A}_L \mathcal{P}_{L-1}^{L}$.

This approach requests an initial setup phase, where the interpolations and the Galerkin coarse grid operators are computed successively. To be competitive, the number of operations for the setup phase must be bounded by a constant times the number of unknowns.

In this subsection, we consider approaches with a standard coarsened grid sequence as in subsection 4.1, but now the interpolation and restriction matrices and, by means of the Galerkin approach, also the discrete operators are chosen dependently of the fine grid matrix. In connection with this interpolations ILU-type smoothers can lead to multigrid methods with robust convergence rates, see Wittum [56, 57], Wesseling [54, 55], de Zeeuw [19], Reusken [43], Wagner [52], Fuhrmann [23], and the more preconditioning-oriented algorithms of Axelsson and Vassilevski [2, 3], and van den Ploeg et al. [48, 49]. As stated already in the introduction all these approaches are more or less attempts to approximate the Block Gaussian elimination (Schur complement process) of the unknowns belonging to grid points of the fine grid but not belonging to the next coarser grid (recursively over all levels). For a general framework to that, see [31]. Some of these methods are called algebraic due to the algebraic, i.e. matrix-dependent nature of the construction of interpolation, restriction and coarse grid operators. Nevertheless, they belong to the geometric type of multilevel methods due to the geometric coarsening process of the involved sequence of grids.

In [20], Dendy proposes a coarsening scheme for nine-point discretizations. There, as in the case of bilinear interpolation, regular coarsened grids $\Omega_h, \Omega_{2h}, \dots, \Omega_{1/2}$, $h = 2^{-M}$ are used again.

Let the discretization matrix $\mathcal{A}_M$ be given in stencil notation by

$$\begin{bmatrix} \alpha_{-1,-1}^{i,j} & \alpha_{-1,0}^{i,j} & \alpha_{-1,1}^{i,j} \\ \alpha_{0,-1}^{i,j} & \alpha_{0,0}^{i,j} & \alpha_{0,1}^{i,j} \\ \alpha_{1,-1}^{i,j} & \alpha_{1,0}^{i,j} & \alpha_{1,1}^{i,j} \end{bmatrix}, \qquad i,j = 1, \dots, 2^L - 1. \tag{38}$$

The upper indices i, j should suggest that the operator may be domain-dependent and the stencils are therefore varying from grid point to grid point. The interpolation is expressed by nine-point stencils, too:

$$\left[\begin{array}{ccc} \pi^{i,j}_{-1,-1} & \pi^{i,j}_{-1,0} & \pi^{i,j}_{-1,1} \\ \pi^{i,j}_{0,-1} & \pi^{i,j}_{0,0} & \pi^{i,j}_{0,1} \\ \pi^{i,j}_{1,-1} & \pi^{i,j}_{1,0} & \pi^{i,j}_{1,1} \end{array}\right] , \qquad i,j = 1,\ldots,2^{L-1}-1. \tag{39}$$

After reordering these are our weights $\pi^{L-1,i}_{L,j}$ of (37) again. Here,

$$\pi^{i,j}_{0,0} = 1 \ , \qquad i,j = 1,\ldots,2^{L-1}-1,$$

which means values in the coarse grid points are identically transfered to the corresponding fine grid points, as in the bilinear interpolation. For the weights $\pi^{i,j}_{\pm 1,0}, \pi^{i,j}_{0,\pm 1}$ we build two one-dimensional stencils by summing up the weights of the stencil (38) in the coordinate directions. Then we request a grid function $\mathcal{P}^L_{L-1}u_{L-1}, u_{L-1} \in \mathbb{R}^{n_{L-1}}$ to solve the local homogeneous one-dimensional problems defined by this stencils exactly in the points (ih_L, jh_L), where $i+j$ is odd. Thus, we obtain the weights

$$\pi^{I,J}_{\pm 1,0} = -\frac{\sum_{s=-1}^{1} \alpha^{i\pm 1,j}_{\mp 1,s}}{\sum_{s=-1}^{1} \alpha^{i\pm 1,j}_{0,s}} \ ,$$

and

$$\pi^{I,J}_{0,\pm 1} = -\frac{\sum_{s=-1}^{1} \alpha^{i,j\pm 1}_{s,\mp 1}}{\sum_{s=-1}^{1} \alpha^{i,j\pm 1}_{s,0}} \ .$$

Here, for convenience of notation, let $(i,j) = (2I, 2J)$. The diagonal weights are derived from the demand

$$(\mathcal{A}_L \mathcal{P}^L_{L-1} u_{L-1})(ih_L, jh_L) = 0 \ , \qquad u_{L-1} \in \mathbb{R}^{n_{L-1}}$$

for odd i and j. This leads to the formulas

$$\begin{aligned} \pi^{I,J}_{\pm 1,\pm 1} &= -\frac{\alpha^{i\pm 1,j\pm 1}_{\mp 1,\mp 1} + \alpha^{i\pm 1,j\pm 1}_{\mp 1,0}\pi^{I,J}_{0,\pm 1} + \alpha^{i\pm 1,j\pm 1}_{0,\mp 1}\pi^{I,J}_{\pm 1,0}}{\alpha^{i\pm 1,j\pm 1}_{0,0}} \ , \\ \pi^{I,J}_{\pm 1,\mp 1} &= -\frac{\alpha^{i\pm 1,j\mp 1}_{\mp 1,\pm 1} + \alpha^{i\pm 1,j\mp 1}_{\mp 1,0}\pi^{I,J}_{0,\mp 1} + \alpha^{i\pm 1,j\mp 1}_{0,\pm 1}\pi^{I,J}_{\pm 1,0}}{\alpha^{i\pm 1,j\mp 1}_{0,0}} \ . \end{aligned}$$

Although the sequence of grids for the Dendy algorithm is the same as for the bilinear interpolation scheme, due to the matrix-dependent choice of the weights in (37) special basis functions on the coarser levels are constructed implicitly which are in general not identical to the nodal basis.

In comparison with bilinear interpolation, the Dendy interpolations perform much better in connection with symmetric interface problems. But for certain other problems they are still not robust enough.

4.3. Algebraic coarsening with matrix-dependent interpolations. Algebraic multigrid methods were first introduced in the early eighties by Brandt, McCormick, Ruge and Stüben [13, 14, 44, 45]. Here, first a grid is set up on the next coarser level by using algebraic information from $\mathcal{A}_L$ and then an appropriate interpolation scheme is defined. After computing $\mathcal{A}_{L-1}$ via the Galerkin identity the process is repeated until a sufficiently coarse level system is obtained analogously to the short algorithm in section 4.2. AMG is necessarily less efficient than highly specialized geometric multigrid solvers for elliptic problems on uniform rectangular grids. However, for more complicated cases with complex domains, AMG has been shown to behave robust and thus performs quite favorably in terms of operation count and CPU time. AMG also works for problems where geometric multigrid methods are impossible to design. AMG uses no sophisticated smoother, but only standard Gauß-Seidel. The robustness of AMG is obviously the merit of the appropriate chosen grid coarsening strategy and the associated interpolations. So it is a natural approach to examine the preconditioning properties of multilevel preconditioners with AMG-type interpolations.

Algebraic multigrid goes one step further than the matrix-dependent schemes. The grids should be nested again, but they need not to be uniform anymore. In fact, uniformity, if given for the finest grid, is in general not maintained in the process. We will nevertheless start with fine level discretizations based on regular grids Ω_h where h is a negative power of two to be able to compare the performance of AMG with the other two coarsening schemes which depend on regular grids. In general, it also is not predictable how many grids will be constructed by the AMG algorithm, i.e. $M \neq \log_2(h^{-1})$. It is more appropriate to think in terms of graphs in connection with algebraic multigrid than to think in terms of grids. In the following we will denote the set of indices of the grid corresponding to level L by N_L. For the two coarsening schemes defined above these sets were the numbers $1, 2, \ldots, (2^L - 1)^2$. Now we only demand that the index sets are nested

$$N_1 \subset N_2 \subset \ldots \subset N_{M-1} \subset N_M$$

Once again, to each grid point there corresponds a basis function with the same index. But in contrast to the geometric coarsening schemes, we now will assume that basis functions on different levels with equal indices are centered around the same point of Ω.

For an AMG algorithm, the sequence of matrices $\mathcal{A}_L$ must be constructed algebraically. The $\mathcal{A}_{L-1}, L = M, \ldots, 2$ are computed successively by selecting a subset of the unknowns of the level L system and by evaluating the *strength of the connections* between the unknowns in $\mathcal{A}_L$. The basis for our implementation is the AMG method of Ruge and Stüben [44, 45] which uses the assumption that the initial discretization matrix $\mathcal{A}_M$ is symmetric positive definite and a M-Matrix.

According to a well-known variational principle it is the best for a given interpolation to determine the coarse-grid discretization via Galerkin-coarsening. All error components lying in the range of the interpolation are then eliminated by a single coarse grid correction. In multigrid theory one has to take care that error components which are persistent to the smoother are well represented on coarser grids.

The effect of Gauß-Seidel iterations on symmetric positive definite $\mathcal{A}_M$ is well understood and can be used to guide the construction of the coarser level systems $\mathcal{A}_L$ for $L = M-1, \ldots, 1$. Gauß-Seidel smoothing is stalling whenever the error e_L is big in comparison to the residual r_L.

Because of $\mathcal{A}_L e_L = r_L$, we have $\mathcal{A}_L e_L \approx 0$ then. Or for a single unknown

$$(e_L)_i = -\frac{1}{(\mathcal{A}_L)_{ii}} \sum_{\substack{j=1 \\ j \neq i}}^{n_L} (\mathcal{A}_L)_{ij} (e_L)_j \ .$$

This sum may be splitted into the error components visible on the coarse grid (and thus eliminated by a single coarse grid correction step) and those which are not, i.e.

$$(e_L)_i = -\frac{1}{(\mathcal{A}_L)_{ii}} \left(\sum_{\substack{j \in C_L \\ j \neq i}} (\mathcal{A}_L)_{ij} (e_L)_j + \sum_{\substack{j \in F_L \\ j \neq i}} (\mathcal{A}_L)_{ij} (e_L)_j \right) \ .$$

Here $C_L := N_{L-1}$ and $F_L := N_L \setminus N_{L-1}$. If the second sum could be eliminated on all levels, AMG would be a direct solver. In this case, the ideal interpolation weights would be given by

$$(P_{L-1}^L e_{L-1})_i = \begin{cases} (e_{L-1})_i \ , & i \in C_L \\ -\dfrac{1}{(\mathcal{A}_L)_{ii}} \displaystyle\sum_{\substack{j \in C_L \\ j \neq i}} (\mathcal{A}_L)_{ij} (e_{L-1})_j \ , & i \in F_L. \end{cases} \tag{40}$$

Unfortunately, this ideal assumption can hardly be fulfilled when we want a geometric decrease of the number of grid points on each level. Nevertheless, we try to minimize the second sum by choosing the coarse grid points $C_L := N_{L-1}$ from N_L appropriately.

We will briefly review the coarse grid selection part of AMG, as introduced by Ruge and Stüben [44, 45]. For reasons of simplicity the level index L is omitted. Here, we have to define the set of strongly coupled neighbours S_i of a point i. Let

$$d(i, I) := \frac{1}{\max_{k \neq i}\{-\mathcal{A}_{ik}\}} \sum_{j \in I} -\mathcal{A}_{ij} \ ,$$

where I is any subset of N, and

$$S^i := \{j \in N | d(i, \{j\}) \geq \alpha\}, \qquad S^{i,T} := \{j \in N | i \in S^j\} \ .$$

The partitioning in fine and coarse grid points is performed in two phases on each level. There, we select coarse grid points in such a manner, that as many strong couplings as possible are taken into consideration.

Selection of coarse grid points:
Setup Phase I

1. Set $C = \emptyset$ and set $F = \emptyset$
2. **While** $C \cup F \neq N$ **do**
   Pick $i \in N \setminus (C \cup F)$ with maximal $|S^{i,T}| + |S^{i,T} \cap F|$
   If $|S^{i,T}| + |S^{i,T} \cap F| = 0$
   then set $F = N \setminus C$
   else set $C = C \cup \{i\}$ and set $F = F \cup (S^{i,T} \setminus C)$;
   endif

The measure $|S^{i,T}| + |S^{i,T} \cap F|$ is purely heuristical. The first term is associated to the total number of strongly coupled neighbours, the second one to the number of strongly coupled neighbours which are in F. Domains with the same discretization stencil for most nodes (typically inner nodes), tend to have the same value of the measure $|S^{i,T}| + |S^{i,T} \cap F|$ for them. Note that the action to pick an index in step 4 of the above algorithm is non-deterministic and allows different implementations, depending on the chosen underlying data structures, see also [15]. Furthermore, using dynamic data structures and incremental techniques, it is possible to implement the overall setup algorithm (i.e. phase I and II) to need a number of operations proportional to the number of fine grid unknowns. Further improvements should be possible, if one would handle nodes situated next to the boundary of the domain and inner nodes differently.

In a second phase the final C-point choice is made.

Selection of coarse grid points:
Setup Phase II

1. Set $T = \emptyset$
2. **While** $T \subset F$ **do**
   Pick $i \in F \setminus T$ and set $T = T \cup \{i\}$
   set $\tilde{C} = \emptyset$ and set $C^i = S_i \cap C$
   set $F^i = S_i \cap F$
   While $F^i \neq \emptyset$ **do**
   Pick $j \in F^i$ and set $F^i = F^i \setminus \{j\}$
   If $d(j, C^i)/d(i, \{j\}) \leq \beta$
   then if $|\tilde{C}| = 0$
   then set $\tilde{C} = \{j\}$ and set $C^i = C^i \cup \{j\}$
   else set $C = C \cup \{i\}$, set $F = F \setminus \{i\}$ and **Goto 2**
   endif
   endif
   set $C = C \cup \tilde{C}$, set $F = F \setminus \tilde{C}$

This second algorithm has to make sure, that each point in F is strongly coupled directly with points in C or at least with points in F, which are strongly coupled with points in C. Again, the strategy to force the set $\tilde{C}$ to contain at most one element is purely heuristic.

After the points N_L where divided into the sets F_L and C_L, we could define the interpolation as given in (40). Indeed, in the first AMG implementations it was defined like this. In the algorithm of Ruge and Stüben, a little more sophisticated interpolation is used, which gives much more better results in numerical experiments:

$$(41) \qquad (P_{L-1}^L e_{L-1})_i := \begin{cases} (e_{L-1})_i \ , & i \in C_L \\ -\dfrac{\sum_{j \in C_L^i} \left((\mathcal{A}_L)_{ij} + c_{ij}\right)(e_{L-1})_j}{(\mathcal{A}_L)_{ii} + c_{ii}} \ , & i \in F_L, \end{cases}$$

where

$$c_{ij} := \sum_{\substack{k \notin C_L^i \\ k \neq i}} \frac{(\mathcal{A}_L)_{ik}(\mathcal{A}_L)_{kj}}{(\mathcal{A}_L)_{ki} + \sum_{l \in C_L^i} (\mathcal{A}_L)_{kl}} \ .$$

Once the interpolation matrix $\mathcal{P}_{L-1}^L$ is constructed, the system matrix $\mathcal{A}_{L-1}$ is determined by the Galerkin identity (31). Then, the coarsening proceeds recursively according to the scheme in section 4.2 until the number of remaining unknowns equals one. Note that, in an elaborate implementation of AMG, the number of operations needed for the whole setup phase, i.e. the coarse grid selection and the computation of the interpolation and coarse grid matrices, is about 2-3 V-cycles of the corresponding multigrid method.

5. Numerical experiments. We now consider the results of our numerical experiments. We discretized problem (35) on the finest grid Ω_h by a finite element or finite difference method. Then, in a setup phase we computed the interpolation weights where necessary. With this weights we built up our multilevel preconditioned matrices $\mathcal{P}_M^E \mathcal{B}_M^E (\mathcal{P}_M^E)^T \mathcal{A}_M$ and computed their condition numbers κ.

From the condition number, an upper bound of the error reduction after i iteration steps of the associated conjugate gradient method (for symmetric problems) is given by $2\rho^i$ where

$$\rho := \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}$$

describes the worst case convergence rate of the method. But since often, if the eigenvalue spectrum is clustered somehow, a better convergence rate can be observed in practice, we also give the number of iteration steps *it* the conjugate gradient method needs to reduce the L^2-norm of the residual by a factor 10^{-14}. We also give the measured average reduction rate

$$\rho := \left(\frac{|r_{it}|_2}{|r_{it-10}|_2} \right)^{\frac{1}{10}}$$

over the last 10 iteration steps of the corresponding conjugate gradient method. As starting vector for the iteration we used random values.

Since, at least for the AMG coarsening process, the resulting grids and matrices on the coarser levels give an interesting insight to the method for singular perturbed problems and operators with locally varying coefficients, we also give certain figures describing the resulting grid sequence and show snapshots of some of the coarse grid matrices. Here, the size of a grid point indicates the coarsest level it belongs to and in the graphs of the matrices, large entries of the matrix correspond to graphically thicker displayed edges. Furthermore, if not indicated otherwise, we used the values $\alpha = 0.25$ and $\beta = 0.35$ in the AMG coarsening process since these values turned out in our experiments to produce mostly quite robust results, at least for the case $\Omega = [0,1]^2$ with Dirichlet boundary conditions.

5.1. Multilevel preconditioning of the Laplacian. In the following, we consider the Laplace operator

$$-\Delta := -\frac{\delta^2}{\delta x_1^2} - \frac{\delta^2}{\delta x_2^2}. \tag{42}$$

Here, we discretize on the finest grid Ω_h by three different approaches, i.e. by bilinear finite elements resulting in the usual 9-point stencil, by finite differences resulting in the well known 5-point stencil, and by a finite difference method, giving a 5-point stencil which is rotated by an angle of 45 degrees with respect to the coordinate axes. We then study the convergence properties of the different multilevel preconditioners.

5.1.1. Finite element discretization. If we discretize the Laplacian using bilinear finite elements, we obtain the usual 9-point stencil

$$\frac{1}{3h^2}\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}. \tag{43}$$

Interestingly, in this case, all three coarsening schemes result in almost the same enlarged system matrices, since, also for Dendy's approach and our implementation of AMG, the standard bilinear interpolation is constructed in the interior of the domain. For Dendy's approach, this can be seen immediately by putting the 9-point stencil of the Laplacian in the formulas in section 4.2. For the AMG method, our specific implementation, i.e. the ordering in which the points are picked in the setup phase, is the reason for it. Compare also Figure 1, where the hierarchy of constructed grids and the graph of the coarse grid matrix after four coarsening steps is given for the AMG approach. We see that AMG constructs a regular sequence of grids. However, due to boundary effects, its interpolation weights next to the boundary $\delta\Omega$ are not identical to those of bilinear interpolation and the Dendy scheme. As a consequence the condition numbers and convergence rates are slightly different.

Table 1 shows the resulting condition numbers, convergence rates and iteration counts. Results for the Dendy scheme are omitted, because they are identical with bilinear interpolation for this problem.

TABLE 1
Condition numbers, convergence rates and iteration counts for the preconditioning of the Laplace operator, 9-point stencil on the finest grid.

h^{-1}		MDS	AMG
8	κ	2.96	2.91
	ρ	0.15	0.14
	it	22	22
16	κ	3.59	3.55
	ρ	0.25	0.25
	it	30	30
32	κ	4.07	4.04
	ρ	0.31	0.30
	it	35	35
64	κ	4.46	4.43
	ρ	0.34	0.34
	it	39	39
128	κ	4.77	4.76
	ρ	0.36	0.36
	it	42	42

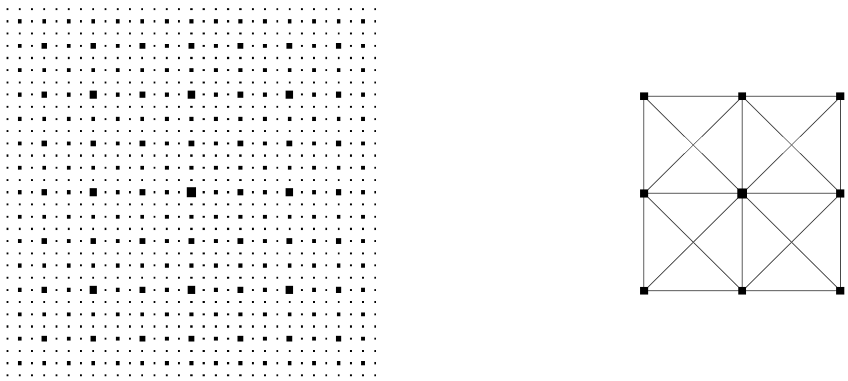


FIG. 1. *AMG-Hierarchy of grids for the Laplace problem with 9-point stencil, initial meshwidth $h = 1/32$, and resulting coarse grid matrix after the third coarsening step.*

5.1.2. Finite difference discretization. Now, we use a finite difference method for the discretization of the Laplacian on the fine grid which corresponds to the well-known 5-point stencil

$$\frac{1}{h^2}\begin{bmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix}. \tag{44}$$

Once again, the Dendy scheme reproduces the weights of the bilinear interpolation. Its results are therefore omitted. The corresponding condition numbers, convergence rates and iteration counts are given in Table 2. They are in general by about a factor 1.3

worse than the values from subsection 5.1.1.

TABLE 2
Condition numbers, convergence rates and iteration counts for the preconditioning of the Laplace operator, 5-point stencil on the finest grid.

h^{-1}		MDS	AMG
8	κ	4.02	4.32
	ρ	0.17	0.18
	it	28	28
16	κ	4.88	5.73
	ρ	0.33	0.34
	it	37	38
32	κ	5.65	6.12
	ρ	0.38	0.38
	it	43	41
64	κ	6.29	6.95
	ρ	0.41	0.42
	it	47	47
128	κ	6.83	8.20
	ρ	0.43	0.42
	it	51	52

Figure 2 shows the hierarchy of grids constructed by the AMG approach for the discretization with meshwidth $h = 1/32$. Now, in contrast to the 9-point stencil case, the coarsening structure is not more totally regular and coarsening takes place mainly with respect to 45 degree rotated coordinate axes. Furthermore, boundary effects become visible. Note that now 6 coarsening levels are produced by our AMG method. Figure 2 also shows the graph of the coarse grid matrix after three coarsening steps.

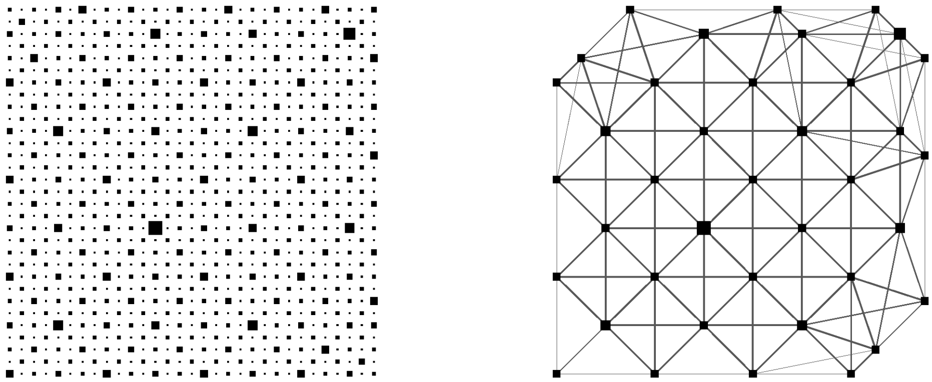


FIG. 2. *AMG-Hierarchy of grids for the Laplace problem with 5-point stencil, initial meshwidth* $h = 1/32$*, with 6 resulting coarse levels and coarse grid matrix after four coarsening steps.*

5.1.3. Rotated finite difference discretization. It is interesting to use a finite difference discretization for the Laplacian on the fine grid that corresponds to the 5-point stencil which is rotated by 45 degrees, i.e.

$$\frac{1}{2h^2}\begin{bmatrix} -1 & & -1 \\ & 4 & \\ -1 & & -1 \end{bmatrix}.$$

Note that the associated fine grid matrix $\mathcal{A}_h$ is (after permutation) a fully decoupled 2 by 2 block diagonal matrix. In other words, the set of grid points (ih, jh) where $i + j$ is even and the set of grid points with odd $i + j$ are not connected in the graph of $\mathcal{A}_h$. The condition numbers, convergence rates and iteration counts for our approaches are given in Table 3.

TABLE 3

Condition numbers, convergence rates and iteration counts for the the preconditioning of the Laplace operator, rotated 5-point stencil on the finest grid.

h^{-1}		MDS	AMG
8	κ	17.3	3.67
	ρ	0.09	0.16
	it	26	24
16	κ	77.6	4.49
	ρ	0.34	0.26
	it	61	32
32	κ	341	5.73
	ρ	0.60	0.33
	it	129	38
64	κ	1466	5.92
	ρ	0.76	0.36
	it	269	43
128	κ	6213	6.53
	ρ	0.86	0.40
	it	556	48

We see that the preconditioner based on bilinear interpolation, i.e. the BPX-type method, gives larger and larger condition numbers and convergence rates for successively finer discretizations. Its convergence behaviour is dependent on the mesh size and the unknowns. Its condition number grows with h^{-2} like that of the unpreconditioned fine grid matrix $\mathcal{A}_h$ itself. Once again, the interpolation weights according to the Dendy scheme reproduce the bilinear interpolation and its results are omitted. However the method based on the AMG coarsening shows a condition number which is independent on the fine grid mesh size. Its is slightly worse than that for the 9-point stencil but slightly better than that for the usual 5-point stencil. The AMG approach has no problems with the decoupled fine grid stiffness matrix whereas the other methods do.

It maintains a two block structure under the coarsening process, as it can be seen from Figure 3. Here, the hierarchy of grids constructed by the AMG approach starting with the discretization with initial meshwidth $h = 1/32$ is given. Now, 5 levels are produced by our AMG method. The graph of the coarse grid matrix after three coarsening steps consists of two non-connected subgraphs showing that the 2 by 2 diagonal block property of the fine level matrix is inherited to the coarser levels also.

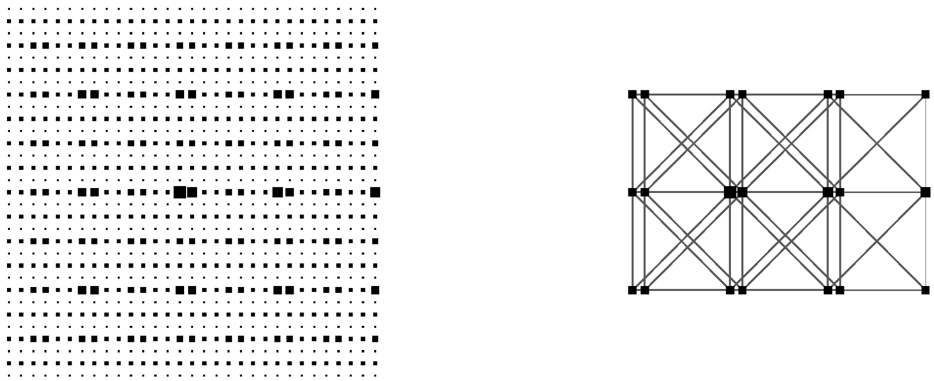


FIG. 3. *AMG-Hierarchy of grids for the Laplace problem discretized with rotated 5-point stencil, initial meshwidth $h = 1/32$, with 5 resulting coarse levels and coarse grid matrix graph after three coarsening steps.*

5.2. Multilevel preconditioning of the Helmholtz operator. Now, we consider the Helmholtz operator

$$-\Delta + \varepsilon \cdot \mathrm{id}. \tag{45}$$

We discretize it on the finest grid Ω_h by the finite difference method which results in the 5-point stencil [4]

$$\frac{1}{h^2}\begin{bmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix} + \varepsilon \begin{bmatrix} & & \\ & 1 & \\ & & \end{bmatrix} \tag{46}$$

and study the convergence properties of the different multilevel preconditioners for varying values ε. The results are given in Table 4. We restrict our presentation here to the case of the finite difference discretization, since, in further numerical experiments, the finite element discretization for the finest level did not change things substantially. The results were analogous. Note that for $\varepsilon \to \infty$ basically the identity appears (after scaling). If ε equals $-\lambda_{\min}(-\Delta_h)$ then the fine grid matrix $\mathcal{A}_h$ gets singular[5].

From Table 4 the following can be seen: First, for positive values of ε, the preconditioner based on bilinear interpolation, i.e. the BPX-type method gives rising condition

[4] To be able to treat the case $\varepsilon \to \infty$, we implemented $-\frac{1}{\varepsilon}\Delta + \mathrm{id}$ instead.

[5] The same holds for all other eigenvalues of the discrete Laplacian. For values $\varepsilon < -\lambda_{\min}(-\Delta_h)$ we would have an indefinite matrix $\mathcal{A}_h$.

TABLE 4

Condition numbers, convergence rates and iteration counts for the preconditioning of the Helmholtz operator, 5-point stencil on the finest grid.

		$\varepsilon = -19$			$\varepsilon = 100$		
h^{-1}		MDS	Dendy	AMG	MDS	Dendy	AMG
8	κ	33.6	15.8	15.1	3.54	4.21	4.91
	ρ	0.14	0.20	0.25	0.18	0.21	0.20
	it	32	31	31	24	27	27
16	κ	28.4	10.9	24.9	4.92	6.07	7.40
	ρ	0.36	0.37	0.34	0.32	0.32	0.32
	it	42	40	41	35	38	38
32	κ	29.2	10.5	20.3	5.66	7.26	8.11
	ρ	0.40	0.40	0.41	0.38	0.40	0.43
	it	47	45	45	42	44	47
64	κ	30.8	10.8	21.9	6.06	7.98	8.13
	ρ	0.42	0.42	0.42	0.40	0.41	0.45
	it	52	49	50	46	49	52
128	κ	32.5	11.3	43.5	6.44	8.46	8.71
	ρ	0.44	0.44	0.57	0.43	0.43	0.45
	it	56	53	59	50	53	54

		$\varepsilon = 10^6$			$\varepsilon = \infty$		
h^{-1}		MDS	Dendy	AMG	MDS	Dendy	AMG
8	κ	3.52	3.00	4.00	3.51	3.00	1.00
	ρ	0.01	0.01	0.01	0.06	0.00	0.00
	it	13	10	11	8	3	1
16	κ	5.44	4.00	4.00	5.44	4.00	1.00
	ρ	0.13	0.01	0.01	0.13	0.00	0.00
	it	28	13	11	22	5	1
32	κ	7.58	5.01	4.00	7.57	5.00	1.00
	ρ	0.29	0.04	0.01	0.25	0.00	0.00
	it	42	26	12	33	8	1
64	κ	9.82	6.06	4.00	9.78	6.00	1.00
	ρ	0.41	0.10	0.01	0.34	0.01	0.00
	it	54	20	13	42	10	1
128	κ	12.3	7.32	5.00	12.0	7.00	1.00
	ρ	0.51	0.16	0.06	0.42	0.02	0.00
	it	65	27	19	51	13	1

numbers and convergence rates for large values ε and successively finer discretizations. For large ε its convergence behaviour seems to be dependent on the mesh size and the unknowns. Especially for the case $\varepsilon = \infty$, i.e. the identity (after scaling), the condition number seems to grow faster than $\#\text{levels} = \log_2(h^{-1})$. For the Dendy approach $\kappa = \#\text{levels}$ holds exactly there. By plugging in the corresponding values into the formulas in section 4.2 we see that the interpolation weights equal zero then, except for the central weights which are 1.

However, the method based on the AMG coarsening shows a condition number which is independent of both, the fine grid mesh size *and* the value of ε. For $\varepsilon = \infty$, it even stops the coarsening process, recognizing that the unknowns are totally decoupled.

For the case $\varepsilon = -19$ all methods still behave not to bad, with Dendy's approach relatively the best. However, as more we approach the first eigenvalue of the Laplacian, as worse the convergence results get for all three methods[6]. Altogether, the AMG based preconditioner provides an efficient and robust solution method for the Helmholtz problem for values of ε sufficiently away from the first eigenvalue of the discrete Laplacian.

For the initial meshwidth of the discretization $h = 1/32$ and the values $\varepsilon = 100, 1000$ and 10^6, Figure 4 shows the hierarchy of grids constructed by the AMG approach. Here, 7, 7 and 3 coarse levels are set up by the method. Interestingly, for the case $\varepsilon = \infty$, the AMG process does no refinement at all. It therefore detects that the identity operator needs no coarsening. The coarse grid matrices for the values $\varepsilon = 100, 1000$ and 10^6 after 4, 3 and 2 coarsening steps, respectively, are given in Figure 5. We see for the case $\varepsilon = 10^6$ that the coarse grid matrix after two coarsening steps is basically diagonal.

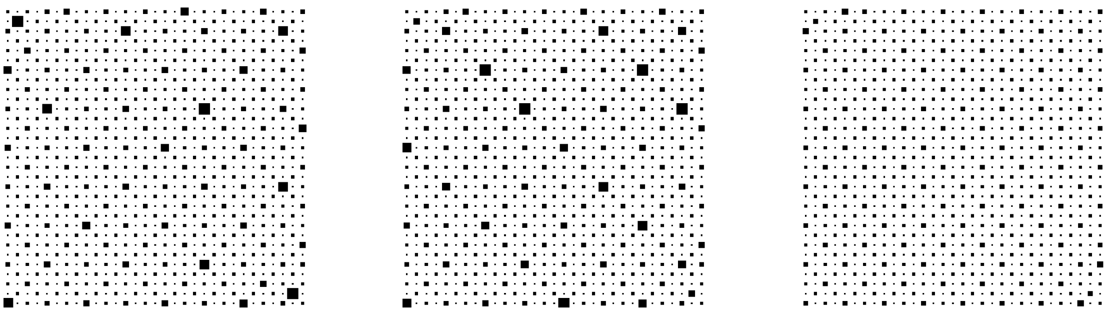


FIG. 4. *AMG-Hierarchy of grids, initial meshwidth $h = 1/32$, for the Helmholtz problem with ε=100, 1000 and 10^6, discretized by 5-point stencils. 7, 7 and 3 resulting coarse levels.*

5.3. Multilevel preconditioning for diffusion problems with jumps in the coefficients. Now we consider so-called interface problems, which arise for example in the area of porous media flow or in contact problems and related questions where media with different material properties stick together. The underlying operator is

$$d(x_1, x_2) \cdot \Delta, \tag{47}$$

[6] Note that by another matrix-dependent coarsening process which is based on a directional splitting of the 2D Helmholtz operator involving a energy based interpolation that stems from the resulting 1D problems, very good condition numbers and convergence rates can be gained also for values of ε , which are extremely close to the first eigenvalue of the discrete Laplacian. For details, see [24].

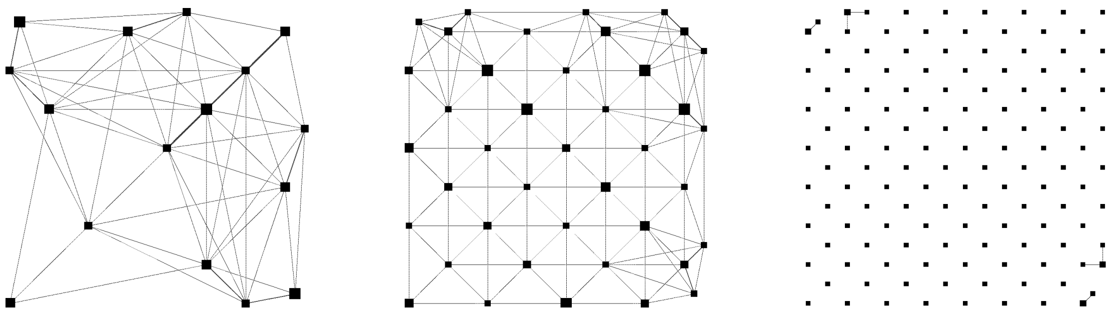


FIG. 5. *Coarse grid matrix graph for the the Helmholtz problem with* ε*=100, 1000 and* 10^6*, discretized by 5-point stencils. Snapshots after 4, 3 and 2 coarsening steps, respectively.*

where d is assumed to be a piecewise constant scalar function, which possesses large jumps in Ω. In the discrete case, this means that the diffusion coefficient can vary by a few orders of magnitude within one mesh width.

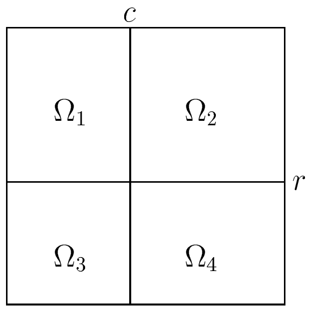
In the following, we assume that the jumps in the diffusion coefficient are aligned exactly with the lines of the finest grid and the diffusion value is not changing within one cell of the discretization. This implies that the diffusion coefficient can have different values at most in the four quadrants of the support of a finest level basis function. Therefore, we obtain in the point (ih, jh) the stencil

$$\frac{1}{3h^2}\begin{bmatrix} -d_1 & -(d1+d2)/2 & -d_2 \\ -(d_1+d_3)/2 & 2(d_1+d_2+d_3+d_4) & -(d_2+d_4)/2 \\ -d_3 & -(d_3+d_4)/2 & -d_4 \end{bmatrix}, \tag{48}$$

where d_1, d_2, d_3 and d_4 denote the values of the diffusion coefficient in the quadrants of the support of the respective basis function.

Now, we first consider the *four corner problem.* Here, the domain Ω is subdivided by the lines $x_1 = r$ and $x_2 = c$ into four subdomains (see figure 6). In each subdomain $\Omega_i, i = 1, \dots 4$, the diffusion coefficient can take a different constant value.

FIG. 6. *Decomposition of the domain* Ω *for the four corner model problem.*



In our experiments, we choose d to be

$$1 \text{ in } \Omega_1, \Omega_4 \quad \text{and} \quad 10^{\varepsilon} \text{ in } \Omega_2, \Omega_3$$

and study the convergence properties of the different multilevel preconditioners for varying values ε. There, we consider two different cases. In the first case, the lines subdivide Ω exactly into four equally spaced subdomains. This implicates that the so-called cross point is contained in all coarser grids, if standard coarsening is assumed. In the second case, we shift the jump lines by one mesh size h on the finest grid. Thus, the jumps and especially the cross point belongs only to the finest grid, if standard coarsening is assumed, and are not more resolved on the coarser levels.

This model problem is known to cause severe problems to standard multigrid methods, whereas Dendy's approach is able to cope with it. As a matter of fact, matrix-dependent geometric coarsening methods were first developed for such kinds of problems, see also [20] or [30], pp 212-217. In [21] and [39] and the references cited therein, there also exists some convergence theory to it.

TABLE 5

Condition numbers, convergence rates and iteration counts for the preconditioning of the four corner problem with jumps located at $r = c = 1/2$, finite element discretization (48) on the finest grid.

h^{-1}		$\varepsilon = 1$			$\varepsilon = 2$			$\varepsilon = 4$		
		MDS	Dendy	AMG	MDS	Dendy	AMG	MDS	Dendy	AMG
8	κ	3.95	3.95	3.93	4.40	4.40	4.39	4.47	4.47	4.45
	ρ	0.16	0.17	0.17	0.15	0.15	0.16	0.12	0.13	0.14
	it	25	25	25	26	26	25	27	27	25
16	κ	5.26	5.26	5.25	6.21	6.21	6.17	6.35	6.35	4.45
	ρ	0.28	0.27	0.27	0.26	0.26	0.27	0.25	0.25	0.27
	it	33	33	33	35	35	33	38	38	35
32	κ	6.58	6.58	6.59	8.28	8.28	8.20	8.56	8.56	8.46
	ρ	0.32	0.32	0.31	0.31	0.31	0.32	0.31	0.32	0.33
	it	38	38	41	37	41	40	45	45	42
64	κ	7.90	7.90	7.92	10.6	10.6	10.5	11.1	11.1	11.0
	ρ	0.34	0.34	0.36	0.34	0.34	0.36	0.34	0.35	0.37
	it	43	43	44	46	46	45	50	50	47
128	κ	9.18	9.18	9.23	13.3	13.3	13.1	14.0	14.0	13.8
	ρ	0.36	0.36	0.38	0.36	0.36	0.40	0.46	0.46	0.42
	it	47	47	48	50	50	53	57	57	58

The results of our experiments are given in Tables 5 and 6. We see that the preconditioner based on bilinear interpolation gives rising condition numbers and convergence rates for larger values ε and successively finer discretizations. Also for fixed h but varying ε, we see that the condition numbers are dependent on ε for the second case $r = c = 1/2 + h$, whereas they are not in the first case $r = c = 1/2$.

In contrast to that, for both, the approach due to Dendy and the AMG based method, the condition numbers remain constant for fixed h but varying values of ε. However, the results of the Laplacian can not be reached. Interestingly, the second case, i.e. $r = c = 1/2 + h$, gives even slightly better condition numbers than the first one.

TABLE 6

Condition numbers, convergence rates and iteration counts for the preconditioning of the four corner problem with jumps located at $r = c = 1/2 + h$, finite element discretization (48) on the finest grid.

h^{-1}		$\varepsilon = 1$			$\varepsilon = 2$			$\varepsilon = 4$		
		MDS	Dendy	AMG	MDS	Dendy	AMG	MDS	Dendy	AMG
8	κ	4.98	3.08	5.76	6.93	3.05	5.73	7.31	3.02	5.71
	ρ	0.18	0.17	0.21	0.17	0.16	0.22	0.16	0.17	0.20
	it	28	25	30	30	26	30	32	28	30
16	κ	7.71	4.08	8.05	12.11	4.42	5.54	13.1	4.48	5.56
	ρ	0.34	0.26	0.35	0.35	0.28	0.30	0.38	0.27	0.32
	it	42	33	43	49	35	39	54	38	41
32	κ	10.2	5.15	10.6	20.3	6.03	8.23	26.1	6.11	8.15
	ρ	0.42	0.33	0.41	0.49	0.35	0.39	0.55	0.32	0.37
	it	52	39	50	68	42	48	76	46	49
64	κ	12.0	6.24	10.5	29.3	8.21	10.3	46.9	8.46	8.39
	ρ	0.50	0.39	0.46	0.58	0.40	0.44	0.60	0.38	0.43
	it	61	45	58	87	51	54	100	56	56
128	κ	13.1	7.39	11.7	36.0	10.5	10.8	74.4	11.8	10.7
	ρ	0.54	0.41	0.49	0.65	0.44	0.47	0.64	0.44	0.47
	it	69	50	65	103	59	64	129	66	69

The hierarchy of grids for the AMG approach and the graphs of some resulting coarse grid matrices are given in Figure 7.

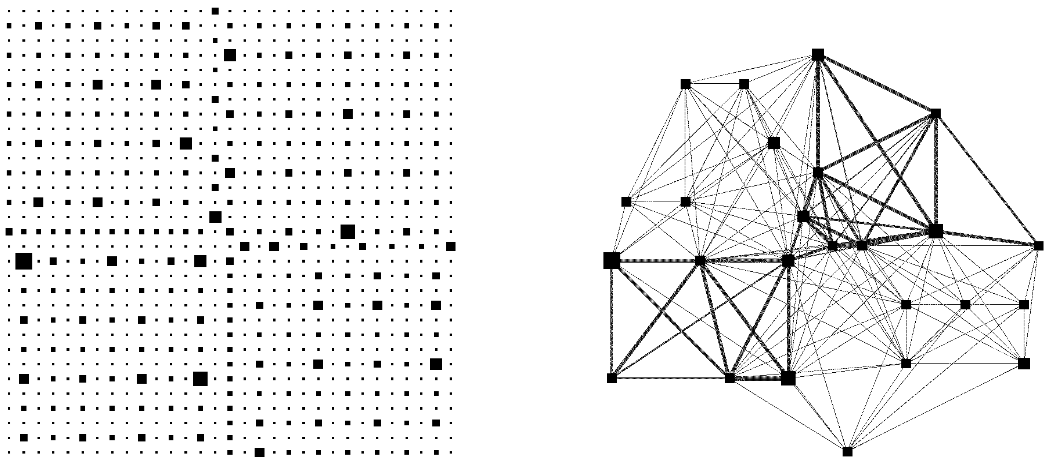


FIG. 7. *AMG-Hierarchy of grids for the four-corner problem $h = 1/32$, $r, c = 1/2 + h$ and $\varepsilon = 4$, with 6 resulting coarse levels and graph of the coarse grid matrix after three coarsening steps.*

Next, we consider the so-called staircase problem due to [1]. Here, we use however Dirichlet boundary conditions. Figure 8 gives the decomposition of the domain Ω into subdomains with different diffusion coefficients. They take the value 1 in Ω_1 and the value 10^ε in Ω_2. Here, one length unit corresponds to $h^{-1}/16$ mesh widths.

FIG. 8. *Decomposition of the domain Ω for the staircase problem.*

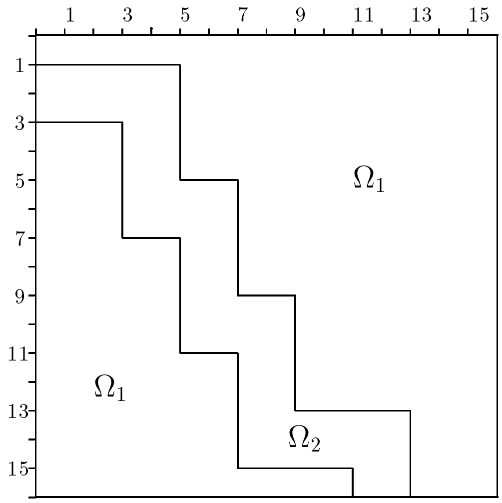


Table 7 gives the measured convergence results. We see that the preconditioner based on bilinear interpolation possesses the worst condition numbers. The results of the Dendy and the AMG approach are better.

TABLE 7
Condition numbers, convergence rates and iteration counts for the preconditioning of the staircase problem with Dirichlet boundary conditions, finite element discretization (48) on the finest grid.

h^{-1}		$\varepsilon = 1$			$\varepsilon = 2$			$\varepsilon = 4$		
		MDS	Dendy	AMG	MDS	Dendy	AMG	MDS	Dendy	AMG
16	κ	10.7	5.23	7.12	22.0	7.85	6.55	25.8	8.74	6.62
	ρ	0.37	0.26	0.35	0.45	0.29	0.36	0.45	0.26	0.34
	it	46	33	42	56	35	41	62	36	42
32	κ	11.1	5.77	7.48	22.9	8.64	7.64	26.9	9.60	7.76
	ρ	0.41	0.34	0.41	0.45	0.34	0.41	0.44	0.31	0.42
	it	49	39	47	60	40	47	67	42	50
64	κ	11.8	6.24	8.58	24.2	9.23	8.82	28.4	10.2	8.97
	ρ	0.43	0.37	0.43	0.44	0.37	0.45	0.47	0.37	0.45
	it	54	43	52	64	46	53	71	48	56
128	κ	12.4	6.62	9.33	25.2	9.66	11.0	29.6	10.7	11.3
	ρ	0.44	0.39	0.44	0.46	0.39	0.49	0.48	0.39	0.48
	it	57	47	53	68	50	59	76	53	62

The hierarchy of grids for the AMG approach and the graphs of resulting coarse grid matrices are given in Figure 9.

At last, we study a diffusion problem where the interface lines possess a spiral shaped structure. The subdivision of the domain is given in Figure 10. Again, we use Dirichlet boundary conditions. The value of the diffusion coefficient is 1 in Ω_1 and 10^ε in Ω_2. Now, one length unit corresponds to $h^{-1}/32$ mesh widths.

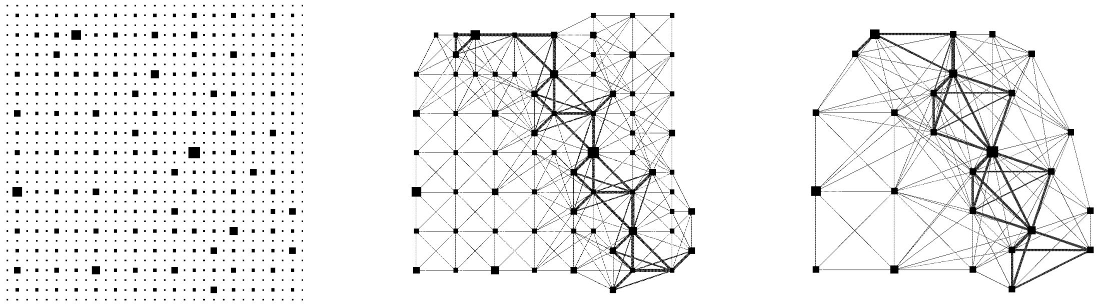
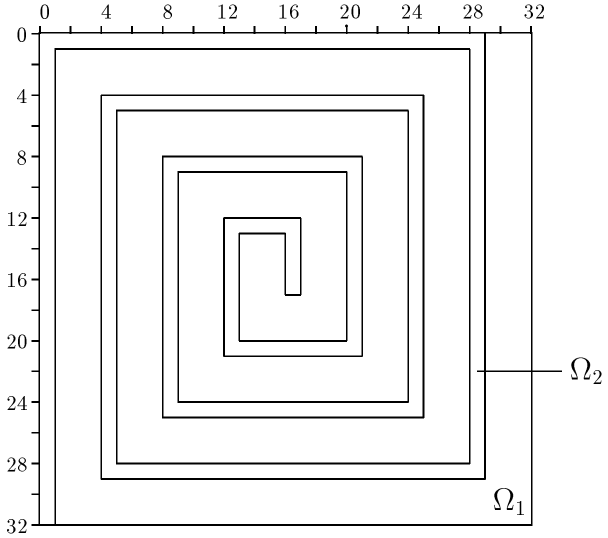


FIG. 9. *AMG-Hierarchy of grids for staircase problem, initial meshwidth $h = 1/32$ and $\varepsilon = 4$, with 6 resulting coarse levels and graphs of the coarse grid matrices after two and three coarsening steps.*

FIG. 10. *Decomposition of the domain Ω for the spiral problem.*



For such a problem, conventional multigrid methods are known to have quite bad convergence rates. This is also reflected for the preconditioner based on the bilinear interpolation in Table 8. Interestingly, also the preconditioner based on Dendy's method is not robust any more. The resulting condition numbers are smaller than the ones for the bilinear interpolation coarsening, but they still grow with rising ε. However the AMG based method behaves fully robust and produces for all values of ε basically the same value for fixed h.

The reason why the AMG approach behaves so favourably lies in its possibility to adapt itself on the coarser levels to the physical nature of the problem under consideration by setting up coarser grids which allow to resolve features of the operator also on coarser levels sufficiently. This can be observed in Figure 11. There, the hierarchy of grids for the AMG approach and the graphs of some resulting coarse grid matrices are given. We clearly see that the coarse grid matrices exhibit the spiral structure of the problem. They become nearly tridiagonal matrices where strong couplings follow the spiral line. Then, basically semi-coarsening along the spiral takes place on the coarser levels.

TABLE 8

Condition numbers, convergence rates and iteration counts for the preconditioning of the spiral problem, finite element discretization (48) on the finest grid.

h^{-1}		$\varepsilon=1$			$\varepsilon=2$			$\varepsilon=4$		
		MDS	Dendy	AMG	MDS	Dendy	AMG	MDS	Dendy	AMG
32	κ	15.8	6.30	8.56	126	38.0	10.2	1960	670	8.76
	ρ	0.52	0.37	0.46	0.54	0.39	0.45	0.88	0.80	0.44
	it	68	44	56	120	68	55	173	98	55
64	κ	16.8	6.79	11.4	137	40.3	10.5	2140	715	10.7
	ρ	0.49	0.39	0.46	0.54	0.44	0.49	0.94	0.72	0.48
	it	72	48	59	126	74	59	199	117	62
128	κ	17.8	7.21	12.6	145	42.2	12.6	2264	749	12.8
	ρ	0.53	0.42	0.52	0.57	0.44	0.49	0.76	0.68	0.51
	it	77	52	64	133	78	69	215	128	76

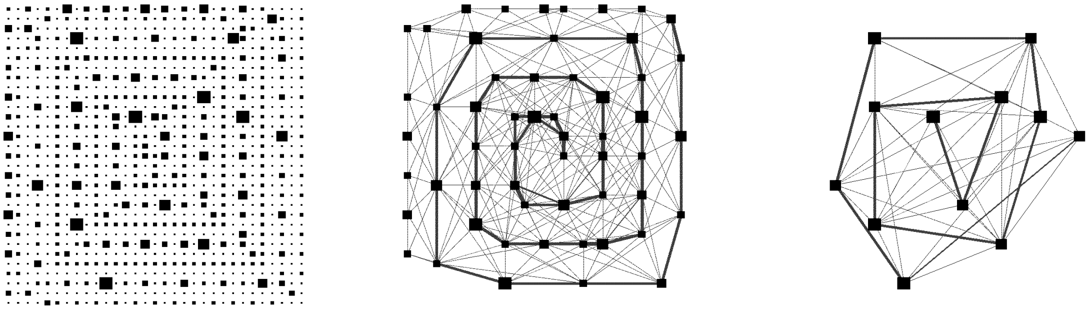


FIG. 11. *AMG-Hierarchy of grids, initial meshwidth $h = 1/32$ and $\varepsilon = 4$, with 8 resulting coarse levels and graphs of the coarse grid matrices after three and five coarsening steps.*

5.4. Multilevel preconditioning for anisotropic diffusion problems. Now, we consider the operator

$$\varepsilon \frac{\delta^2}{\delta x_1^2} + \frac{\delta^2}{\delta x_2^2}, \tag{49}$$

which describes for values $\varepsilon \neq 1$ anisotropic diffusion processes. We discretize it on the finest grid Ω_h by the finite difference method and obtain the 5-point stencil

$$\frac{1}{h^2}\begin{bmatrix} & -1 & \\ -\varepsilon & 2+2\varepsilon & -\varepsilon \\ & -1 & \end{bmatrix}. \tag{50}$$

The convergence properties of the different multilevel preconditioners for varying values ε can be seen from Table 9.

As it can be expected, the condition numbers and convergence rates of the preconditioner based on bilinear interpolation deteriorate for decaying ε. The same holds also for the preconditioner based on Dendy's approach[7] which leads to the same results,

[7] However, if additionally preconditioners corresponding to a line smoother or an ILU-smoother are

TABLE 9

Condition numbers, convergence rates and iteration counts for preconditioning of the anisotropic diffusion problem, finite difference discretization 50 on the finest grid.

h^{-1}		$\varepsilon = 1$			$\varepsilon = 0.9$			$\varepsilon = 0.5$		
		MDS	Dendy	AMG	MDS	Dendy	AMG	MDS	Dendy	AMG
8	κ	4.02	4.02	4.32	4.21	4.21	4.32	6.06	6.06	4.86
	ρ	0.16	0.16	0.19	0.27	0.27	0.20	0.22	0.22	0.20
	it	28	28	28	31	31	28	32	32	29
16	κ	4.88	4.88	5.73	5.18	5.18	5.62	8.56	8.56	5.57
	ρ	0.33	0.33	0.34	0.36	0.36	0.31	0.44	0.44	0.32
	it	37	37	38	39	39	37	47	47	37
32	κ	5.65	5.65	6.12	6.06	6.06	6.84	10.4	10.4	6.82
	ρ	0.38	0.38	0.38	0.39	0.39	0.41	0.50	0.50	0.38
	it	43	43	41	40	40	43	51	51	42
64	κ	6.29	6.29	6.95	6.79	6.79	8.50	11.9	11.9	7.05
	ρ	0.41	0.41	0.42	0.42	0.42	0.42	0.53	0.53	0.41
	it	47	47	47	44	44	49	57	57	48
128	κ	6.83	6.83	8.20	7.41	7.40	9.58	13.1	13.1	7.93
	ρ	0.43	0.43	0.42	0.45	0.45	0.46	0.55	0.55	0.45
	it	51	51	52	47	47	55	62	62	52

h^{-1}		$\varepsilon = 10^{-2}$			$\varepsilon = 10^{-3}$			$\varepsilon = 0$		
		MDS	Dendy	AMG	MDS	Dendy	AMG	MDS	Dendy	AMG
8	κ	37.6	37.6	4.14	42.0	42.0	4.11	42.5	37.7	2.87
	ρ	0.19	0.19	0.16	0.26	0.26	0.08	0.07	0.03	0.00
	it	45	45	23	42	42	20	25	15	4
16	κ	146	146	5.35	210	210	5.30	222	181	3.48
	ρ	0.68	0.68	0.30	0.82	0.82	0.24	0.33	0.11	0.02
	it	137	137	33	132	132	32	77	43	10
32	κ	344	346	6.68	880	880	5.49	1070	827	3.98
	ρ	0.88	0.88	0.38	0.91	0.91	0.32	0.81	0.28	0.11
	it	257	257	40	310	310	37	220	87	24
64	κ	523	523	7.57	2657	2657	6.48	4871	3643	4.39
	ρ	0.91	0.91	0.40	0.92	0.92	0.37	0.87	0.44	0.25
	it	342	342	45	672	672	42	487	175	33
128	κ	628	628	8.23	4973	4973	7.53	21516	15684	4.72
	ρ	0.92	0.92	0.45	0.96	0.96	0.40	0.94	0.51	0.29
	it	392	392	52	1022	1022	48	1049	353	38

except for the case $\varepsilon = 0$. Here, the Dendy approach leads to stencils which are equivalent to semi-coarsening *but only on every second line.* However, the preconditioner based on the AMG approach is not affected if ε decays. Of course, this is due to the semi-coarsening effect of the AMG coarsening process, as it can be seen from Figures 12. There, the hierarchy of grids for the AMG approach and the graphs of resulting coarse grid matrices are given for the values $\varepsilon = 0.9,\ 0.1$ and 0. Note that for $\varepsilon = 0$ only 5 levels are constructed due to pure semi-coarsening. On the coarsest level only a diagonal matrix associated to the middle line unknowns appears, the problem is fully decoupled and AMG stops correctly the coarsening process. The corresponding condition numbers are just that of the one-dimensional problem with MDS-preconditioning applied to it.

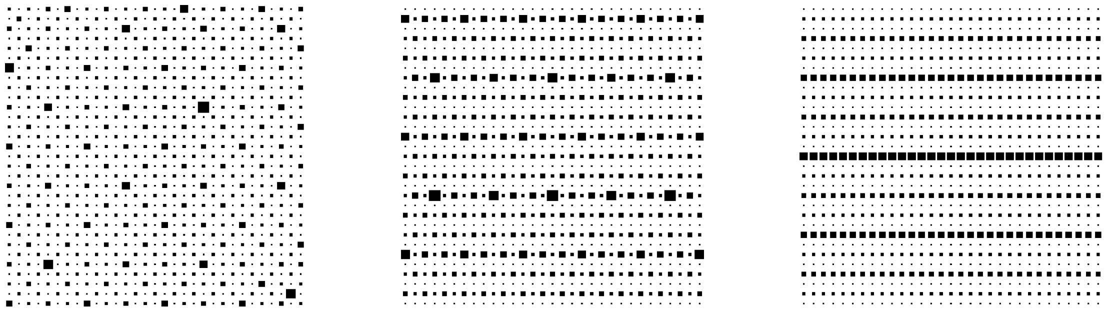


FIG. 12. *AMG-Hierarchy of grids for the anisotropy problem, initial meshwidth* $h = 1/32$*,* $\varepsilon = 0.9, 0.1$ *and 0 with 6, 7 and 4 resulting coarse levels.*

Since the simple anisotropic diffusion problem produced well expected results and gives no further insight into our preconditioners than the expected ones, we turn to a diffusion problem with locally varying anisotropy. To this end, we consider the problem illustrated by Figure 13. The anisotropy changes direction along the diagonal line of Ω. The underlying operator is $\varepsilon_1 \frac{\delta^2}{\delta x_1^2} + \varepsilon_2 \frac{\delta^2}{\delta x_2^2}$, where $\varepsilon_1 = 1, \varepsilon_2 = 0$, in Ω_1 and $\varepsilon_1 = 0, \varepsilon_2 = 1$, in Ω_2.

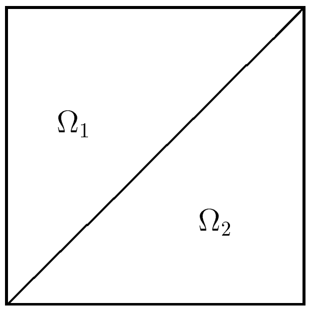


FIG. 13. *Decomposition of the domain* Ω *for the locally varying anisotropy problem.*

For the discretization, we used finite differences again. Table 10 gives the results obtained for the different preconditioning methods. We see that the condition numbers

applied on each level, good condition numbers can be regained for this model problem, at least for the multiplicative Schwarz method.

and convergence rates of all three methods deteriorate with finer mesh sizes h. However the AMG based approach seems to be less sensitive.

TABLE 10
Condition numbers, convergence rates and iteration counts for the preconditioning of the diffusion operator with jumping anisotropy, finite differences discretization on the finest grid.

h^{-1}		MDS	Dendy	AMG
8	κ	33.8	34.3	4.93
	ρ	0.23	0.22	0.23
	it	47	46	30
16	κ	189	202	7.57
	ρ	0.65	0.61	0.35
	it	150	149	41
32	κ	963	1198	12.2
	ρ	0.88	0.85	0.44
	it	458	464	53
64	κ	4652	6640	16.7
	ρ	0.98	0.97	0.49
	it	1078	1116	65
128	κ	20391	24676	25.1
	ρ	0.98	0.98	0.54
	it	2380	2512	82

By means of the AMG process, semi-coarsening is now performed locally. This is illustrated in Figure 14 where the produced hierarchy of grids and the graph of the coarse grid matrix after two coarsening steps are shown. Whether a different choice of α than 0.25 might help to further improve the convergence behaviour of AMG for this example is an open question yet.

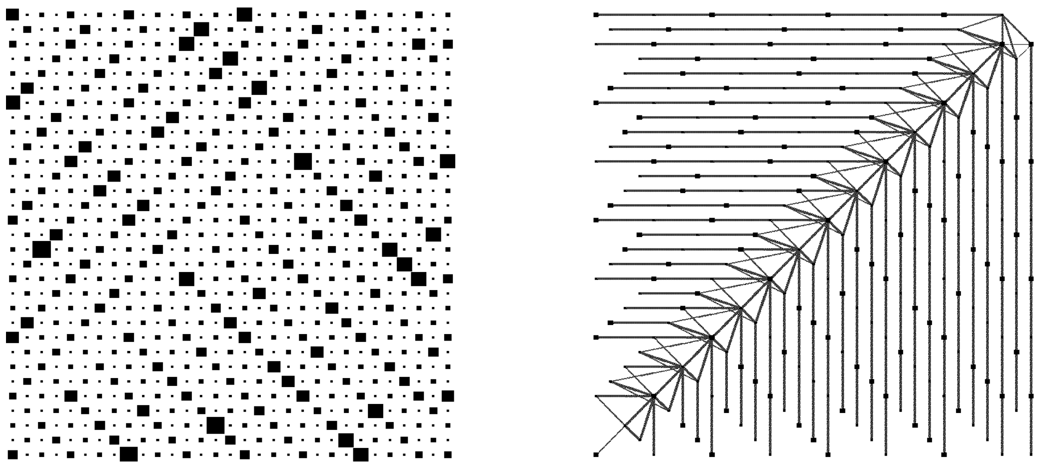


FIG. 14. *AMG-Hierarchy of grids for locally varying anisotropy, initial meshwidth $h = 1/32$, with 7 resulting coarse levels and graph of the coarse grid matrix after two coarsening steps.*

5.5. Multilevel preconditioner for convection diffusion problems. At last, we consider the convection diffusion operator

$$-\Delta + a(x_1, x_2)\frac{\delta}{\delta x_1} + b(x_1, x_2)\frac{\delta}{\delta x_2} \ . \tag{51}$$

We discretize it on the finest grid using finite differences. The 9-point stencil belonging to the discretization grid point (ih, jh) is then given by

$$\frac{1}{h^2}\begin{bmatrix} & -1-b^+ & \\ -1-a^+ & 4+\Sigma & -1-a^- \\ & -1-b^- & \end{bmatrix} , \tag{52}$$

where $a^{\pm} := h\max(\pm a((i \mp 1/2)h, jh), 0)$, $b^{\pm} := h\max(\pm b(ih, (j \mp 1/2)h), 0)$, $\Sigma := a^+ + a^- + b^+ + b^-$. The convection coefficients must be evaluated between the grid points to get stable discretizations. For the convection term we use here pure upwind discretization which results in a first order method only[8].

Now, the considered problem and the resulting matrices are *not symmetric* anymore. This renders any convergence theory following the lines of Xu [58, 59] or Oswald [35, 36, 37, 38] useless since it is only valid for symmetric operators. Note however that the framework given in section 2 can be applied to non-symmetric problems as well. Using the different coarsening methods we still can build up an enlarged, now non-symmetric system matrix belonging to a generating system[9] or alternatively a multilevel preconditioner (24). But now, the corresponding eigenvalues may be complex. Additionally, a conjugate gradient method is not working since the preconditioned system matrix is non-symmetric. Alternatively BiCGstab [50], CGsym [47], GMRES [46], or QMR [22] might be applied.

Here, we simply symmetrize our diagonally scaled enlarged linear system matrix (34) by multiplying it with its transposed. Thus we obtain

$$(\mathcal{B}_M^E)^{1/2}(\mathcal{P}_M^E)^T(\mathcal{A}_M)^T\mathcal{P}_M^E(\mathcal{B}_M^E)(\mathcal{P}_M^E)^T\mathcal{A}_M\mathcal{P}_M^E(\mathcal{B}_M^E)^{1/2} \tag{53}$$

and apply the conjugate gradient method to this system. Consequently, we study the square root of the generalized condition number of this matrix.

This approach is superior to the symmetrization of the basis version of the multilevel preconditioned matrix, $\mathcal{P}_M^E\mathcal{B}_M^E(\mathcal{P}_M^E)^T\mathcal{A}_M$, since even for symmetric matrices $\mathcal{A}_M$ the generalized condition number of the symmetrizised version would in general not give the square of the condition number of the preconditioned matrix.

[8] Using central differences instead results in a second order method but gives also stability problems for larger values of the convection coefficients. Note that in this case, for values of a, b resulting in a stable discretization for given h and also for discretizations using artificial viscosity or the Il'in scheme on the finest grid, the convergence properties of the different multilevel preconditioners were basically analogous to the results presented for the upwind discretization case.

[9] Especially for the non-symmetric case of the convection-diffusion equation, the Petrov-Galerkin approach makes sense as well, both with respect to the overall generating system matrix and with respect to the AMG coarse grid setup strategy itself.

In our first examples we have chosen the coefficient functions $a(x_1, x_2) \equiv a$ and $b(x_1, x_2) \equiv b$ to be constant over the whole domain Ω. Then, $\phi := \arctan(b/a)$ gives the angle between the convection and the x_1-axis and $\varepsilon := \sqrt{a^2 + b^2}$ gives its strength. We consider the cases $b = 0$ and $a = b$, i.e. $\phi = 0$ and $\phi = \pi/4$. The square root of the condition numbers for the enlarged matrices (53) associated to our three coarsening approaches are given in Tables 11 and 12. We see that the enlarged matrix (53) based on bilinear interpolation gives rising condition numbers for rising values of ε. Thus, it is not robust, especially not in the case $\phi = 0$. The same holds also for the Dendy approach. However the enlarged matrix using the AMG approach results in comparison to that in fairly good condition numbers. Whether an other choice of the AMG-parameters (α, β) might lead to further improvements is an open question yet.

TABLE 11

Square root of the condition numbers of (53) for the convection diffusion problem for $\phi = 0$, i.e. with convection in x_1-direction, finite difference upwind discretization (52) on the finest grid.

h^{-1}	$\varepsilon = 100$			$\varepsilon = 10^4$			$\varepsilon = \infty$		
	MDS	Dendy	AMG	MDS	Dendy	AMG	MDS	Dendy	AMG
8	14.9	14.0	6.80	22.4	19.7	9.00	22.5	17.9	2.80
16	26.1	22.0	9.71	88.7	59.5	13.2	90.6	50.6	3.58
32	29.5	21.9	11.8	336	152	21.9	365	130	4.41
64	29.7	20.0	15.2	1065	309	25.8	1468	318	5.25
128	29.5	22.0	18.4	2221	394	27.7	5878	747	6.08

TABLE 12

Square root of the condition numbers of (53) for the convection diffusion problem for $\phi = \pi/4$, i.e. with convection in diagonal direction, finite difference upwind discretization (52) on the finest grid.

h^{-1}	$\varepsilon = 100$			$\varepsilon = 10^4$			$\varepsilon = \infty$		
	MDS	Dendy	AMG	MDS	Dendy	AMG	MDS	Dendy	AMG
8	4.75	7.20	8.94	5.25	8.02	9.57	5.25	8.03	6.06
16	8.08	11.3	9.01	10.5	13.8	10.4	10.5	13.8	8.15
32	12.7	15.0	13.9	21.3	20.9	17.4	21.3	21.0	11.7
64	17.8	17.7	16.6	42.3	31.1	20.4	43.0	31.4	16.2
128	22.4	19.3	15.3	83.7	45.6	37.8	86.2	46.6	24.6

In Figure 15 we give the hierarchy of grids for the AMG approach with initial meshwidth $h = 1/32$, for the convection diffusion model problem and graphs of coarse grid matrices after two coarsening steps. We see, that in the case $\phi = 0$, the matrix on the coarser level mainly consists of the upwind stencil for the x_1-derivative, whereas in the case $\phi = \pi/4$ it consists of a sort of upwind stencil for $\frac{\partial}{\partial x_1} + \frac{\partial}{\partial x_2}$.

Finally we consider a convection diffusion problem with locally varying coefficient functions (see [44, 45]):

$$(54) \quad a(x_1, x_2) = \varepsilon \cdot 4x_1(x_1 - 1)(1 - 2x_2) \ , \quad b(x_1, x_2) = -\varepsilon \cdot 4x_2(x_2 - 1)(1 - 2x_1) \ .$$

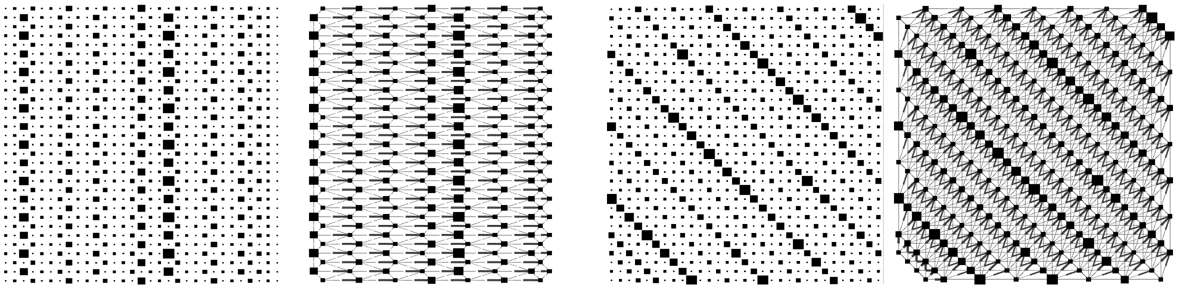


FIG. 15. *Left: AMG-Hierarchy of grids for the convection diffusion problem with $\phi = 0$, $\varepsilon = 10^4$, initial meshwidth $h = 1/32$, 6 resulting coarse levels and graph of the coarse grid matrix after two coarsening steps. Right: AMG-Hierarchy of grids for the convection diffusion problem with $\phi = \pi/4$, $\varepsilon = 10^4$, initial meshwidth $h = 1/32$, 6 resulting coarse levels and graph of the coarse grid matrix after two coarsening steps.*

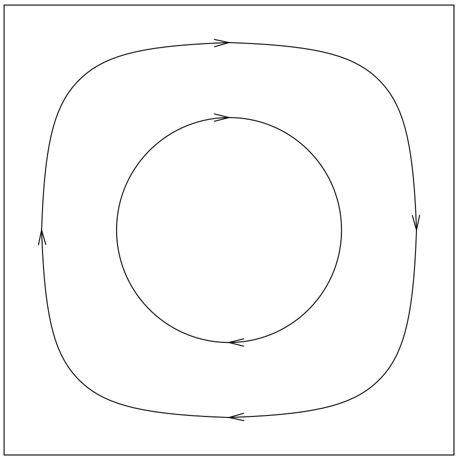


FIG. 16. *Direction of the convection for problem (54).*

Here, the direction of the convection is then almost circular around the point (1/2,1/2), see figure 16. The measured results are given in Table 13. We see that the MDS-based method has for larger values of ε really bad condition numbers. The values for the Dendy approach are somewhat better but far away from beeing robust. The AMG-based enlarged matrix (53), however, gives relatively good results[10]. Here, further work is necessary to improve the method.

Figure 17 shows the hierarchy of grids produced by AMG coarsening, and additionally the coarse grid matrix after three coarsening steps for the case $\varepsilon = 10^5$. Note that, here on the coarse grid, upwind stencils appear which follow the circular convection direction.

Finally, it is interesting to look at the basis functions that were developed due to the specific weighted interpolations induced by the Dendy and the AMG coarsening schemes. This is illustrated in Figure 18.

[10] Note that, for values of ε up to 10^{15} the AMG-type method gives really robust condition numbers with basically the same values as for $\varepsilon = 10^4$ and 10^6. However, for the exceptional case $\varepsilon = \infty$, where we use the pure upwind stencil, AMG gives bad results.

TABLE 13
Square root of the condition numbers of (53) for the circular convection problem (54) finite difference upwind discretization (52) on the finest grid.

h^{-1}	$\varepsilon = 100$			$\varepsilon = 10^4$			$\varepsilon = 10^6$		
	MDS	Dendy	AMG	MDS	Dendy	AMG	MDS	Dendy	AMG
8	8.76	10.2	6.74	17.5	34.9	8.09	17.7	36.0	8.11
16	12.1	13.7	10.8	63.6	136	13.4	67.3	124	16.5
32	13.3	14.6	10.9	221	347	14.5	273	438	14.8
64	13.2	14.2	13.1	577	780	29.5	1096	1320	36.4
128	12.8	14.6	18.6	984	475	35.5	4079	3444	41.1

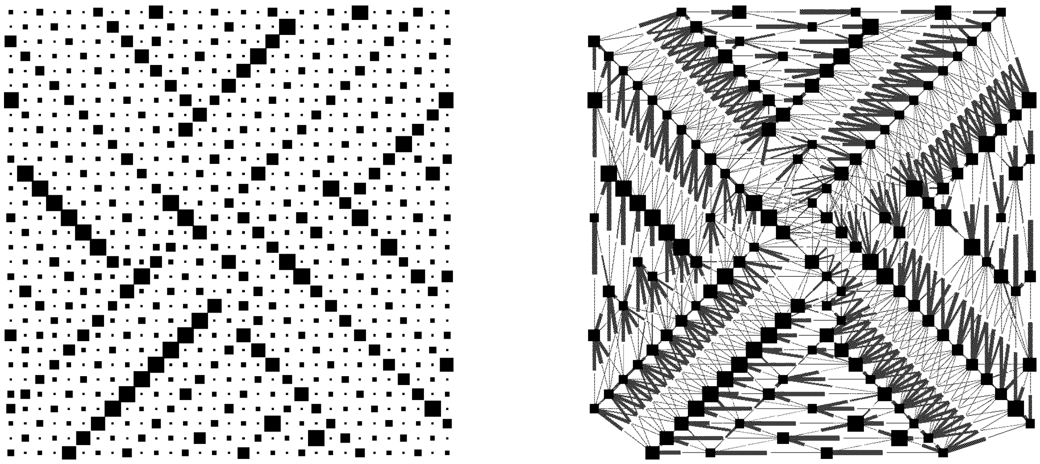


FIG. 17. *AMG-Hierarchy of grids for convection diffusion problem (54) with locally varying coefficients, initial meshwidth $h = 1/32$ and $\varepsilon = 10^5$, with 10 resulting coarse levels (the six finest levels are shown only) and graph of the coarse grid matrix after three coarsening steps.*

6. Concluding remarks. We considered the convergence behavior of three different additive multilevel preconditioners, first the BPX-type method based on bilinear interpolation, second, an algorithm involving an example of matrix-dependent geometric coarsening due to Dendy and, third, a preconditioner using the algebraic multigrid technique.

For 2nd order elliptic problems, the AMG-type preconditioner behaved favourably and resulted for most considered problems in relatively good and robust convergence rates. Since, for our implementation, the additional effort needed in its setup phase is about the same as 2-3 iterations of the plain BPX-method, which is not robust at all, we think that the AMG-type preconditioner is a reasonable method with respect to both, robustness and work involved. Interestingly, the properties with respect to robustness known from its multiplicative counterpart, i.e. the conventional AMG method, carries over to the additive preconditioner for the considered problem class. We believe that due to its additive construction principle, it will perform favourable when it comes to parallelization. Here, a geometric coarsening method with matrix-dependent interpolations needs additionally some ILU-type preconditioners on all levels, or in its multiplicative

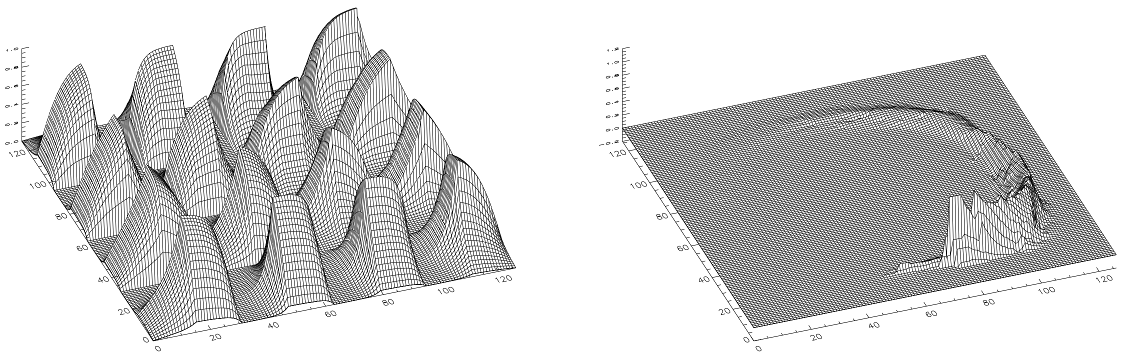


FIG. 18. *Left: A selection of coarse-level basis functions as they are produced by Dendy's approach for the convection diffusion problem (54) with* $\varepsilon = 100$, *initial meshwidth* $h = 1/128, M = 7$ *on level 3. Right: A single coarse-level basis function as it is produced by the AMG approach for the convection diffusion problem (54) with* $\varepsilon = 1000$, *initial meshwidth* $h = 1/128, M = 14$ *on level 4.*

variant ILU-type smoothers, to gain robustness for singular perturbed problems. This is more difficult to parallelize. But how good the setup phase needed in the AMG-based method parallelizes is an open question yet. Furthermore, it is unclear to us, whether the additive or the multiplicative variant of the AMG approach results in faster overall computing times.

Finally, we remark that for the AMG-based approach no general theory analogously to that in [10, 58, 59], [37, 38] or [4] exists, which shows both, h-independence of the condition number *and* its independence from the coefficient functions of the operator under consideration, i.e. robustness. This will be future work.

REFERENCES

[1] R. ALCOUFFE, A. BRANDT, J. DENDY, AND J. PAINTER, *Theoretical and practical aspects of a multigrid method*, SIAM J. Sci. Stat. Comput., 2 (1981), pp. 430–454.

[2] O. AXELSSON AND P. VASSILEVSKI, *Algebraic multilevel preconditioning methods*, Numerische Mathematik, 56 (1989), pp. 157–177.

[3] ——, *Algebraic multilevel preconditioning methods II*, Siam J. Numer. Anal., 27 (1990), pp. 1569–1590.

[4] F. BORNEMANN AND H. YSERENTANT, *A basic norm equivalence for the theory of multilevel methods*, ZIB, Berlin, Preprint 92-1, 1992.

[5] D. BRAESS, *The convergence rate of a multigrid method with Gauß-Seidel relaxation for the Poisson equation*, in Multigrid Methods, W. Hackbusch and U. Trottenberg, eds., Lecture Notes in Mathematics 960, Springer, Berlin, Heidelberg, New York, 1982.

[6] D. BRAESS, *Towards algebraic multigrid for elliptic problems of second order*, Tech. Report, Institut für Mathematik, Ruhr-Universität Bochum, 1994.

[7] ——, *Iterative methods for 3D groundwaterflow problems*, 1st GAMM Seminar Stuttgart On Modelling and Computation in Environmental Sciences, (1995).

[8] D. BRAESS AND W. HACKBUSCH, *A new convergence proof for the multigrid method including the V-cycle*, SIAM J. Numer. Anal., 20 (1983), pp. 967–975.

[9] J. BRAMBLE AND J. PASCIAK, *New estimates for multilevel algorithms including the V-cycle*, Report BNL-46730, Dept. Appl. Science, Brookhaven National Laboratories, 1991.

[10] J. Bramble, J. Pasciak, J. Wang, and J. Xu, *Convergence estimates for multigrid algorithms without regularity assumptions*, Math. Comp., 57 (1991), pp. 23–45.
[11] J. Bramble, J. Pasciak, and J. Xu, *Parallel multilevel preconditioners*, Math. Comp., 31 (1990), pp. 333–390.
[12] A. Brandt, *Guide to multigrid development*, in Multigrid Methods, W. Hackbusch and U. Trottenberg, eds., Lecture Notes in Mathematics 960, Springer, Berlin, Heidelberg, New York, 1982.
[13] A. Brandt, *Algebraic multigrid theory: The symmetric case*, in Preliminary Proceedings of the International Multigrid Conference, Copper Mountain, Colorado, April 6–8, 1983, S. McCormick and U. Trottenberg, eds., 1983.
[14] A. Brandt, S. McCormick, and J. Ruge, *Algebraic multigrid (AMG) for automatic algorithm design and problem solution*, report, Colorado State University, Ft. Collins, 1982.
[15] H. Bungartz, *Beschränkte Optimierung mit algebraischen Mehrgittermethoden*, Diplomarbeit, Institut für Informatik, Technische Universität München, 1988.
[16] Q. Chang and Y. S. Wong, *Recent developments in algebraic multigrid methods*, in Preliminary proceedings of the 2nd Copper Mountain Conference on Iterative Methods, Copper Mountain, April 9-14, 1992, T. Manteuffel and S. McCormick, eds., University of Colorado at Denver, 1992.
[17] ———, *A new approach for the algebraic multigrid method*, tech. report, Institute of Applied Mathematics, Academia Sinica, Beijing, China, 1994.
[18] W. Dahmen and A. Kunoth, *Multilevel preconditioning*, Numer. Math, 63 (1992), pp. 315–344.
[19] P. de Zeeuw, *Matrix-dependent prolongations and restrictions in a blackbox multigrid solver*, J. Comp. appl. Math., 33 (1990), pp. 1–27.
[20] J. Dendy, *Blackbox multigrid for nonsymmetric problems*, Appl. Math. Comput, (1983), pp. 261–283.
[21] M. Dryja and O. Widlund, *Multilevel additive methods for elliptic finite element problems*, in Parallel Algorithms for Partial Differential Equations, Proceedings of the Sixth GAMM-Seminar, Kiel, January 19-21, 1990, W. Hackbusch, ed., Vieweg-Verlag, 1991.
[22] R. Freund and Nachtigal, *QMR: A quasi-minimal residual method for non-hermitian linear systems*, Numerische Mathematik, 60 (1991), pp. 315–339.
[23] J. Fuhrmann, *Zur Verwendung von Mehrgitterverfahren bei der numerischen Behandlung elliptischer Differentialgleichungen mit variablen Koeffizienten*, Shaker, Aachen (Berichte aus der Mathematik, Techn. Univ. Chemnitz-Zwickau, Dissertation), 1995.
[24] M. Griebel, *Zur Lösung von Finite-Differenzen- und Finite-Element-Gleichungen mittels der Hierarchischen-Transformations-Mehrgitter-Methode*, TU München, Institut f. Informatik, TUM-I9007; SFB-Report 342/4/90 A, 1990.
[25] ———, *Multilevel algorithms considered as iterative methods on semidefinite systems*, SIAM J. Sci. Comput., 15 (1994), pp. 547–565.
[26] ———, *Multilevelmethoden als Iterationsverfahren über Erzeugendensystemen*, Teubner Skripten zur Numerik, (1994).
[27] ———, *Parallel domain-oriented multilevel methods*, SIAM J. Sci. Comput., 16 (1995), pp. 1105–1125.
[28] M. Griebel and P. Oswald, *On the abstract theory of additive and multiplicative Schwarz methods*, Numer. Math., 70 (1995), pp. 163–180.
[29] W. Hackbusch, *Convergence of multigrid iterations applied to differential equations*, Math. Comp., 34 (1980), pp. 425–440.
[30] W. Hackbusch, *Multigrid Methods and Applications*, Springer-Verlag, Berlin, Heidelberg, New York, 1985.
[31] S. Knapek, *Multiskalenverfahren bei der Modellierung, Diskretisierung und Lösung von Diffusionsproblemen*, Diplomarbeit, Institut für Informatik, Technische Universität München, 1995.
[32] Y. Kuznetsov, *Algebraic multigrid domain decomposition methods*, Sov. J. Numer. Anal. Math. Modelling, 4 (1989), pp. 351–380.
[33] R.D. Lonsdale, *An algebraic multigrid solver for the Navier-Stokes equations on unstructered*

meshes, Int. J. Num. Meth. Heat Fluid Flow 3 (1993), pp. 3-14
[34] P. Oswald, *Approximationsräume und Anwendungen in der Theorie der Multilevel-Verfahren*, Manuskript, FSU Jena, 1991.
[35] ——, *On discrete norm estimates related to multilevel preconditioners in the finite element method*, Proc. int. Conf. Constr. Theory of Functions, Varna, 1991.
[36] ——, *Two remarks on multilevel preconditioners*, Bericht Math/91/1, FSU Jena, Mathematische Fakultät, 1991.
[37] ——, *Norm equivalencies and multilevel Schwarz preconditioning for variational problems*, Bericht Math/92/1, FSU Jena, Mathematische Fakultät, 1992.
[38] ——, *Stable splittings of Sobolev spaces and fast solution of variational problems*, Bericht Math/92/5, FSU Jena, Mathematische Fakultät, 1992.
[39] ——, *On the robustness of the BPX-preconditioner with respect to jumps in the coefficients.* Manuskript, Texas AM University, 1995.
[40] M. Raw, *A coupled algebraic multigrid method for the 3D Navier-Stokes equations*, tech. report, Advanced Scientific Computing Ltd., 554 Parksite Drive, Unit 4, Waterloo Ontario, N2L 5Z4, Canada, 1994.
[41] H. Regler and U. Rüde, *Layout optimization with algebraic multigrid methods*, SFB Bericht 342/11/93 A/ I-9320, Institut für Informatik, TU München, August 1993.
[42] ——, *Layout optimization with algebraic multigrid methods (AMG)*, in Proceedings of the Sixth Copper Mountain Conference on Multigrid Methods, Copper Mountain, April 4-9, 1993, Conference Publication, NASA, 1993, pp. 497–512.
[43] A. Reusken, *Fourier analysis of a robust multigrid method for convection-diffusion equations*, GAMM Seminar Meisdorf, (1994). to appear in Computing.
[44] J. Ruge and K. Stüben, *Efficient solution of finite difference and finite element equations by algebraic multigrid (AMG)*, Arbeitspapiere der GMD, 89 (1984).
[45] J. Ruge and K. Stüben, *Algebraic multigrid (AMG)*, Arbeitspapiere der GMD, 210 (1986).
[46] Y. Saad and M. Schultz, *GMRES: A generalized minimal residual method for solving non-symmetric linear systems*, SIAM J. Sci. Stat. Comp., 7 (1986), pp. 856–869.
[47] P. Sonneveld, P. Wesseling, and P. de Zeeuw, *Multigrid and conjugate gradient methods as convergence acceleration techniques*, in Multigrid methods for integral and differential equations, D. Paddon and H. Holstein, eds., The Institute of Mathematics and its Applications, Conference Series, New Series 3, Clarendon Press, Oxford, 1985, pp. 117–167.
[48] A. van der Ploeg, *Preconditioning for sparse matrices with applications*, PhD thesis, Wiskunde, Rijksuniversiteit Groningen, 1994.
[49] A. van der Ploeg, E. Botta, and F. Wubs, *Grid-independent convergence based on preconditioning techniques*, in Multigrid Methods, W. Hackbusch and U. Trottenberg, eds., vol. IV, Lecture Notes in Mathematics, Springer, Berlin, Heidelberg, New York, 1993.
[50] H. Van Der Vorst, *BI-CGSTAB: A fast and smoothly converging variant of BI-CG for the solution of nonsymmetric linear systems*, SIAM J. Sci Stat. Com., 13 (1992), pp. 631–644.
[51] P. Vanek, J. Mandel, and M. Brezina, *Algebraic multigrid on unstructured meshes*, tech. report, Univ. Bolder, Colorado, 1995.
[52] C. Wagner, *Ein robustes Mehrgitterverfahren für Diffusions-Transport-Probleme in der Bodenphysik*, Preprint 93-70, Univ. Heidelberg, IWR, Dec. 1993.
[53] R. Webster, *An Algebraic Multigrid Solver for Navier-Stokes problems*, Int. J. Numer. Methods Fluids (1994), vol. 18, pp. 761-780
[54] P. Wesseling, *A robust and efficient multigrid method*, in Multigrid Methods, Lecture Notes in Mathematics, Springer-Verlag, 960 (1982).
[55] ——, *Theoretical and practical aspects of a multigrid method*, SIAM J. Sci. Stat. Comput., 3 (1982), pp. 387–402.
[56] G. Wittum, *Linear iterations as smoothers in multigrid methods: Theory with applications to incomplete decompositions*, Impact of Computing in Science and Engineering, 1 (1989), pp. 180–215.
[57] ——, *On the robustnes of ILU-smoothing*, SIAM J. Sci. Stat. Comput., 10 (1989), pp. 699–717.
[58] J. Xu, *Theory of multilevel methods*, Report No. AM48, Department of Mathematics, Pennsyl-

vania State University, 1989.
[59] ———, *Iterative methods by space decomposition and subspace correction: A unifying approach*, SIAM Review, 34 (1992), pp. 581–613.
[60] H. Yserentant, *Old and new convergence proofs for multigrid methods*, Acta Numerica, (1993).
[61] L. Y. Zaslavsky, *An adaptive algebraic multigrid for multigroup neutron diffusion reactor core calculations*, Appl. Math. Comput., 53 (1993), pp. 13–26.
[62] L. Y. Zaslavsky, *An adaptive algebraic multigrid for reactor criticality calculations*, SIAM J. Sci. Comp., 16 (1995).
[63] X. Zhang, *Multilevel Schwarz methods*, Numer. Math., 63 (1992), pp. 521–539.