GRID- AND POINT-ORIENTED MULTILEVEL ALGORITHMS

M. GRIEBEL *

Abstract. fehlt noch

Key words. partial differential equations, multilevel methods, multigrid methods, Gauss-Seidel iteration, semidefinite system.

AMS(MOS) subject classifications. 65F10, 65N20, 65N30

1. Introduction. Recently, see [4], a new concept for the development of multigrid and BPX-like multilevel algorithms for standard full grid problems had been presented. There, instead of a basis approach on the finest grid and the acceleration of the basic iteration by MG-coarse grid correction or a BPX-type preconditioner, a generating system was used to allow a non-unique level wise decomposed representation of the solution. The degrees of freedom are associated to the nodal basis functions of all levels under consideration. With this non-unique multilevel decomposed representation of a function, the Galerkin approach leads to a semidefinite linear system with unknowns on all levels. Its solution is non-unique but equivalent to the unique solution of the standard problem on the finest grid.

Furthermore, it was shown that traditional iterative methods for the semidefinite system are equivalent to modern elaborated multilevel methods which exhibit optimal convergence properties. The conjugate gradient method (with appropriate diagonal scaling) for the semidefinite system is equivalent to the BPX-conjugate gradient method for the fine grid system. Gauss-Seidel type iterations for the semidefinite system are equivalent to certain multigrid methods. For details see [4]. These methods are level-oriented and can be considered as level-block techniques. An outer iteration switches from level to level and an inner iteration operates on the specific grid.

Now, we consider the semidefinite system from a different point of view. We group all unknowns together which are associated to one grid point. This results in point-oriented methods and can be seen as point-block techniques. Now, an outer iteration switches from grid point to grid point. The local system that belongs to all basis functions of different levels centered in one grid point can be solved either directly or by an inner iteration that runs over all levels that are associated to the grid point under consideration. Furthermore, grid points can be grouped together to form subdomains. In this sense we get some sort of simple domain decomposition method which exhibits MG-like convergence properties.

In contrast to the parallelization of a multilevel method where communication has to take place on all levels to maintain good convergence rates, our point-block approach needs substantially less communication due to its domain decomposition qualities. In this sense, our new method is superior to other parallel multigrid and multilevel methods and we believe that its future will be bright.

* Institut für Informatik, Technische Universität München, Arcisstraße 21, D-8000 München 2

Note that our point-oriented methods are closely related to the frequency filtering ILU technique [12]. Instead of *sin*-functions for the filtering, we use the frequency componenents with respect to different mesh sizes, that belong to the same grid point.

We report the results of numerical experiments regarding the convergence rates of these new algorithms.

2. The semidefinite system. In this section, we introduce a generating system that replaces the usual finite element basis in the discretization of a boundary value problem of a partial differential equation. We then derive the associated semidefinite linear system and discuss its properties.

Consider a partial differential equation in d dimensions with a linear, second-order operator on the domain $\Omega = (0,1)^d$, d=1,2,...,

$$Lu = f \ on \ \Omega, \tag{1}$$

with appropriate boundary conditions and solution u. For reasons of simplicity, we restrict ourselves to homogeneous Dirichlet boundary conditions. Given an appropriate function space V, the problem can be expressed equivalently in its variational form: Find a function $u \in V$ with

$$a(u,v) = (f,v) \quad \forall \ v \in V. \tag{2}$$

(In the case of homogeneous Dirichlet boundary conditions, V would be the Sobolev space $H_0^1(\Omega)$.) Here, a: $V \times V \rightarrow V$ is a bounded, positive-definite, symmetric bilinear form and (.,.) is the usual linear form for the right-hand side. Let $||.||_a := \sqrt{a(.,.)}$ denote the induced energy norm. We assume that V is complete with respect to $||.||_a$, which is true if $a(.,.)$ is H_0^1-elliptic. The Lax-Milgram lemma then guarantees the existence and uniqueness of the solution of (2). If we consider directly the energy functional $1/2 \cdot a(u,u) - (f,u)$, the problem can be stated alternatively as minimization of the energy in V.

2.1. Spaces, bases, and the generating system. Assume we are given a sequence of uniform, equidistant, nested grids,

$$\Omega_1 \subset \Omega_2 \subset ... \subset \Omega_{k-1} \subset \Omega_k, \tag{3}$$

on Ω with respective mesh sizes h_l=2^{-l}, l=1,..,k, and an associated sequence of spaces of piecewise d-linear functions,

$$V_1 \subset V_2 \subset .... \subset V_{k-1} \subset V_k, \tag{4}$$

with dimensions

$$n_l := dim(V_l) = (2^l - 1)^d, l = 1, .., k. \tag{5}$$

Here, 1 denotes the coarsest and k the finest level of discretization. Consider also the sequence

$$N_1 \subset N_2 \subset ... \subset N_{k-1} \subset N_k \tag{6}$$

of sets of grid points N_l=$\{x_1, ..., x_{n_l}\}$ in the interior Ω_l, l=1,..,k.

The standard finite element basis which spans V_l on the equidistant grid Ω_l is denoted by B_l. It contains the nodal basis functions $\phi_i^{(l)}$, i=1,..,n_l, which are defined by

$$\phi_i^{(l)}(x_j) = \delta_{i,j}, \quad x_j \in N_l. \tag{7}$$

Note that we use rectangular grids. Therefore, the 2D basis functions can be written as the product of two 1D basis functions for the different coordinate directions.

Any function $u \in V_k$ can be expressed uniquely by

$$u = \sum_{\phi \in B_k} u_\phi \cdot \phi \tag{8}$$

with the vector u_k:=$(u_\phi)^T_{\phi \in B_k}$ of nodal values associated to some given ordering of the functions of B_k.

Following [14], a hierarchical basis can be constructed that takes the basis functions from B_l into account that are not contained in $B_{l-1}, l = k, .., 2$. With

$$\tilde{B}_l = \begin{cases} B_1 & \text{for } l = 1, \\ \{\phi_i^{(l)} \in B_l : x_i \notin N_{l-1}\} & \text{for } l > 1, \end{cases} \tag{9}$$

we obtain the hierarchical basis

$$H_k = \bigcup_{l=1}^{k} \tilde{B}_l. \tag{10}$$

Now, a function $u \in V_k$ can be expressed uniquely by

$$u = \sum_{\phi \in H_k} v_\phi \cdot \phi \tag{11}$$

with the vector v_k:=$(v_\phi)^T_{\phi \in H_k}$ of hierarchical values associated to some given ordering of the functions of H_k.

In contrast to these basis approaches expressed by the expansion in (8) and (11), we instead consider the set of functions E_k defined as the union of all the different nodal bases B_l for the levels l=1,..,k,

$$E_k = B_1 \cup B_2 \cup .... \cup B_{k-1} \cup B_k. \tag{12}$$

Clearly, because E_k is a linearly dependent set of functions, it is no longer a basis for V_k, but merely a generating system. See Figure 1 for a simple 1D example.

In any case, an arbitrary function $u \in V_k$ can be expressed in terms of the generating system by

$$u = \sum_{\phi \in E_k} w_\phi \cdot \phi \tag{13}$$

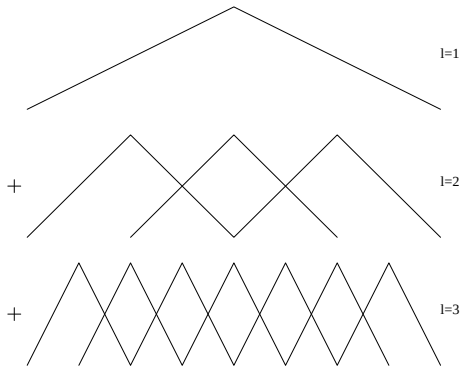

FIG. 1. *Functions of the generating system E_3 in the 1D case.*

with the vector $w_k^E := (w_\phi)_{\phi \in E_k}^T$ associated to some given ordering of the functions of E_k.

Here and in the following, we denote representations in terms of the generating system E_k by the superscript E. The length of w_k^E is

$$n_k^E = \sum_{l=1}^{k} n_l, \tag{14}$$

which is in the 1D case about twice, in the 2D case about 4/3 times, and in the 3D case about 8/7 times larger than the length of the vectors for the basis representation above. This is due to the geometric rate of decrease of the number of grid points from fine to coarse levels, with factors of 1/2, 1/4, and 1/8 for 1D, 2D, and 3D, respectively. The generalization of this concept to higher dimensions is straightforward.

Note that the representation of u in terms of E_k is not unique. In general, there exists an n_{k-1}^E-dimensional variety of level-wise decompositions of $u \in V_k$. However, for a given representation w_k^E of u in E_k, we can easily compute its unique representation u_k with respect to B_k. This involves the bilinear interpolation which can be expressed and implemented by MG-prolongation operators.

2.2. Galerkin approach and linear systems. Using the nodal basis B_k, the Galerkin-approach results in the discrete variational problem for $u \in V_k$

$$\forall \phi_i \in B_k : a(u, \phi_i) = f(\phi_i), \tag{15}$$

and the equivalent linear system of equations for the vector of nodal values

$$L_k u_k = f_k, \tag{16}$$

where

$$\begin{aligned} (L_k)_{i,j} &:= a(\phi_j^{(k)}, \phi_i^{(k)}), \qquad 1 \le i, j \le n_k \\ (f_k)_i &:= (f, \phi_i^{(k)}), \qquad 1 \le i \le n_k \end{aligned} \tag{17}$$

for some appropriate ordering of the functions of B_k.

For the generating system E_k, the Galerkin-approach leads to the variational problem

$$\forall \phi_i \in E_k : a(u, \phi_i) = f(\phi_i), \tag{18}$$

and, with representation (13), to the linear system

$$L_k^E w_k^E = f_k^E, \tag{19}$$

where

$$\begin{array}{rcll} (L_k^E)_{i,j} & := & a(\phi_j, \phi_i), & 1 \le i,j \le n_k^E \\ (f_k^E)_i & := & (f, \phi_i), & 1 \le i \le n_k^E \end{array} \tag{20}$$

for some appropriate ordering of the functions of E_k.

The system $L_k^E w_k^E = f_k^E$ has the following properties. The matrix L_k^E is semidefinite. It has the same rank as L_k. Thus $n_{k-1}^E = n_k^E - rank(L_k)$ eigenvalues are zero. The system is solvable because the right-hand side is constructed in a consistent manner. It has not just one unique solution, but rather numerous different solutions. However, the evaluation of two different solutions $w_k^{E,1}$ and $w_k^{E,2}$ with respect to their representation in B_k by means of MG-prolongation operators results in u_k which is the unique solution of the system $L_k u_k = f_k$. Therefore, it is sufficient to compute just one solution of the enlarged semidefinite system to obtain, via interpolation, the unique solution of the system $L_k u_k = f_k$.

Note that the enlarged matrix L_k^E contains the submatrices L_l that arise from the use of the standard basis B_l, $l = 1, .., k$. A similar property holds for the right-hand sides. The matrix and right hand side of the system

$$\hat{L}_k \hat{u}_k = \hat{f}_k \tag{21}$$

where

$$\begin{array}{rcll} (\hat{L}_k)_{i,j} & := & a(\phi_j, \phi_i), & \phi_i, \phi_j \in H_k \\ (\hat{f}_k)_i & := & (f, \phi_i), & \phi_i \in H_k, \end{array} \tag{22}$$

that stems from the hierarchical basis discretization is also contained in L_k^E.

3. Iterative Methods. Now, multilevel-type algorithms are easy to construct. In [4] it was shown that (after appropriate scaling) the conjugate gradient method for the semidefinite system $L_k^E w_k^E = f_k^E$ is equivalent to CG with the preconditioner of Bramble, Pasciak and Xu [2] for the system $L_k u_k = f_k$. The Gauss-Seidel-iteration for the semidefinite system is equivalent to the multigrid-method with Gauss-Seidel smoother. Here, depending on the ordering of the unknowns of the semidefinite systems, different MG-cycle strategies can be modeled easily. See the discussion in [4] for further guidance.

In the following we focus on Gauss-Seidel-type iterations for the semidefinite system only. First, we consider level-oriented Gauss-Seidel-iterations and demonstrate once

more that they are equivalent to multigrid and hierarchical basis multigrid methods. An outer iteration switches from level to level and an inner iteration operates on the respective grids. Then, we study block-oriented Gauss-Seidel-iterations. There, the unknowns of the semidefinite system that belong to basis functions centered at the same grid point are grouped together. Now, an outer iteration switches from grid point to grid point and an inner iteration or a direct solver works on the local subsystem of the semidefinite problem belonging to the respective grid point.

Note that, in both cases, it is not necessary to assemble the matrix L_k^E and the right hand side f_k^E explicitly. It is possible to use MG-prolongation and restriction operators and the fine grid discretizations L_k, f_k to express L_k^E and f_k^E in a certain product form. For an example for the level-oriented approach see (35) in [**?**].

Thus, the level- and point-oriented GS-methods can be implemented to need $O(n_k^E) = O(n_k)$ operations per iteration step only. Especially for the point-oriented block GS, this is quite tricky. A detailed description of implementation details will be given in a separate paper. The basic approach follows the idea presented in [15] and [3]. We use binary trees (of binary trees) to implement the fine grid. Altogether, the required storage and the number of operations to perform a GS- or block GS-step is proportional to the number of grid points employed.

3.1. Level-oriented Gauss-Seidel algorithms. We consider Gauss-Seidel iterations for the semidefinite system $L_k^E w_k^E = f_k^E$. As usual, we decompose the semidefinite matrix L_k^E by $L_k^E = F_k^E + G_k^E + (G_k^E)^T$, where F_k^E and G_k^E denote the diagonal and strictly lower triangular parts of L_k^E, respectively. Then, the Gauss-Seidel iteration (GS) is expressed by using $C_k^{E,GS} := (F_k^E + G_k^E)^{-1}$ and its symmetric counterpart SGS is expressed by using $C_k^{E,SGS} := (F_k^E + G_k^E)^{-T} F_k^E (F_k^E + G_k^E)^{-1}$ in the iteration

$$w_k^E := w_k^E + C_k^E (f_k^E - L_k^E w_k^E). \tag{23}$$

Note that $F_k^E + G_k^E$ is generally not symmetric but it is positive definite and invertible.

Of course, the lower triangular part of L_k^E depends on the ordering of the unknowns. For practical reasons, not all orderings are advisable. Here, we order the unknowns level-wise (starting for example with the finest level). This introduces a level-oriented block partition of L_k^E. The unknowns of each level can be ordered for example lexicographically or in a four color manner.

Note that this level-wise ordering corresponds to a level wise decomposition

$$V_k = \sum_{l=1}^{k} V_l = \sum_{l=1}^{k} \sum_{i=1}^{n_l} V_{l,x_i} \tag{24}$$

of V_k where $V_{l,x_i} = span\{\phi_i^{(l)}\}$.

In [**?**] it was shown that for a level wise ordering of the unknowns Gauss-Seidel iteration for the semidefinite system corresponds to the multigrid method for the standard system. The switching from level to level in the multigrid method corresponds to an outer (block) Gauss-Seidel iteration for the semidefinite system whereas the MG-smoothing steps resemble an inexact solver for each block by inner Gauss-Seidel itera-

tions. Especially the multigrid V-cycle with 1 pre- and post-smoothing step by Gauss-Seidel iterations deteriorates to the SGS method, which is also known as Aitken's double sweep. But other MG-cycle types can be modeled by different orderings of the block GS-traversal. The case of multiple smoothing steps corresponds to multiple inner iterations. Furthermore, other smoothers can be incorporated in the block GS-method as inner iterations.

These algorithms on the semidefinite system (19) can be interpreted alternatively as subspace correction methods [**?**]. The relaxation of an iterate $u^{(\kappa)} \in V_k$ then takes place with respect to a $\phi \in E_k$ by

$$u^{(\kappa+1)} := u^{(\kappa)} + \lambda \cdot \phi \quad \text{with} \quad a(u^{(\kappa+1)}, \phi) = f(\phi). \tag{25}$$

The different MG-type GS algorithms for the generating system involve the the cyclic application of (25) for a corresponding sequence of functions of E_k.

Note that the proofs of [**?**] and [**?**] directly carry over to the Gauss-Seidel iteration for the semidefinite system. There, it is shown that the MG-V cycle that corresponds now for our level-wise ordering to SGS has a an error propagation operator and thus a convergence rate independent of k. It behaves like

$$\rho = 1 - 1/C \qquad 1 < C < \infty. \tag{26}$$

If we iterate the unknowns of the semidefinite system that are associated to the functions of H_k only, i.g. if we use only the subset $H_k \subset E_k$ in (25) with some appropriate ordering, then we obtain the hierarchical basis multigrid method of Bank and Yserentant [1]. For practical reasons, the unknowns also have to be ordered level-wise. The error propagation operator of the HB-MG method and thus the convergence rate behaves in 2D like

$$\rho = 1 - C/k^2 \qquad 1 < C < \infty. \tag{27}$$

If we iterate only the unknowns of the semidefinite system that are associated to the functions contained in B_k, i.e. if we use only the (ordered) subset $B_k \subset E_k$ in (25), we get the standard Gauss-Seidel method for the fine grid system. It's convergence rate behaves like

$$\rho = 1 - C/4^k, \quad 1 < C < \infty. \tag{28}$$

Note that any iteration is convergent if the set of functions involved in one iteration sweep spans V_k. However, (**??**)-(28) indicate that we have to use a sufficiently large subset of E_k or even E_k itself to obtain a convergence rate independent of k.

3.2. Point-oriented Block Gauss-Seidel algorithms. Now we partition the unknowns of the semidefinite system into groups and perform a block-Gauss-Seidel iteration on the associated block-partitioned system. Thus, we decompose the semidefinite matrix L_k^E by $L_k^E = \mathcal{F}_k^E + \mathcal{G}_k^E + (\mathcal{G}_k^E)^T$, where $\mathcal{F}_k^E$ and $\mathcal{G}_k^E$ now denote the block diagonal and strictly lower block triangular parts of L_k^E, respectively. Then, the block

GS iteration is expressed by using $\mathcal{C}_k^{E,GS} := (\mathcal{F}_k^E + \mathcal{G}_k^E)^{-1}$ and its symmetric counterpart SGS is expressed by using $\mathcal{C}_k^{E,SGS} := (\mathcal{F}_k^E + \mathcal{G}_k^E)^{-T} \mathcal{F}_k^E (\mathcal{F}_k^E + \mathcal{G}_k^E)^{-1}$ in the iteration

$$w_k^E := w_k^E + \mathcal{C}_k^E (f_k^E - L_k^E w_k^E). \tag{29}$$

Note that $\mathcal{F}_k^E + \mathcal{G}_k^E$ is uniquely invertible only for certain choices of partitions, where each group contains unknowns associated to linear independent functions. Nevertheless, if this is not fulfilled, an iterative method still produces a nonunique solution for the corresponding semidefinite subsystem.

Alternatively, one block Gauss-Seidel step can be interpreted as subspace correction method. The relaxation of an iterate $u^{(\kappa)} \in V_k$ now takes place simultaneously with respect to a set $\Phi \subset E_k$ by

$$u^{(\kappa+1)} := u^{(\kappa)} + \sum_{\phi \in \Phi} \lambda_\phi \cdot \phi \quad \text{with} \quad \forall \phi \in \Phi : a(u^{(\kappa+1)}, \phi) = f(\phi). \tag{30}$$

For the computation of the values of $\lambda_\Phi = \{\lambda_\phi : \phi \in \Phi\}$ the system

$$L_\Phi \lambda_\Phi = r_\Phi \tag{31}$$

with

$$\begin{array}{rcll} (L_\Phi)_{i,j} & := & a(\phi_j, \phi_i), & \phi_i, \phi_j \in \Phi \\ (r_\Phi)_i & := & (f_\Phi)_i - a(u^{(\kappa)}, \phi_i) & \\ (f_\Phi)_i & := & (f, \phi_i), & \phi_i \in \Phi, \end{array} \tag{32}$$

has to be solved. The block Gauss-Seidel method for the semidefinite system is the cyclic application of (30) for a given sequence of (disjunct) subsets of $\subset E_k$. This type of iteration is only convergent if the union of all involved sets of functions span V_k.

Of course this approach is heavily dependent on the chosen sequence of subsets of E_k. In the following we suggest a point-oriented approach. This overcomes practical problems that occur with level-oriented multigrid methods if parallelization or adaptive local refinement is considered.

For the parallelization of a MG-algorithm, the domain is usually decomposed into blocks or stripes [?]. The grids covering these subdomains are assigned to different processors. To maintain the functionality of the smoother, data has to be exchanged between the subdomains and thus between the processors. This communication has to take place on all levels and contributes substantially to the overall cost of the parallel algorithm. In the case of adaptive refinement, 'holes' in the grids of certain levels can occur that might hinder efficient smoothing on the respective level.

These problems are overcome by the following approach. We group all unknowns together that are associated to the same grid point. This results in point-oriented methods and can be considered as a point-block technique. Now, an outer iteration switches from grid point to grid point. The local system that belongs to all basis functions of different levels centered in the same grid point can be solved either directly or by an inner iteration running over all unknowns associated to the grid point under

consideration. Furthermore, grid points can be grouped together to form subdomains. In this sense we get some sort of simple domain decomposition method which exhibits MG-type convergence properties.

To be more specific, we consider as blocks the unknowns associated to all functions

$$P_x := \{\phi \in E_k : x \in N_k, \phi(x) = 1\} \tag{33}$$

that are centered in the same grid point $x \in N_k$. This corresponds to a point-oriented decomposition

$$V_k = \sum_{x \in N_k} \sum_{l: x \in N_l} V_{l,x} \tag{34}$$

of the space V_k. Note that in comparison to the level-wise decomposition (24) just the summations are exchanged.

Now we can step through the set N_k of grid points and relax the unknowns that belong to the same grid point simultaneously. This results in systems of linear equations (31) with $\Phi = P_x$, $x \in N_k$, that form the block diagonal matrix $\mathcal{F}_k^E$ involved in (29) Note that the size of the systems belonging to $x \in N_l \setminus N_{l-1}, l = k, .., 1$, is $k - l + 1$. For the case of an operator with constant coefficients, the point block matrices L_{P_x} are full and symmetric Toeplitz matrices. It turns out that their condition number is $O(1)$. See also Table 3. Thus, the solution of (31) for $x \in N_l \setminus N_{l-1}$ can be obtained by some appropriate iterative method in $O((k - l + 1)^2)$ operations. The number of operations necessary for solving all arising point block subsystems is in 2D

$$C\frac{3}{4}n_k \left(1^\gamma + \frac{1}{4}2^\gamma + \frac{1}{16}3^\gamma + \frac{1}{64}4^\gamma + ...\frac{1}{4^{k+1-l}}(k+1-l)^\gamma + ... + k^\gamma\right) \tag{35}$$

where $\gamma = 2$ for SGS and $\gamma = 3$ for direct solution by Gaussian elimination. Altogether, this results in $O(n_k)$ operations. The coupling between two point blocks is described by the respective submatrix entries of $\mathcal{G}_k^E$, or in (32) by $a(u^\kappa, \phi_i), i = 1, .., |P_x|$, with a nonunique representation (13) of u^κ. It is easy to see that by this coupling information is exchanged on all respective levels of discretization simultaneously.

In practice, however, not all traversal orderings through the set of grid points N_k are advisable. We restrict ourselves to the sequence of point blocks, where the grid points of $N_k \setminus N_{k-1}$ are ordered first, for example in a three-color fashion. Second, the remaining grid points of N_{k-1} are ordered recursively in the same way. Then, neither the point block matrix nor the of diagonal blocks have to be assembled explicitly, but can be expressed by means of prolongation and restriction operators. However, the efficient solution of the diagonal block problems and the block Gauss-Seidel iteration is tricky to implement. It can be shown that one point block Gauss-Seidel step can be implemented to need $O(n_k)$ operations. Details are reported in [**?**].

The point-oriented approach can be easily generalized. We may allow an arbitrary domain decomposition of Ω into K nonoverlapping subdomains with associated decomposition

$$N_k = \bigcup_{i=1}^{K} N_k^i, \quad N_k^i \cap N_k^j = \emptyset \text{ for } i \neq j \tag{36}$$

of the grid points N_k. Now we group together the unknowns of the semidefinite system that are associated to functions of E_k whose center points are situated in the same N_k^i

$$P_{N_k^i} := \bigcup_{x \in N_k^i} P_x. \tag{37}$$

The resulting block Gauss-Seidel algorithm now switches from subdomain to subdomain in some prescribed order. For practical purposes, a nested dissection-like ordering [**?**] of the grid points into subdomains is advisable. Compare also Figure 3.2. Then, by using multigrid prolongation and restriction operators, the submatrices for the subdomains have not to be assembled explicitly and one overall block Gauss-Seidel iteration can be performed with $O(n_k)$ operations.

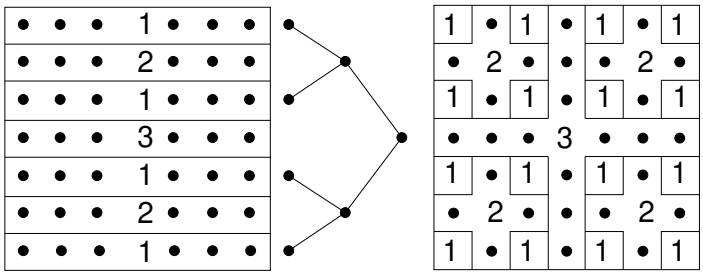

FIG. 2. *Stripe and box wise nested dissection decompositions.*

In contrast to the usual domain decomposition approach, where just the standard basis is used for discretization, the matrix of the subdomain problems is now in general not more invertible, since it is semidefinite. However, the application of an (level-oriented) GS-iteration automatically results in a multigrid like solver for the subdomain problem, which, due to the generating system approach, now produces a non unique representation of the solution for the subdomain. This non unique solution can be cast easily via interpolation operators into its unique representation with respect to B_k.

By using the generating system we have no longer to bother for the computation of the Schur complement. Instead, a semidefinite subsystem appears for each stripe-like subdomain with good $O(1)$ condition number. Furthermore, there is no more need to speed up convergence by means of global coarse grid transport steps or by using overlapping domains like in usual domain decomposition methods.

Now consider for example the stripe wise approach in figure 3.2 (left). All functions of E_k associated to the grid points of different stripes with the same number are mutually orthogonal with respect to $a(.,.)$, since they possess disjunct supports. Therefore, the stripe subproblems with the same numbers can be computed fully in parallel and we see directly a binary tree ordering ordering for the parallel execution of our algorithm. Communication has to take place only between father and son stripes, i.e. between next-neighbored stripes with successive number, and not on each level of discretization like in many parallel MG algorithms.

For the solution of the subproblem that belong to one stripe often an inexact solver (one or two SGS iterations which correspond to a MG-V-cycle with one or two pre-

and post-smoothing steps on the stripe only) is sufficient. In this sense, the point block GS method with analogous nested dissection ordering of the stripes (see figure 3.2) and additional level wise ordering of the grid points within the stripes is a direct realization of the domain-oriented method with inexact local solver. Thus, the point block method can be parallelized in the same way as the domain-oriented method. Note that with the special case of each N_k^i containing only one single $x \in N_k$, we obviously have the point-oriented method again.

4. Numerical experiments. We consider the simple model problem

$$\Delta u = f \ on \ \Omega = (0,1)^2 \tag{38}$$

with Dirichlet boundary conditions.

We measured the reduction rate ρ after a sufficient number of iterations by

$$\rho = \frac{||r_k^{\kappa+1}||_2}{||r_k^{\kappa}||_2} \tag{39}$$

with r_k^κ being the residual after κ iterations. Furthermore, we measured the number it of iterations necessary to reduce $|r_k^0|$ by a factor 10^{-10}.

First, we consider the level wise Gauss-Seidel method for $L_k^E w_k^E = f_k^E$ with such orderings that a MG-V cycle with v_1 pre- and $v2$ post-smoothing steps is realized. The unknowns on each level are ordered in a four color manner. The results of our experiments are shown in Table 1. Of course, as the level-oriented GS iteration for the semidefinite system is equivalent to MG, these reduction rates are reported in many papers already. We list it here only for comparison with the point-oriented techniques.

TABLE 1
Convergence rates and iteration steps for the level-wise Gauss-Seidel iterating on E_k (MG).

$v1$	$v2$	$k =$	3	4	5	6	7	8	9	10
1	0	ρ	0.11	0.11	0.15	0.18	0.20	0.23	0.26	
		it								
0	1	ρ	0.105	0.106	0.107	0.113	0.122	0.127	0.129	
		it								
1	1	ρ	0.020	0.031	0.038	0.043	0.045	0.046	0.046	
		it								
2	1	ρ	0.004	0.006	0.016	0.021	0.023	0.024	0.024	
		it								
1	2	ρ	0.004	0.004	0.014	0.019	0.021	0.022	0.022	

Now consider the GS iteration for the semidefinite system that relaxes only the unknowns belonging to the functions of H_k. Thus, we obtain the HB-MG method. Here, we traverse the unknowns in the ordering $(N_k \setminus N_{k-1}, N_{k-1} \setminus N_{k-2}, .., N_1)$. We relax the unknowns of $N_l \setminus N_{l-1}$ v_1 times, where each set $N_l \setminus N_{l-1}$ v_1 is ordered itself in a three color fashion. Then, we traverse the unknowns in inversed order and relax each set $N_l \setminus N_{l-1}$ v_2 times. The observed convergence factors and number of iteration steps are shown in Table 2. The dependency (27) of the convergence factor from the number k of levels can be seen clearly.

TABLE 2
Convergence rates and iteration steps for the level-wise Gauss-Seidel iterating on H_k (HB-MG).

$v1$	$v2$	$k=$	3	4	5	6	7	8	9	10
1	0	ρ	0.549	0.660	0.737	0.794	0.833	0.863		
		it								
0	1	ρ	0.537	0.648	0.729	0.787	0.827	0.858		
		it								
1	1	ρ	0.445	0.571	0.659	0.724	0.773	0.810		
		it								
2	1	ρ	0.424	0.553	0.642	0.710	0.758	0.800		
		it								
1	2	ρ	0.409	0.538	0.633	0.696	0.749	0.788		

In contrast to that, the point-block method relaxes *all* unknowns simultaneously that belong to the functions of E_k centered at the same grid point. In comparison to the HB-MG method, where only the unknown is relaxed that belongs to the function with the largest support, now all unknowns with associated functions that are centered in the grid point under consideration are relaxed simultaneously. In this sense, the point block method can be seen as an extension of the hierarchical basis MG method that overcomes its disadvantage with respect to the convergence behavior. In contrast to standard MG, all algorithmical advantages of HB-MG remain, especially with respect to parallelization and adaptivity.

In Table 3 we show the condition numbers of the point block matrix belonging to one single grid point (here the point (0.5,0.5)) for different k. Furthermore the spectral radius of the symmetric Gauss-Seidel iteration relaxing on the point-block system is shown. We see that the point block subsystems can be solved iteratively by a number

TABLE 3
Condition number and SGS spectral radius for the submatrix belonging to one point block.

$k=$	5	10	15	20	30	40	50	100	200	300
κ	11.735	15.612	16.873	17.429	17.889	18.070	18.161	18.287	18.321	18.326
ρ	0.4365	0.4426	0.4436	0.4440	0.4442	0.4443	0.4444	0.4444	0.4444	0.4444

of SGS-steps independent of k. SGS converges at least with a convergence factor of 0.4444.

The reduction rates and iterations steps for the point block GS method is shown in Table 4. We clearly see that the reduction rates and the number of point block GS iterations is independent of k.

TABLE 4
Convergence rates and iteration steps for the point block Gauss-Seidel.

$v1$	$v2$	$k=$	3	4	5	6	7	8	9	10
1	0	ρ	0.132	0.136	0.137	0.145	0.154	0.156		
		it								
0	1	ρ	0.125	0.133	0.135	0.136				
		it								
1	1	ρ	0.035	0.045	0.051	0.055	0.057			
		it								

Now consider the domain-oriented block GS iteration with stripe-like decomposition and nested dissection ordering like in Figure 3.2. The observed convergence factors

and numbers of iterations are shown in Table 5. We see that the obtained rates are independent of k. However, in comparison to the results of Table 4, no significant improvement can be observed. Thus, with respect to the point-oriented block GS method, the exact solution of the subproblems on the stripes is not necessary and less efficient. It can be substituted by one or 2 SGS steps working on the stripe subdomains.

TABLE 5
Convergence rates and iteration steps for the stripe wise domain decomposition block Gauss-Seidel.

$v1$	$v2$	$k=$	3	4	5	6	7	8	9	10
1	0	ρ	0.076	0.151	0.188	0.211	0.225			
		it								
0	1	ρ	0.077	0.152	0.184	0.206				
		it								
1	1	ρ	0.053	0.113	0.141	0.165	0.174			
		it								

5. Concluding remarks. In this paper we presented different multilevel algorithms based on the generating system approach. We studied level-oriented techniques where GS methods for the arising semidefinite system turn out to result either in conventional MG algorithms or in the HB-MG method. Additionally, we presented new point- (and domain-) oriented methods. There, block GS iterations for the semidefinite system are obtained that exhibit a convergence rate independent of the grid size and number of levels like conventional MG methods. Furthermore, these algorithms possess favorable properties with respect to parallelization and adaptivity. In contrast to the parallelization of a MG method, where communication has to take place on every level of discretization, our new algorithms have communication requirements like conventional domain decomposition methods. Thus, we demonstrated that the use of the generating system is an excellent guideline for the development of new multilevel algorithms.

The theory for proving k-independent convergence rates for our algorithms has to follow the lines of Oswald [10], [11], Xu [13] and Zhang [?]. For the level-oriented approach, existing prooves [13], [?] can be rewritten directly in terms of the generating system. For the point-oriented approach this will be done in near future.

Note that the point- and domain-oriented approach is closely related to the frequency filtering ILU technique [12] but show a stronger grid oriented flavor.

In [7], our level- and point-oriented approach is adopted to an extended generating system that contains additionally the basis functions of all grids that result from semi-refinement steps with respect to both coordinate directions. Furthermore, the full and the sparse grid case [?] is considered. For the level-oriented methods, we obtain MG algorithms similar to that of Hackbusch [5], [6], Naik and van Rosendale [9], Moulder [8] and Wesseling [?]. For the point-oriented method we observe a substantial improvement of the error reduction properties for the resulting algorithm in comparison to the case using E_k only. We obtain a really fast algorithm with a convergence rate of about 0.02. For details, see [7].

[1] R. Bank, T. Dupont, and H. Yserentant, *The hierarchical basis multigrid method*, Num. Math., 52 (1988), pp. 427–458.
[2] J. Bramble, J. Pasciak, and J. Xu, *Parallel multilevel preconditioners*, Math. Comp., 31 (1990), pp. 333–390.
[3] H.-J. Bungartz, *Dünne Gitter und deren Anwendung bei der Lösung der dreidimensionalen Poisson-Gleichung.*, Promotion.
[4] M. Griebel, *Multilevel algorithms considered as iterative methods on indefinite systems*, TU München, Institut f. Informatik, TUM-I9143; SFB-Report 342/29/91 A, 1991.
[5] W. Hackbusch, *A new approach to robust multi-grid solvers*, Bericht, Inst. f. Informatik und praktische Mathematik, CAU, Kiel, 1987.
[6] ——, *The frequency decomposition multi-grid method, part I: applications to anisotropic equations*, Numerische Mathematik, Vol.—, 1989.
[7] C. Z. M. Griebel and S. Zimmer, *Improved multilevel algorithms for full and sparse grid problems*, TU München, Institut f. Informatik, SFB-Report, 1992.
[8] V. Moulder, *titel*, in Journal.
[9] N. H. Naik and J. van Rosendale, *The improver robustness of multigrid elliptic solvers based on multiple semicoarsened grids*, in Journal.
[10] P. Oswald, *Two remarks on multilevel preconditioners*, FSU Jena, Mathematische Fakultät, Bericht Nr. Math/91/1, 1991.
[11] ——, *Norm equivalencies and multilevel Schwarz preconditioning for variat ional problems*, FSU Jena, Mathematische Fakultät, Bericht Nr. Math/92/1, 1992.
[12] G. Wittum, *Filternde Zerlegungen, schnelle Löser für große Gleichumngssysteme*, Teubner Scripten zur Numerik, Hrsg. H. Bock, W. Hackbusch, R. Rannacher, 1992.
[13] J. XU, *Iterative Methods by Space Decomposition and Subspace Correction: A Unifying Approach*, Report No. AM67, Department of Mathematics, Penn State Universit y, revised 1992.
[14] H. Yserentant, *On the multi-level splitting of finite element spaces*, Numer. Math., 49 (1986), pp. 379–412.
[15] C. Zenger, *Sparse Grids*, in Parallel Algorithms for Partial Differential Equations, Proceedings of the Sixth GAMM-Seminar, Kiel, January 19-21, 1990, W. Hackbusch, ed., Vieweg-Verlag, 1991.