Konrad-Zuse-Zentrum
für Informationstechnik Berlin

Takustraße 7
D-14195 Berlin-Dahlem
Germany

RUDOLF BECK

Algebraic Multigrid by Component Splitting for Edge Elements on Simplicial Triangulations

Preprint SC 99-40 (December 1999)

Algebraic Multigrid by Component Splitting for Edge Elements on Simplicial Triangulations

Rudolf Beck

Abstract

Our focus is on Maxwell's equations in the low frequency range; two specific applications we aim at are time-stepping schemes for eddy current computations and the stationary double-curl equation for time-harmonic fields. We assume that the computational domain is discretized by triangles or tetrahedrons; for the finite element approximation we choose Nédélec's $H(curl)$-conforming edge elements of the lowest order.

For the solution of the arising linear equation systems we devise an algebraic multigrid preconditioner based on a spatial component splitting of the field. Mesh coarsening takes place in an auxiliary subspace, which is constructed with the aid of a nodal vector basis. Within this subspace coarse grids are created by exploiting the matrix graphs. Additionally, we have to cope with the kernel of the $curl$-operator, which comprises a considerable part of the spectral modes on the grid. Fortunately, the kernel modes are accessible via a discrete Helmholtz decomposition of the fields; they are smoothed by additional algebraic multigrid cycles.

Numerical experiments are included in order to assess the efficacy of the proposed algorithms.

Keywords: Algebraic multigrid, mesh coarsening, edge elements, Nédélec spaces, Maxwell's equations

AMS(MOS) subject classifications: 65N55, 65N30, 65F10, 35Q60

1 Introduction

Meanwhile edge elements, sometimes also called Whitney-1-forms or Nédélec's $H(curl)$-conforming elements of the lowest order, have gained widespread popularity in numerical electromagnetic field computations by finite element methods [2, 7, 14, 21]. There are two specific virtues speaking in favor of these elements: their capabilities in suppressing non-physical modes and in modeling fields consistently at re-entrant corners [6, 8].

Though all macroscopic electromagnetic phenomena are governed by Maxwell's equations and some additional material laws, there is a wide range of electromagnetic problem types which are tackled by specific theoretical and numerical approaches (see, eg., [4, 19, 20]).

In this paper we confine ourselves to fields varying on a slow time scale. By this we mean that the wavelengths of the fields are not substantially smaller than the diameter of the computational domain; a typical example is the calculation of eddy currents. In general we have to distinguish between transient and time-harmonic field computations; in the latter case a stationary boundary value problem for one fixed frequency has to be solved. If we tackle the transient case by an implicit time-stepping method – which is advisable in order to avoid small time steps dictated by the CFL-restriction – we arrive at a boundary value problem in each time step.

Thus for both types of field simulations the fast and robust solution of linear equation systems may become an important task. For very large systems usually iterative solvers are the method of choice. Their convergence behaviour may be substantially enhanced by multigrid techniques if a suitable sequence of nested triangulations and finite element spaces is available; we speak of *geometric multigrid* in this case [16, 27].

However, the prerequisites of this technique are often not met in practice. If, for example, we have to run our simulation on a large irregular mesh provided by some grid generation tool, it is desirable to construct coarser finite element spaces in a recursive manner. Such procedures usually work with the matrix of the equation system and have therefore been summarized under the name *algebraic multigrid* (in the sequel abbreviated by AMG).

The first research efforts in this field started more than ten years ago [11, 12, 24] and meanwhile a variety of numerical problems has been tackled. The resulting algorithms are often sophisticated and based on different techniques like matrix-weighted interpolation methods, incomplete LU-factorization, or aggregation [1, 9, 13, 15, 23, 25, 29]. For an overview the reader is referred to [26].

When dealing with edge elements in electrodynamical problems, the situation is rather delicate. The matrices generated here are not diagonally dominant and lacking the M-matrix-property required in some established AMG-algorithms. Furthermore, a naive application of aggregation techniques does not make sense: If fine grid basis functions are aggregated on a certain area, the resulting coarse grid function represents a vector field with fixed direction on this part of the domain and thus need not allow an adequate approximation of the solution. To circumvent this problem appears to be a tricky task without taking into account a substantial amount of geometric information.

An additional difficulty is posed by the nullspace of the $curl$-operator, which typically governs dynamical electromagnetic phenomena. This nullspace comprises a considerable part of all spectral modes on the finite element grids and impedes smoothing iterations like Gauss-Seidel relaxation, which play an essential role in multigrid algorithms, but cannot cope with small eigenvalues.

In this paper we present an algebraic multigrid scheme which may be used as a preconditioner for a suitable Krylov-subspace solver like conjugate gradients or residuals. It is based on separate V-cycles in the nullspace of $curl$ and in the Nédélec space by exploiting discrete potentials via a Helmholtz-decomposition. Similar strategies for geometric multigrid have been proposed in [4, 5, 17]. The space of potentials is a subspace of the Nédélec space and spanned by linear Lagrange-type basis functions. Within this subspace a Poisson-problem has to be solved, thus an efficient application of algebraic mesh coarsening is not difficult here. As for the Nédélec space, coarsening will be carried out with the help of an auxiliary basis, whose construction is induced by linear Lagrange-type vector-elements. In both cases we apply an algebraic multigrid algorithm, which utilizes merely the graphs defined by the system matrices. It is vertex-based and selects coarse grid vertices by an advancing-front method.

The organization of the paper is as follows: In section 2 we are going to give a brief description of the problem types in question, the finite element spaces and their decomposition will be addressed in section 3. The following two sections are devoted to algebraic coarsening and the resulting multigrid preconditioner. In section 6 some numerical examples are included.

2 Variational Formulations

In this section we are going to state the problem types suitable for our algebraic multigrid procedures. We shall omit all non-relevant details, for a more comprehensive description the reader is referred to [4]. We confine ourselves to electromagnetic phenomena in linear materials where no free charges are present. Here Maxwell's equations

$$curl\boldsymbol{H} = \quad \boldsymbol{j} \ + \epsilon\, \partial_t \boldsymbol{E}\ , \tag{1}$$

$$curl\boldsymbol{E} = -\mu\, \partial_t \boldsymbol{H} \tag{2}$$

are linking the electric field $\boldsymbol{E} = \boldsymbol{E}(\boldsymbol{x}, t)$, the magnetic field $\boldsymbol{H} = \boldsymbol{H}(\boldsymbol{x}, t)$ and the electric current density $\boldsymbol{j} = \boldsymbol{j}(\boldsymbol{x}, t)$ on a computational domain Ω, $\Omega \subset \mathbb{R}^2$ or $\Omega \subset \mathbb{R}^3$. Both the permittivity ϵ and the permeability μ are bounded, possibly discontinuous functions with $\epsilon > 0$ and $\mu > 0$. Additionally we introduce the conductivity σ and assume that Ohm's law $\boldsymbol{j} = \sigma \boldsymbol{E}$ holds in conducting regions.

As boundary conditions will not be of any interest for our algorithms, we shall always assume that "appropriate boundary conditions" are stated in a problem definition. Hence all the respective boundary integrals will be omitted in the arising variational formulations.

The appropriate function space for the fields $\boldsymbol{E}$ and $\boldsymbol{H}$ naturally is provided by

$$H(curl; \Omega) := \{\boldsymbol{v} \in \boldsymbol{L}^2(\Omega)\, ;\ curl\boldsymbol{v} \in \boldsymbol{L}^2(\Omega)\}\ .$$

In the sequel we shall write $(\cdot\,,\cdot)$ for the usual $\boldsymbol{L}^2(\Omega)$ inner product.

2.1 Eddy current computations

If we are dealing with transient processes on a slow time scale featuring substantial dissipation, the current density $\boldsymbol{j}$ becomes the dominant term on the right hand side in (1) and we may neglect the displacement current $\epsilon\,\partial_t\boldsymbol{E}$. After taking the derivative of (1) with respect to time and inserting $\partial_t\boldsymbol{H}$ from eq. (2) we get

$$curl(\frac{1}{\mu}curl\boldsymbol{E}) = \sigma\,\partial_t\boldsymbol{E}\ .$$

Thus a parabolic problem emerges, where implicit schemes are the methods of choice for time-stepping. If, for example, we apply the backward Euler method, switch to the weak form and use Green's formula, we obtain the following variational problem for the n-th time step:

Seek $\boldsymbol{E}_n \in H(curl;\Omega)$ such that for all $\boldsymbol{v} \in H(curl;\Omega)$

$$(\sigma\boldsymbol{E}_n, \boldsymbol{v}) + \Delta t_n\,(\frac{1}{\mu}curl\boldsymbol{E}_n,\, curl\boldsymbol{v}) = \Delta t_n\,(\sigma\boldsymbol{E}_{n-1}, \boldsymbol{v})\ . \tag{3}$$

Here Δt_n denotes the length of the time step; for the sake of simplicity we have dropped the boundary integral arising in the partial integration formula.

It is obvious that on insulating regions ($\sigma = 0$) eq. (3) does not have a unique solution. In this case additional means must be taken into account in order to enforce the uniqueness of $\boldsymbol{E}_n$, eg. by imposing the continuity of flux normals on element faces by a penalty term. Thus a (weighted) zero-divergence constraint for the flux density $\epsilon\boldsymbol{E}_n$ is applied and we get a symmetric, positive definite system matrix on discretization.

2.2 Time-harmonic problems

Here we consider fields varying sinusoidally in time with a fixed angular frequency ω:

$$\boldsymbol{E}(\boldsymbol{x},t) = \mathrm{Re}\big\{\widehat{\boldsymbol{E}}(\boldsymbol{x})e^{i\omega t}\big\}\ ,\ \boldsymbol{H}(\boldsymbol{x},t) = \mathrm{Re}\big\{\widehat{\boldsymbol{H}}(\boldsymbol{x})e^{i\omega t}\big\}\ ,\ \boldsymbol{j}(\boldsymbol{x},t) = \mathrm{Re}\big\{\widehat{\boldsymbol{j}}(\boldsymbol{x})e^{i\omega t}\big\}\ ,$$

where $\widehat{\boldsymbol{E}}$, $\widehat{\boldsymbol{H}}$, and $\widehat{\boldsymbol{j}}$ denote complex amplitudes. From (1) and (2) we can derive the so-called double-curl equation for the electric field $\widehat{\boldsymbol{E}}$:

$$curl(\frac{1}{\mu}curl\widehat{\boldsymbol{E}})\ -\ \omega^2\,\widehat{\epsilon}\,\widehat{\boldsymbol{E}}\ =\ 0\ . \tag{4}$$

Observe that $\widehat{\boldsymbol{j}}$ has been dropped by taking into account Ohm's law $\widehat{\boldsymbol{j}} = \sigma\widehat{\boldsymbol{E}}$ and introducing the complex coefficient $\widehat{\epsilon} = \epsilon - i\sigma/\omega$. Neglecting boundary integrals again, eq. (4) may be cast into the following weak problem:

Seek $\widehat{\boldsymbol{E}} \in H(curl;\Omega)$ such that for all $\widehat{\boldsymbol{v}} \in H(curl;\Omega)$

$$(\frac{1}{\mu}curl\widehat{\boldsymbol{E}},\, curl\widehat{\boldsymbol{v}})\ -\ \omega^2(\widehat{\epsilon}\,\widehat{\boldsymbol{E}}, \widehat{\boldsymbol{v}})\ =\ 0\ . \tag{5}$$

In the sequel we shall mainly deal with the time-harmonic case (5). In contrast to (3) it will give rise to complex indefinite problems.

3 Helmholtz Decomposition of Finite Element Spaces

It is well-known that edge elements yield a conforming discretization of the space $H(curl;\Omega)$. We shall employ them on triangular or tetrahedral grids $\mathcal{T}_h$. In the following the symbol N_h will – as an abbreviation of the term "Nédélec space" – denote the finite element space associated with $\mathcal{T}_h$.

After discretizing our domain Ω with edge elements, the variational formulation (5) yields a sparse linear equation system with a complex symmetric (but not hermitian) matrix A_N. The kernel of the $curl$-operator comprises a considerable number of the eigenmodes of A_N, which are shifted to eigenvalues with negative real part by the second term in (5). This also holds for modes of the kernel's orthogonal complement; the number of these modes increases with ω.

The consequences become apparent on considering iterative and especially multigrid solvers for such linear systems: Eigenmodes belonging to small or even negative eigenvalues are impervious to standard smoothers like the Gauss-Seidel iteration. But effective smoothing iterations are a vital ingredient in multigrid cycles.

A (partial) remedy is offered via the discrete Helmholtz-decomposition of a vector field $\boldsymbol{E}_h \in N_h$; its exploitation by "hybrid smoothing" was originally suggested in [17]. We know that in the continuous space irrotational fields can be represented as the gradients of potentials. It is a specific feature of edge elements that they do reflect this property in a neat manner: the irrotational part $\boldsymbol{E}_{p,h}$ of $\boldsymbol{E}_h$ is the gradient of a discrete potential ϕ_h, with ϕ_h lying in the space of piecewise linear continuous finite element functions S_h on $\mathcal{T}_h$ (see, eg., [18]). Thus for any $\boldsymbol{E}_h \in N_h$ we obtain a decomposition

$$\boldsymbol{E}_h = \boldsymbol{E}_{p,h} + \boldsymbol{E}_{s,h}\,, \quad \boldsymbol{E}_{p,h} = grad\,\phi_h \quad \text{with } \phi_h \in S_h\,. \tag{6}$$

Here $\boldsymbol{E}_{s,h}$ denotes the solenoidal part of $\boldsymbol{E}_h$, lying in the orthogonal complement of the kernel (for a more comprehensive description we refer to [4, 5, 17]). However, we need a numerically efficient mechanism for transfers between potential representations in S_h and fields in the Nédélec space N_h.

A suitable transfer operator can be found by considering the definition of global degrees of freedoms in N_h. For a vector field $\boldsymbol{E}$ the degree of freedom $\boldsymbol{E}_j$ associated with the edge e_j of a triangulation $\mathcal{T}_h$ is given by the path integral

$$\boldsymbol{E}_j = \int_{e_j} \boldsymbol{E}\cdot \mathtt{t}_j\, ds \tag{7}$$

along the edge e_j with (oriented) tangent vector $\mathtt{t}_j$ [21]. Thus, if we use (6) for expressing the irrotational part $\boldsymbol{E}_{p,h}$ of $\boldsymbol{E}_h$ as the gradient of a potential ϕ_h, its representation in N_h can be obtained easily by utilizing (7):

$$\boldsymbol{E}_j = \int_{\boldsymbol{x}_1}^{\boldsymbol{x}_2} grad\,\phi_h \cdot \mathtt{t}_j\, ds \;=\; \phi_h(\boldsymbol{x}_2) - \phi_h(\boldsymbol{x}_1)\,. \tag{8}$$

Here $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ denote the coordinates of the vertices connected by e_j. By (8) we have obtained a transfer or prolongation operator P_{S_h}: $S_h \to N_h$, which may be evaluated by simple local computations. In the sequel $P_{S_h}^T$ will denote the adjoint operator, defining the canonical restriction.

We are now able to construct a symmetric preconditioner based on the described decomposition of N_h. As we are working with nested spaces, the system matrix A_ϕ of the potential space S_h will be computed by the Galerkin product $A_\phi = P_{S_h}^T A_N P_{S_h}$. The preconditioner is intended to embedded into a Krylov-subspace iteration like conjugate gradients or conjugate residuals. In the subsequent algorithms coefficient vectors and matrices will be tagged with N if belonging to N_h; ϕ will be used to tag those of S_h. For brevity we shall use "GS" as an abbreviation of "Gauss-Seidel" in the sequel.

Algorithm 1: Single-level decomposition for an approximate solution of $A_N x_N = r_N$:

(1) $x_N \leftarrow 0,\ x_\phi \leftarrow 0$
(2) One forward GS-sweep on $A_\phi x_\phi = P_{S_h}^T r_N$
(3) $x_N \leftarrow x_N + P_{S_h} x_\phi$
(4) One symmetric GS-sweep on $A_N x_N = r_N$
(5) $x_\phi \leftarrow 0$
(6) One backward GS-sweep on $A_\phi x_\phi = P_{S_h}^T (r_N - A_N x_N)$
(7) $x_N \leftarrow x_N + P_{S_h} x_\phi$

We point out that the additional smoothing operations in S_h can improve the convergence rate of a solver substantially. In the follwing sections we are going to devise algebraic multigrid procedures in order to replace the steps (2), (4), and (6).

4 Algebraic Coarsening of Potential Spaces

The goal of this section is to construct an algebraic multigrid procedure for an approximate solution of the equation system $A_\phi x_\phi = r_\phi$ appearing in steps (2) and (6) of algorithm 1. In effect we are solving a Poisson problem in the space of linear Lagrange-type finite elements; to see this, simply replace $H(curl; \Omega)$ with $grad S_h$ in (3) and (5) (see also [5]).

Algebraic coarsening is fairly standard for such potential problems. In [3] an algorithm is devised, which utilizes only the matrix graph for coarsening. This technique is attractive in our problem context, as the coarse grids created for A_ϕ may also be used for the space of edge elements N_h (cf. section 5). We shall only present the essentials of the algebraic coarsening scheme; further details can be found in [3].

In the context of linear Lagrange-type finite elements it is obvious that the graph defined by the system matrix A_ϕ reflects the geometric structure of the respective triangulation $\mathcal{T}_h$. The diagonal entries of A_ϕ are associated with the vertices (nodes) of the mesh; off-diagonal matrix entries contribute the edges.

The basic idea for coarsening is to select a favorably distributed subset of the grid (or matrix graph) nodes in order to define the nodes of the coarse grid. We shall construct a simple, quasi-linear interpolation for the grid transfer. Thus for any matrix the nodes will be divided into two disjoint sets: the *master nodes*, which are to become the nodes of the coarse grid, and the *slave nodes*, which will be dropped. There are three rules for the selection of master nodes and the definition of the coarse-to-fine transfer operator P:

(1) No master node may be directly connected to another master node (two nodes i and j are connected, if there exists a matrix entry $a_{ij} \neq 0$).
(2) There should be as many master nodes as possible.
(3) The values of all master nodes are transferred with weight 1. The value for a slave node s is interpolated from the n_s master nodes it is connected to; each master node is assigned the weight $\frac{1}{n_s}$.

All nodes not declared to be masters will be slave nodes. An exception appears on the finest grid, where nodes on Dirichlet boundaries are not taken into consideration. By the rules (1) and (2) the master nodes are dispersed in a quasi-uniform manner on the grid, thus trying to imitate geometric coarsening (at least on regular triangulations). Each slave node is coupled to at least one master node, which is essential for effective smoothing. Rule (3) guarantees that on regions with constant solution the transfer is exact; thus the kernel of the gradient is preserved.

We do not aim at an algorithm that satisfies rule (2) exactly, this might become a too costly procedure. Instead, we shall be content with a scheme that achieves a nearly optimal distribution of the master nodes. So we present an “advancing front” algorithm, which proceeds by examining the immediate neighbours of the nodes at a temporary “front”.

Algorithm 2: Selection of master and slave nodes for coarsening:

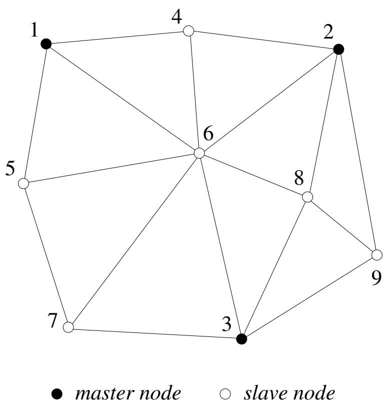
Let N denote the set of all (unconstrained) nodes of the matrix graph, M the set of master nodes, and S the set of slave nodes. Nodes neither in M nor in S form the set of remaining nodes R.

(1) Pick one node i of the grid, preferably on the boundary, which will become the first master node: $M = \{ i \}$. Set $S = \{ \}$ and $R = N \setminus \{ i \}$.
(2) Pick all those nodes of R which are coupled to a node in M or S. These nodes form a temporary set F, the active front: $F = \{f_1, \ldots, f_n\}$, $R \leftarrow R \setminus F$.
(3) Determine the status for the nodes $f_i \in F$ one by one:
   - If f_i is connected to any node in M, move it to the set of slave nodes: $S \leftarrow S \cup \{f_i\}$, $F \leftarrow F \setminus \{f_i\}$.
   - Otherwise f_i is to become a master node: $M \leftarrow M \cup \{f_i\}$, $F \leftarrow F \setminus \{f_i\}$.
(4) Repeat steps (2) and (3) until R is empty.

An example for a two-dimensional mesh is shown in figure 1.

A recursive application of algorithm 2 endows us with the desired sequence of “grids”. We stop coarsening if the number of unknowns is below 500, then a direct factorization can be obtained cheaply. Naturally, all coarse grid matrices are computed by a Galerkin product.

With algorithm 3 we present a recursive algebraic multigrid V-cycle. It is constructed in a symmetric manner, using varying numbers of smoothing steps on the different levels. Increasing the number of smoothing steps n^l on coarser levels l may improve the performance. On the fine grid we smooth just once and increase n^l by one on every subsequent coarser grid, i.e. if L denotes the number of levels, n^l is given by $n^l = L - l + 1$ (we assign $l = 1$ to the coarsest level).



$$P = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ 1 & 0 & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & 1 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

Figure 1: *Distribution of master and slave nodes on a two-dimensional grid and the resulting prolongation matrix P. For practical reasons, the lowest numbers have been assigned to the master nodes, which will be retained on the coarse grid.*

Algorithm 3: $\mathrm{AMG}(A^l, x^l, r^l)$ (V-cycle for an approximate solution of $A^l x^l = r^l$)

(1) n^l forward GS-sweeps on $A^l x^l = r^l$

(2) $r^{l-1} \leftarrow \left(P^l\right)^T (r^l - A^l x^l)$

(3) If $l - 1 = 1$: solve $A^1 x^1 = r^1$ by forward-backward substitution

else : $x^{l-1} \leftarrow 0$, call **AMG**$(A^{l-1}, x^{l-1}, r^{l-1})$

(4) $x^l \leftarrow x^l + P^l\, x^{l-1}$

(5) n^l backward GS-sweeps on $A^l x^l = r^l$

5 Algebraic Coarsening of Nédélec spaces

In the introduction we addressed some obstacles impeding the algebraic coarsening of matrices created via edge element discretizations. On the other hand, in the previous section we have constructed a comparatively simple coarsening scheme for potential problems discretized by linear Lagrange-type basis functions.

So let us take another point of view for our electrodynamical problems (3) and (5): Could we do better if not working with edge elements, but with Lagrange-type basis functions, where each spatial field component is approximated separately? The answer is a restricted “yes”. Considering, for example, the double-curl equation (4) in a homogeneous medium, we may re-formulate the differential operator in order to obtain

$$-\frac{1}{\mu}\Delta\widehat{\boldsymbol{E}} + \frac{1}{\mu} grad(div\widehat{\boldsymbol{E}}) - \omega^2\,\widehat{\epsilon}\,\widehat{\boldsymbol{E}} = 0\,. \tag{9}$$

As $div\widehat{\boldsymbol{E}} = 0$ holds in a homogeneous medium, we have a scalar Helmholtz equation for each of the decoupled components of $\widehat{\boldsymbol{E}}$. Discretizing the domain with linear Lagrange-type bases for each of the components, algebraic multigrid boils down to coarsen three nodal finite element spaces S_h separately (in the sequel the terms "nodal" and "vertex-based" will be used synonymously). Neglecting for the moment the mass term in (9), it becomes obvious that the coarsening algorithm constructed for the potential problem in section 4 should now be applicable to each spatial field component. Moreover, as only the mesh topology is exploited for coarsening, we get the same coarse grid structures both for the potential ϕ and for the individual components of the discretized electric field $\widehat{\boldsymbol{E}}$.

However, it is well-known that the proposed nodal basis for $\widehat{\boldsymbol{E}}$ (henceforth symbolized by S_h^d, with d denoting the space dimension) does not provide a discrete representation which complies with the required continuity relations of the field at re-entrant corners. Similar difficulties are encountered in inhomogeneous media.

In fact, we shall use the nodal vector basis S_h^d only as a framework for creating a subspace of the Nédélec space N_h, and it will be this subspace where algebraic coarsening is carried out. Loosely speaking, we are going to construct a component-wise splitting of the basis functions in N_h, accompanied by a weighted aggregation at the grid vertices.

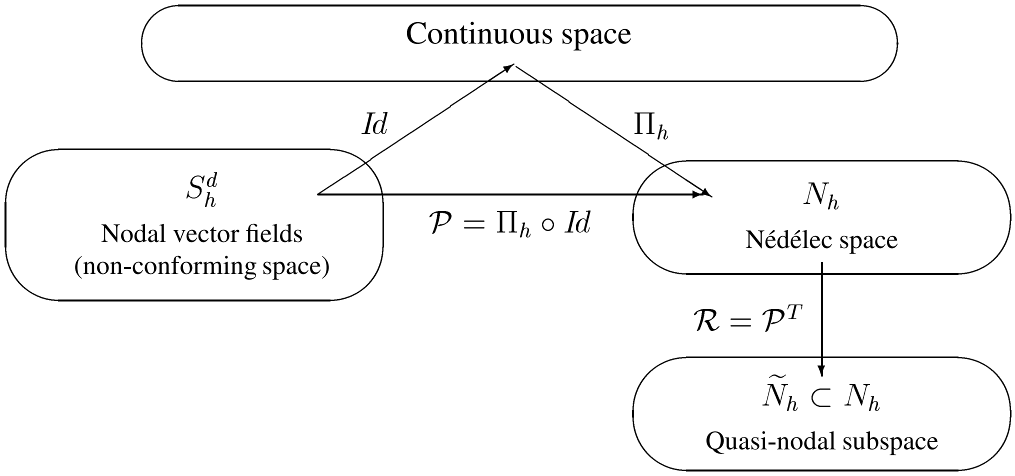


Figure 2: *Construction of the subspace* $\widetilde{N}_h$.

A schematic for this approach is depicted in figure 2. Recall from (7) how a vector field of the continuous space is projected onto the degrees of freedom of the Nédélec space N_h. As well we can project any field $\boldsymbol{E}^S \in S_h^d$:

$$\mathcal{P} = \Pi_h \circ Id : \quad S_h^d \to N_h \; .$$

The identity operator Id has been introduced merely for illustrative purposes, allowing the depicted "detour" in figure 2. Adopting the terminology of multigrid, we call $\mathcal{P}$ a prolongation operator.

On any mesh $\mathcal{P}$ can be evaluated easily. A vector field $\boldsymbol{E}^S \in S_h^d$ has a component-wise splitting $\boldsymbol{E}^S(\boldsymbol{x}) = \sum_i E_i^S(\boldsymbol{x})\, \boldsymbol{e}_i$, where $\boldsymbol{e}_i$ denotes an axis of the global Cartesian coordinate frame. Each component $E_i^S(\boldsymbol{x})$ is expanded with linear Lagrange-type basis functions associated with the vertices of the mesh. Thus, after switching to matrix-vector notation, a matrix entry $\mathcal{P}_{jk}$ of the prolongation operator is given by

$$\mathcal{P}_{jk} \;=\; \int_{e_j} \lambda_m(\boldsymbol{x})\, \boldsymbol{e}_i \cdot \mathbf{t}_j \, ds \;=\; \frac{1}{2}\, \boldsymbol{e}_i \cdot \mathbf{t}_j \; l_j \; . \tag{10}$$

Here e_j is the edge with tangent vector $\mathbf{t}_j$ and length l_j, $\lambda_m(\boldsymbol{x})$ is the Lagrange-type basis function of vertex m, and $\boldsymbol{e}_i$ is one of the global coordinate axes (see figure 3). The matrix index k has to be defined by an appropriate mapping of m and i. Note that a one-point quadrature rule is sufficient for calculating the path integral in (10), yielding the factor $\frac{1}{2}$ with respect to λ_m.

Now we are able to construct our subspace $\widetilde{N}_h$ with the aid of the prolongation operator $\mathcal{P}$. It is well-known from the multilevel theory of nested finite element spaces that the adjoint of a prolongation operator (the canonical restriction $\mathcal{R} = \mathcal{P}^T$) defines the coarse grid bases as linear combinations of fine grid basis functions. Although the term "coarse grid" is (yet) somewhat misleading in our present context, we define the functions of $\widetilde{N}_h$ in the very same manner. Using the symbol φ for basis functions of N_h and $\tilde{\varphi}$ for functions in $\widetilde{N}_h$, we have $\tilde{\varphi}_k = \sum_j \mathcal{R}_{kj}\, \varphi_j$. We thus obtain the quasi-nodal, i.e. vertex-based subspace $\widetilde{N}_h$, where d basis functions are located at each mesh vertex (recall that d denotes the space dimension, $d = 2, 3$). Of course, this arrangement is structurally equivalent to that of S_h^d.

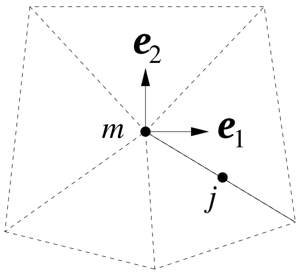


Figure 3: *Triangulation with global Cartesian coordinate axes $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ at vertex m.*

In order to coarsen $\widetilde{N}_h$ in practice, we group its functions into d disjoint subsets, each subset being associated with an axis of the global Cartesian coordinate frame. As already suggested above, each subset is coarsened by utilizing the prolongation operators P^l of algorithm 2 constructed for the potential space S_h. Thus, if we have a coefficient vector $x^l_{\tilde{N}} \in \widetilde{N}^l_h$ on some level l, its components $x^l_{\tilde{N},i}$ associated with the global axis $\boldsymbol{e}_i$ are transferred by $x^{l+1}_{\tilde{N},i} = P^l x^l_{\tilde{N},i}$ to the higher level. Whatever coarsening scheme is chosen in practice, we consider it an essential point not to mix up any spatial components.

Postponing some details for the moment, we now can formulate the algebraic multigrid version of algorithm 1. Single-level smoothing steps are extended to AMG-V-cycles due to algorithm 3.

Algorithm 4: AMG-preconditioner for an approximate solution of $A_N x_N = r_N$:

(1) $x_N \leftarrow 0,\; x_\phi \leftarrow 0$
(2) One AMG-V-cycle on $A_\phi x_\phi = P^T_{S_h}\, r_N$
(3) $x_N \leftarrow x_N + P_{S_h} x_\phi$
(4) One forward GS-sweep on $A_N x_N = r_N$
(5) $x_{\tilde{N}} \leftarrow 0$
(6) One AMG-V-cycle on $A_{\tilde{N}} x_{\tilde{N}} = \mathcal{P}^T\,(r_N - A_N x_N)$
(7) $x_N \leftarrow x_N + \mathcal{P} x_{\tilde{N}}$
(8) One backward GS-sweep on $A_N x_N = r_N$

(9) $x_\phi \leftarrow 0$
(10) One AMG-V-cycle on $A_\phi x_\phi = P^T_{S_h}\,(r_N - A_N x_N)$
(11) $x_N \leftarrow x_N + P_{S_h} x_\phi$

Vectors and matrices belonging to $\widetilde{N}_h$ have been tagged with the symbol $\widetilde{N}$. Naturally, the matrix $A_{\tilde{N}}$ is obtained by the Galerkin product $A_{\tilde{N}} = \mathcal{P}^T A_N \mathcal{P}$ (for time-harmonic fields a modification of A_N is required, which will be discussed below). Observe that the symmetric Gauss-Seidel sweep (4) in algorithm 1 has not been simply replaced. We rather embed the AMG-cycle for $\widetilde{N}_h$ between one pre- and one post-smoothing step within N_h.

Remark: For efficiency reasons, a slight modification is advisable for the AMG-cycle in step (6). As N_h and $\widetilde{N}_h$ belong to the very same grid and thus reflect the same mesh spacing, pre- and post-smoothing should be skipped on the finest level L of the AMG-algorithm. In a certain sense these operations would be redundant to the smoothing steps within N_h.

In order to elucidate the above remark, let us have a brief look at the matrix sizes of the different spaces. The dimension of N_h is given by the number of (non-constrained) edges of the triangulation $\mathcal{T}_h$; the number of vertices and the space dimension determine that of $\widetilde{N}_h$. The respective ratios are shown in the third column of table 1 for the case of uniform grids with infinite extension. Also the ratios of the numbers of matrix entries are reported.

space dimension d	$\frac{n_e}{n_v}$	$\frac{n_e}{n_v \cdot d}$	$\frac{n(A_N)}{n(A_{\tilde{N}})}$
2	3	1.5	0.5
3	6	2	0.7

Table 1: *Typical ratios of quantities relevant for the spaces N_h and $\widetilde{N}_h$ on a regular triangulation of infinite extension (n_e: number of edges, n_v: number of vertices, n: number of matrix entries).*

As table 1 reveals, the matrices of $\widetilde{N}_h$ are larger than the corresponding ones of N_h. Although being of smaller dimension, they contain more entries in each row. This is typical for vertex-based stencils compared to those of edge elements; furthermore we encounter coupling terms related to the different spatial field components. Thus it is advisable not to employ $A_{\tilde{N}}$ for smoothing on the top level of the AMG-cycle in question. $A_{\tilde{N}}$ only plays a temporary role in the setup phase when the coarse grid matrices are computed.

A vexing question remains with regard to boundaries equipped with Dirichlet conditions. Here the tangential components of the field are constrained, a situation, which can be handled easily with edge elements. To handle the nodal components properly in $\widetilde{N}_h$, those parallel to the boundary should be constrained likewise. In general, this may cause severe difficulties, as unique normal vectors need not exist at the vertices of an arbitrary polygonal domain (see fig. 4).

We tested two different approaches for handling Dirichlet-type boundaries in $\widetilde{N}_h$. Our first strategy was to introduce a rotated coordinate system at each boundary vertex instead of using the global Cartesian coordinate frame. The only degree of freedom attached to such a vertex is a field component aligned with the normal vector of the boundary; field components in the tangential direction are constrained to zero. If the vertex is located at a corner like in figure 4, the normal was defined by averaging the normals of the adjacent element faces. Without going into detail, we want to mention that this technique in general did not perform substantially better than the following one.

In our second approach we use a very simple alternative: Here *all* components of any vertex on a Dirichlet boundary are constrained. Although this invariably aggravates the coarse grid approximation, the drawback should not be too serious, as only one layer of elements along the border of the domain is affected. For the numerical experiments in section 6 we shall always utilize this strategy.

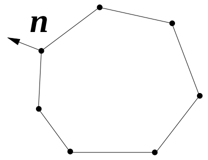


Figure 4: *Polygonal domain with non-unique outer normals at grid vertices.*

Let us now return to our variational equation (5), which is solved in time-harmonic problems. To cope with the resulting indefinite systems in multigrid methods, it is necessary to resolve modes with negative eigenvalues on the coarsest grid [5, 10]. This condition can hardly be fulfilled in algebraic multigrid, so we restrict ourselves to tackling the positive part of the spectrum of A_N. An approach like this was suggested in [28], where the use of a modified matrix is advocated for multigrid preconditioning. In our case this would lead to dropping the mass term in eq. (5), leaving us with a positive semidefinite system matrix. Of course, the coarse grid matrix now may not be factorized, which is no substantial drawback, if we apply coarsening down to very few degrees of freedom and stick to smoothing. However, this remedy did not yield very satisfactory results in numerical experiments; we presume that the large dimension of the nullspace of the $curl$-operator is responsible for this effect.

So we propose to shift the negative eigenvalues into the positive range, i.e. we do not drop the mass term, but change its sign in (5):

$$(\frac{1}{\mu} curl\widehat{\boldsymbol{E}},\, curl\widehat{\boldsymbol{v}}) \;+\; \omega^2(\widehat{\epsilon}\,\widehat{\boldsymbol{E}}, \widehat{\boldsymbol{v}})\ .$$

This modification yields a positive definite matrix A_N^+, which, instead of A_N, is used in the Galerkin product for generating $A_{\tilde{N}}$. This measure can be expected to distort only modes with small eigenvalues. With regard to the kernel modes of $curl$, these are smoothed correctly in the separate AMG-cycles for the potential space S_h. What we have to expect is a deterioration of the convergence rates with increasing angular frequency ω, when more and more modes of the kernel's orthogonal complement are shifted into the negative range. An example for this impact will be presented in the numerical experiments.

Of course, the problem of negative eigenvalues does not arise in time-stepping methods for eddy current computations due to equation (3). So we confine the numerical experiments of the following section to the less favorable case of time-harmonic fields.

6 Numerical Experiments

In all our numerical experiments we use conjugate gradients as a basic solver for the linear systems. In order to assess both the effects of single-level subspace corrections in the potential space S_h (algorithm 1) and the convergence behaviour of algebraic multigrid (algorithm 4), we employ three different preconditioners:

SGS:	One symmetric Gauss-Seidel sweep in N_h is carried out; smoothing in S_h is omitted.
SGS-P:	Preconditioning by algorithm 1.
AMG:	Preconditioning by algorithm 4.

The initial guess for the solution is always zero. The iterations are terminated when the Euclidean norm of the residual $r_N^{(n)}$ in step n has dropped by ten orders of magnitudes compared to the initial one: $|r_N^{(n)}| < 10^{-10}|r_N^{(0)}|$. In the subsequent tables we report the iteration counts and the processor (CPU) times for the different grids with respect to the dimension of the space N_h (i.e. the number of unconstrained edges of the triangulations). In the last columns of the tables the CPU times required for constructing the coarse grid data structures in AMG are given. Results for grids with low dimension are omitted as being of no practical relevance with regard to CPU-times, furthermore we do not report results if the iteration counts are excessively high.

Test runs with alternative solvers, like conjugate residuals or symmetric QMR, in general did not yield substantially different convergence rates compared to the conjugate gradient algorithm.

Our *first experiment* is carried out on the unit square $\Omega =]0,1[^2$. We assume a homogeneous material with $\widehat{\epsilon} = \mu = 1$. On the left hand side of the domain, i.e. along the line between the points $(0,0)$ and $(0,1)$, we apply a Dirichlet boundary condition for the tangential component of the electric field $\widehat{\boldsymbol{E}}$: $\widehat{\boldsymbol{E}}_y = \sin(\pi y)$. On the remaining boundaries a natural boundary condition holds (by choosing this setting we have no specific application in mind). We start with an angular frequency $\omega = 1.5\pi$ and raise it in two steps to 6π in order to demonstrate its influence onto the convergence rates

We start with a regular grid and refine it uniformly in several steps. Pictures of the grid structure and electric fields are shown in figure 5. As the the results in table 2 reveal, the single-level SGS-preconditioners deteriorate with increasing node numbers, whereas AMG remains stable with respect to the iteration counts. As to be expected, all convergence rates decay with increasing frequency, when more and more negative eigenvalues appear in the system matrices; the modes in question cannot be smoothed by any of the preconditioners. For all values of ω the additional smoothing in the potential space on single level (SGS-P) does not pay off compared to plain SGS-preconditioning. The fact that we have a smooth solenoidal field may explain this observation.

For our *second experiment* we distort the domain of the previous one in order to obtain concave corners where the field is singular. The boundary conditions are like in experiment 1. Furthermore we set the coefficient μ to 10^4 on a part of the region, simulating some kind of ferromagnetic material. The situation is depicted in figure 6.

The frequency is set to $\omega = 0.05\pi$. For creating the finite element grids we employ both uniform and adaptive mesh refinement. As figure 6 shows, we produce grids comprising unfavorably shaped elements with obtuse angles.

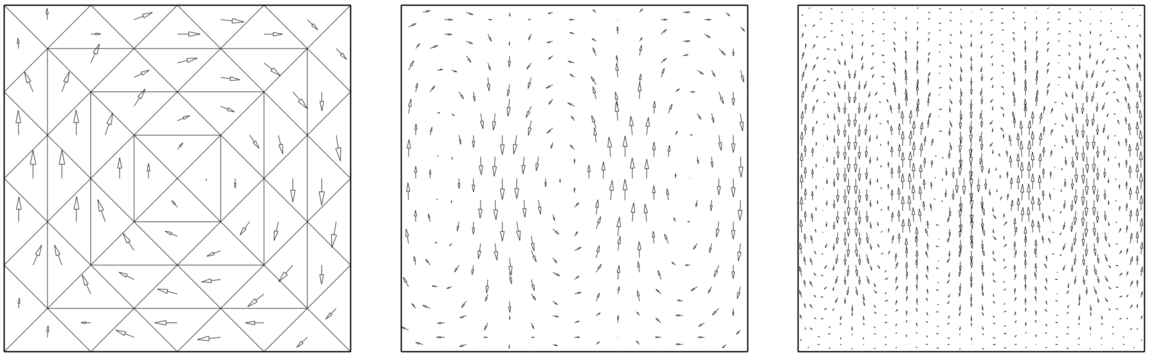


Figure 5: *Grid structure and electric field for* $\omega = 1.5\pi$ *(left hand side), field for* $\omega = 3\pi$ *(center), and field for* $\omega = 6\pi$ *in experiment 1.*

dimN_h	#Iter			CPU [sec]			
	SGS	SGS-P	AMG	SGS	SGS-P	AMG	Setup AMG
$\omega = 1.5\pi$							
6176	400	219	19	10	13	2	0.5
24640	816	424	19	94	111	11	1.5
98432	1592	839	19	764	935	49	6
$\omega = 3\pi$							
6176	500	353	42	13	21	5	0.5
24640	1028	679	41	120	178	24	1.5
98432	2069	1392	42	987	1512	107	6
$\omega = 6\pi$							
6176	847	1114	171	21	66	21	0.5
24640	1680	2586	174	195	678	102	1.5
98432	3122	4961	166	1494	5390	422	6

Table 2: *Iteration counts and CPU-times for experiment 1.*

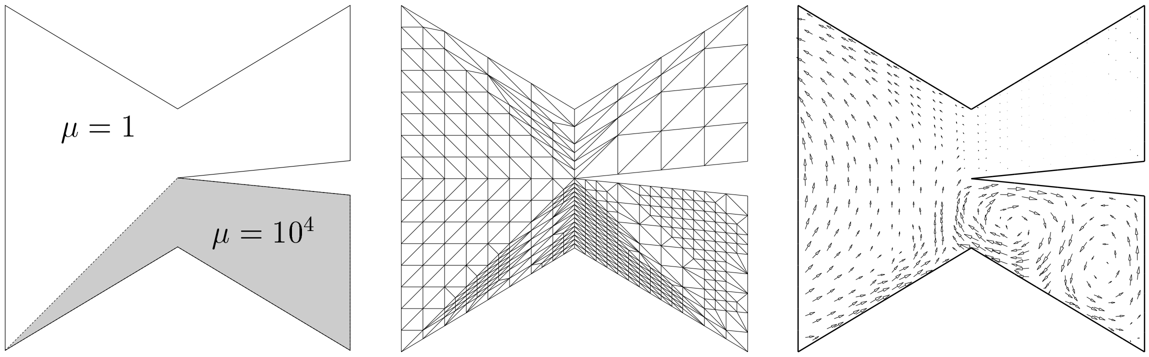


Figure 6: *Values for μ on the domain (left hand side), adaptively refined grid (center), and electric field in experiment 2.*

$\dim N_h$	#Iter			CPU [sec]			
	SGS	SGS-P	AMG	SGS	SGS-P	AMG	Setup AMG
Uniform mesh refinement							
3120	7676	377	28	86	10	2	0.2
12384	$> 10\,000$	778	30		97	8	0.7
49344		1614	27		888	34	2.6
196992		4462	27		9975	141	11
Adaptive mesh refinement							
3853	$> 10\,000$	412	29		14	3	0.3
8509		565	43		48	8	0.8
18989		1268	41		264	19	1.2
40195		1959	51		905	56	2.7
83609		$> 10\,000$	57			133	5.2
171338			56			275	13

Table 3: *Iteration counts and CPU-times for experiment 2.*

Table 3 monitors the outcome of this experiment. Despite the large jump in μ the algebraic multigrid preconditioner maintains a stable convergence rate in the case of uniform mesh refinement, whereas on adaptively created grids some deterioration is encountered. Simple SGS-preconditioning now renders devastating results already on comparatively coarse triangulations.

In our *third experiment* we investigate a three-dimensional structure: A tapered microstrip line is placed on a dielectric substrate with $\epsilon = 10$, the remaining area of the domain is filled with air ($\epsilon = 1$). Figure 7 shows a sketch of the arrangement.

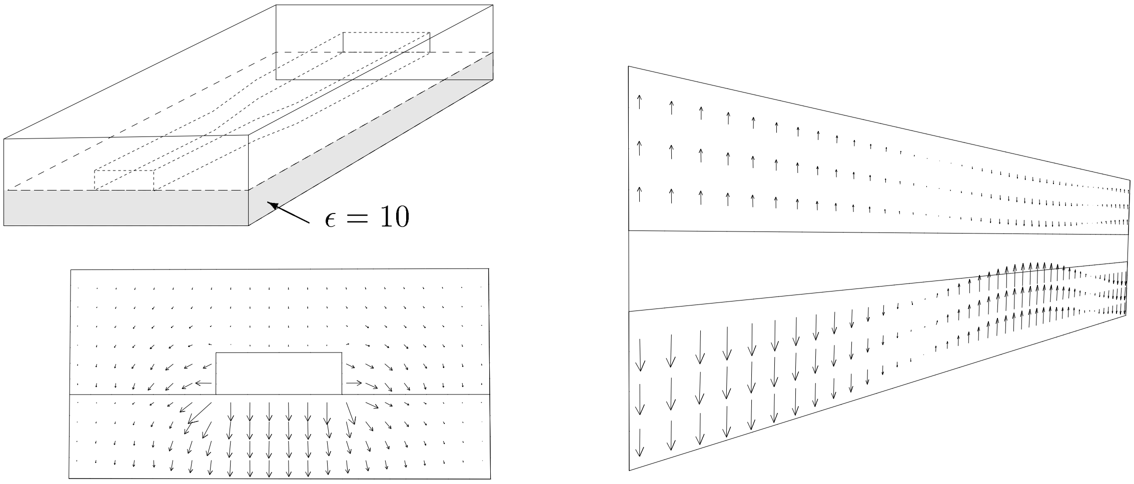


Figure 7: *Tapered microstrip line on a dielectric substrate with $\epsilon = 10$ (top left). Electric field on a cross section (bottom left) and on the central plane along the conductor (experiment 3).*

The microstrip line is enclosed in a metallic box except for the front and rear planes. We assume perfect conductors which are modeled by homogeneous Dirichlet boundary conditions, i.e. the tangential components of $\widehat{\boldsymbol{E}}$ are set to zero on the conductor surfaces.

On the front plane we apply an incoming quasi-TEM-wave, which is the dominating mode guided by striplines. Its field components are nearly perpendicular to the direction of propagation, accounting for the name transverse electromagnetic wave [22]. The wave travels along the line and leaves the structure at the rear plane. The applied boundary conditions on these ports are of Silver-Müller type; for a more detailed description the reader is referred to [5]. The applied frequency is $\omega = 4\pi$; a plot of the resulting field is shown in figure 7. Table 4 gives the results both for uniformly and adaptively refined grids.

Also in this experiment the SGS-preconditioner reveals a poor performance. The transverse components of the fields are of quasi-static type, thus smoothing in the potential space is very effective. What aggravates the situation for AMG is the large part of the boundary equipped with Dirichlet conditions, where all components are constrained in the quasi-nodal space $\widetilde{N}_h$. However, again the convergence rates for AMG are rather stable.

The setup times for the AMG-preconditioner are comparatively high in the 3D-case. Nearly all the time is spent in the Galerkin-products for computing the matrices of $\widetilde{N}_h$. We have

$\dim N_h$	#Iter			CPU [sec]			
	SGS	SGS-P	AMG	SGS	SGS-P	AMG	Setup AMG
Uniform mesh refinement							
5644	1381	141	67	37	8	7	2
49496	5404	256	56	1762	173	70	8
413104	$> 10\,000$	521	59		3300	730	66
3372896		1060	58		53953	6137	596
Adaptive mesh refinement							
16107	4616	209	67	370	39	24	3
64969	$> 10\,000$	316	70		294	117	12
277514		505	68		2188	571	49
1015112		793	71		13120	2284	196

Table 4: *Iteration counts and CPU-times for experiment 3.*

implemented a rather simple version for this procedure, in particular we do not exploit the block structure of the matrices. Thus some acceleration should be possible.

7 Conclusions

We have devised a nested symmetric preconditioner, which may significantly enhance the convergence of linear solvers for edge element discretizations. Algebraic multigrid methods are developed both for an efficient handling of the kernel modes of $curl$ and for the Nédélec space. As for the latter one, the basic idea for mesh coarsening is to switch to an auxiliary quasi-nodal subspace, which is constructed by a spatial component splitting of the original basis functions.

With AMG a rather stable convergence behaviour may even be observed in situations where single-level preconditioning fails to render stability. We have not included numerical examples dealing with eddy current computations. In contrast to the time-harmonic case, there are no modes with negative eigenvalues, for which AMG cannot provide any error reduction. So we can expect the typical low-frequency behaviour reported in the experiments.

Up to now we have no thorough theoretical understanding how the grid modes of N_h are affected by the transfer to the auxiliary subspace $\widetilde{N}_h$ and by the subsequent coarsening steps. The situation is delicate on irregular meshes and on domains with inhomogeneous materials. An additional complication arises by the kernel of $curl$, whose modes mix up with those of its orthogonal complement by the transfer. Nevertheless, a better insight might yield further improvements of the proposed algorithms.

Acknowledgment. This work was supported by the Deutsche Forschungsgemeinschaft.

References

[1] R.E. Bank and C. Wagner. Multilevel ILU decomposition. *Numer. Math.*, 82, pp. 543-576, 1999.

[2] M.L. Barton and Z.J. Cendes. New vector finite elements for three-dimensional magnetic field computation. *J. Appl. Phys.*, 61, pp. 3919-3921, 1987.

[3] R. Beck. Graph-based algebraic multigrid for Lagrange-type finite elements on simplicial meshes. *Preprint SC 99-22*, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 1999.

[4] R. Beck, P. Deuflhard, R. Hiptmair, R.H.W. Hoppe, and B. Wohlmuth. Adaptive multilevel methods for edge element discretizations of Maxwell's equations. *Surveys on Mathematics for Industry*, 8, no. 3-4, pp. 271–312, 1999.

[5] R. Beck and R. Hiptmair. Multilevel solution of the time-harmonic Maxwell's equations based on edge elements. *Int. J. Numer. Meths. in Eng.*, 45, pp. 901-920, 1999.

[6] D. Boffi, P. Fernandes, L. Gastaldi, and I. Perugia. Compuational models of electromagnetic resonators: Analysis of edge element approximation. *SIAM J. Numer. Anal.*, 36, pp. 1264-1290, 1999.

[7] A. Bossavit. Whitney forms: a class of finite elements for three-dimensional computation in electromagnetism. *Inst. Elec. Eng. Proc., Part A*, 135, pp. 493-500(8), 1988.

[8] A. Bossavit. Solving Maxwell's equation in a closed cavity and the question of spurious modes. *IEEE Trans. Magn.*, 26, pp. 702-705, 1990.

[9] D. Braess. Towards algebraic multigrid for elliptic problems of second order. *Computing*, 55, pp. 379-393, 1995.

[10] J.H. Bramble, D.Y. Kwak, and J.E. Pasciak. Uniform convergence of multigrid V-cycle iterations for indefinite and nonsymmetric problems. *SIAM J. Numer. Anal.*, 31 (6), pp. 1746-1763, 1994.

[11] A. Brandt. Algebraic multigrid theory: the symmetric case. *Appl. Math. Comput.*, 19, pp. 23-56, 1986.

[12] A. Brandt, S.F. McCormick, and J.W. Ruge. Algebraic multigrid (AMG) for sparse matrix equations. In D.J. Evans, editor, *Sparsity and its applications*, pp. 257-284, Cambridge University Press, Cambridge, 1985.

[13] Qianshun Chang, Yau Shu Wong, and Hanging Fu. On the algebraic multigrid method. *J. Comp. Phys.*, 125, pp. 279-292, 1996.

[14] B.M. Dillon and J.P. Webb. A comparison of formulations for the vector finite element analysis of waveguides. *IEEE Trans. Microw. Theory Tech.*, 42 (2), pp. 308-316, 1994.

[15] J. Fuhrmann. A modular algebraic multilevel method. Technical report, Preprint 203, WIAS, Berlin, 1995.

[16] Wolfgang Hackbusch. *Multi-Grid Methods and Applications*. Springer, Berlin, 1985.

[17] R. Hiptmair. *Multilevel Preconditioning for Mixed Problems in Three Dimensions*. PhD thesis, Mathematisches Institut, Universität Augsburg, 1996.

[18] R. Hiptmair. Multigrid method for Maxwell's equations. *SIAM J. Numer. Anal.*, 36 (1), pp. 204-225, 1999.

[19] Jianming Jin. *The Finite Element Method in Electrodynamics*. John Wiley & Sons, New York, 1993.

[20] Arnulf Kost. *Numerische Methoden in der Berechnung elektromagnetischer Felder*. Springer, Berlin, 1994.

[21] J.C. Nédélec. Mixed finite elements in $\mathbb{R}^3$. *Numer. Math.*, 35, pp. 315-341, 1980.

[22] S. Ramo, J. R. Whinnery, and Th. van Duzer. *Fields and Waves in Communication Electronics*. Wiley, New York, 1984.

[23] A. Reusken. An approximate cyclic reduction scheme. In W. Hackbusch and G. Wittum, editors, *Multigrid Methods V. Lecture Notes in Computational Science and Engineering*. Springer, Heidelberg, 1998.

[24] J.W. Ruge and K. Stüben. Algebraic multigrid. In S.F. McCormick, editor, *Multigrid Methods*, Vol. 4, pp. 73-130, SIAM, Philadelphia, 1987.

[25] P. Vaňek, J. Mandel, and M. Brezina. Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems. *Computing*, 56, pp. 179-196, 1996.

[26] C. Wagner. Introduction to Algebraic Multigrid. Technical report, University of Heidelberg, Germany, 1999.

[27] Pieter Wesseling. *An Introduction to Multigrid Methods*. Wiley, Chichester, 1992.

[28] H. Yserentant. Preconditioning indefinite discretization matrices. *Num. Math.*, 54, pp. 719-734, 1988.

[29] de Zeeuw. Matrix-dependent prolongations and restrictions in a black-box multigrid solver. *J. Comp. Appl. Math*, 33, pp. 1-27, 1990.