Algebraic MultiGrid Methods for Nodal and Edge based Discretizations of Maxwell's Equations

Abstract— **This paper deals with the efficient solution of Maxwell's equation in the static and eddy current case using enhanced Algebraic MultiGrid methods. The magnetic vector potential is used as the field variable and Lagrange (nodal) as well as Nédélec (edge) finite elements are applied for the spatial discretization. Numerical studies for the static case (TEAM 20), the eddy current case in time domain (Magnetic Resonance Imaging scanner) and in frequency domain (electric transformer) will demonstrate the applicability of the developed Algebraic MultiGrid methods.**

I. Introduction

Numerical computations of electromagnetic fields are performed for more than 20 years. Many different formulations using Lagrange (nodal) as well as Nédélec (edge) finite elements for the spatial discretization have been developed (see e.g. [4], [3]). Even many commercial codes are available for the computation of electromagnetic fields in 3D. However, the efficient solution with respect to CPU-time and memory amount for the resulting algebraic system of equations is still a challenging task, and requires ongoing research.

In this paper we will concentrate on the static and eddy current case (both in time and frequency domain) by applying enhanced Algebraic MultiGrid (AMG) solvers developed in the last 4 years. The paper is organized as follows: In Sec. II we will describe the correct formulation and Finite Element (FE) discretization of Maxwell's equations for the eddy current case. Next, in Sec. III we will review the convergence behavior of iterative solvers, the properties of MultiGrid (MG) methods, and the motivation for AMG solvers. The main part of this paper concentrates on a detailed discussion of AMG methods and special adaption for algebraic systems of equations arising from nodal as well as edge FE-discretizations (see Sec. IV and V). Finally, we will report on the performance of the developed AMG methods for three case studies: TEAM 20, Magnetic Resonance Imaging scanner and electric transformer.

II. Governing Equations and Finite Element Discretizations

The electromagnetic field is fully described by Maxwell's equations [33]. Restricting the problem class to the quasi-static (eddy current) case, we arrive at the following partial differential equation for the magnetic vector potential $\mathbf{A}$

$$\gamma \frac{\partial \mathbf{A}}{\partial t} + \nabla \times \nu \nabla \times \mathbf{A} = \mathbf{J}_\mathrm{i} \tag{1}$$

with boundary condition $\mathbf{n} \times \mathbf{A} = 0$ and $\mathbf{n}$ the unit outward normal vector. In (1) $\mathbf{J}_\mathrm{i}$ denotes the impressed current density, ν the magnetic reluctivity and γ the electric conductivity. Furtheron, the following interface conditions have to be fulfilled

$$[\mathbf{A} \times \mathbf{n}] = 0 \; ; \quad [\nu \; \mathbf{n} \times \nabla \times \mathbf{A}] = 0 \; ; \quad \left[\gamma \frac{\partial \mathbf{A}}{\partial t}\right] = 0 \tag{2}$$

with $[\mathbf{Z}] = \mathbf{Z}_\mathrm{right} - \mathbf{Z}_\mathrm{left}$. For further discussions let Ω be a bounded single connected convex domain with boundary $\partial\Omega = \Gamma$. The function spaces $L^2(\Omega)$ and $H_0^1(\Omega)$ are defined as usual, see e.g. [1]. Therewith, the variational formulation for (1) in the function space

$$\mathbf{H}_0(curl) = \{\mathbf{u} \in (L^2(\Omega))^3 \,|\, \quad \nabla \times \mathbf{u} \in (L^2(\Omega))^3, \; \mathbf{u} \times \mathbf{n}|_\Gamma = 0\} \tag{3}$$

reads as follows: Find $\mathbf{A} \in \mathbf{H}_0(curl)$ such that

$$\int_\Omega \gamma \mathbf{A}' \cdot \frac{\partial \mathbf{A}}{\partial t} \, d\Omega \;+\; \int_\Omega \nabla \times \mathbf{A}' \cdot \nu \, \nabla \times \mathbf{A} \; d\Omega = \int_\Omega \mathbf{A}' \cdot \mathbf{J}_\mathrm{i} \, d\Omega \tag{4}$$

for any $\mathbf{A}' \in \mathbf{H}_0(curl)$ is fulfilled.

It is well known, that an edge FE-discretization of (4) is $\mathbf{H}_0(curl)$-conform [23]. Nevertheless, the solution of the algebraic system requires special care in order to obtain an optimal MultiGrid solver (see e.g. [2], [13]). We suggest to add a fictive electric conductivity γ' to regions with zero electric conductivity to obtain a variational form, which is elliptic [32]. Of course, this fictive conductivity γ' has to be chosen small as compared to the reluctivity of the material. The proof of convergence even in the case of $\gamma' \to 0$ is given in [27].

For the application of nodal finite elements, we have to perform additional steps. As shown in [9], the space $\mathbf{H}_0(curl)$ can be decomposed for any convex domain Ω into

$$\mathbf{H}_0(curl) = \left(H_0^1(\Omega)\right)^3 \oplus \nabla \, H_0^1(\Omega) \,, \tag{5}$$

which corresponds to splitting the magnetic vector potential $\mathbf{A}$ as follows

$$\mathbf{A} = \mathbf{u} + \nabla\phi \,, \quad \nabla \cdot \mathbf{u} = 0 \,, \tag{6}$$

with $\mathbf{u} \in \left(H_0^1(\Omega)\right)^3$ and $\phi \in H_0^1(\Omega)$. The same decomposition is done for the test function $\mathbf{A}' = \mathbf{v} + \nabla\psi$. Since we have to guarantee $\nabla \cdot \mathbf{u} = 0$, we do so by adding the penalty term $\int_\Omega s\nu \, (\nabla \cdot \mathbf{v} \, \nabla \cdot \mathbf{u}) \, d\Omega$ to the variational formulation (4) (weighted regularization method with special function s, see [8], [17]). Therewith, the variational formulation can be stated as follows: Find $(\mathbf{u}, \phi) \in (H_0^1(\Omega)^3, \, H_0^1(\Omega))$ such that

$$\int_\Omega \nu \nabla \times \mathbf{v} \cdot \; \nabla \times \mathbf{u} \; d\Omega + \int_\Omega s \, \nu \; \nabla \cdot \mathbf{v} \, \nabla \cdot \mathbf{u} \, d\Omega + \int_\Omega \gamma(\mathbf{v} + \nabla\psi) \cdot \frac{\partial}{\partial t}(\mathbf{u} + \nabla\phi) \, d\Omega = \int_\Omega \mathbf{J}_\mathrm{i} \cdot \mathbf{v} \, d\Omega \,. \tag{7}$$

for any $(\mathbf{v}, \, \psi) \in (H_0^1(\Omega)^3, \, H_0^1(\Omega))$.

In both cases, nodal as well as edge finite elements, the spatial and subsequent time discretization lead to an algebraic system of equations

$$K_h^n(\underline{A}_h^n, \underline{\phi}_h^n) = \underline{f}_h^n \,, \quad K_h^e A_h^e = \underline{f}_h^e \,. \tag{8}$$

For the rest of the paper the superscripts $\{e, n\}$ are omitted, whenever this is possible without causing any confusion.

III. Iterative and MultiGrid Methods

Let us consider the algebraic system of equations of the form

$$K_h \underline{u}_h = \underline{f}_h \,. \tag{9}$$

Therein $K_h \in \mathbb{R}^{N_h \times N_h}$ denotes the system matrix, $\underline{f}_h \in \mathbb{R}^{N_h}$ the right hand side and $\underline{u}_h \in \mathbb{R}^{N_h}$ the solution vector of the unknown nodal (edge) quantity (mainly the magnetic vector potential). Additionally the entries of K_h are given by $k_{ij} = (K_h)_{ij} \in \mathbb{R}^{p \times p}$ with p defining the number of unknowns per node (edge). The number of unknowns N_h is related to the usual discretization parameter h by the relation $N_h = O(h^{-d})$, with $d = 2, 3$ the spatial dimension. The system matrix K_h is supposed to be sparse and symmetric positive definite (SPD). In general N_h is quite large and due to limited memory resources, iterative solvers have to be used instead of direct ones. However, the convergence of iterative solvers strongly depends on the condition number κ of the system matrix K_h

$$\kappa(K_h) = \frac{\lambda_{\max}(K_h)}{\lambda_{\min}(K_h)} \tag{10}$$

with $\lambda_{\max}$ and $\lambda_{\min}$ the largest and the smallest eigenvalue of K_h, respectively. In general, the convergence rate decreases when κ gets large. Since K_h stems from an FE-discretization of a second order partial differential equation (PDE), the condition number $\kappa(K_h)$ typically behaves like $O(h^{-2})$. In order to cope with large condition numbers, we apply a symmetric preconditioner C_h to (9), i.e.

$$C_h^{-1} K_h = C_h^{-1} \underline{f}_h \,, \tag{11}$$

with the properties

- C_h^{-1} is an approximate inverse of K_h, and
- C_h^{-1} can be applied very fast.

Consequently the condition number of the preconditioned system is much smaller than the original one. For instance, the standard preconditioner is the well known Incomplete Cholesky (IC) method (see e.g.[5]). Furthermore, the preconditioned system (11) is solved via a Krylov subspace method, i.e. Conjugate Gradient (CG) or Quasi Minimal Residual (QMR) method, see [30].
The standard preconditioners (IC, Jacobi, etc.) suffer from the fact, that the condition number still depends on h. In order to get condition numbers independent of the discretization parameter h, MultiGrid methods (MG) have to be used, for which it can be shown that the number of necessary iterations does not depend on the mesh parameter h (see e.g. [10]). As shown in [14], the most robust solution strategy for SPD linear equations (9) is PCG with geometric MultiGrid preconditioner. Recently, investigations have been carried out to adapt geometric MultiGrid (MG) methods for the fast solution of 3D electromagnetic field problems (e.g. [2], [13], [31], [35]).

However, geometric MG methods suffer from the inherent need of a hierarchical FE-mesh (see [10]), and thus algebraic MultiGrid (AMG) methods are of special interest, if at least one of the following cases arises:

1. The discretization provides no hierarchy of FE-meshes, which would be essential for the geometric MG method. This is the case for many FE-codes, especially commercial ones.
2. The coarsest grid of a geometric MultiGrid method is too large to be solved efficiently by a direct or classical iterative solver.
3. Classical iterative solvers are not efficient enough.

AMG methods try to mimic the geometric counterpart, but only rely on the information available on a given single grid (for the pioneer work on AMG see [29]). While within a geometric MG solver the construction of a matrix hierarchy is rather simple if a hierarchy of grids is available (see e.g. [10]), this task is not as easy if either the matrix only or the information on the finest grid are available. The classical AMG approach assumes an SPD system matrix which is additionally an M-matrix [29]. For such matrix classes a matrix hierarchy can be constructed, imitating the geometric counterpart well. It can be easily shown, that the information of an SPD system matrix is not enough in order to construct an efficient and robust AMG method. Therefore, we assume the knowledge of the underlying PDE, the FE-discretization scheme and additional information on the given FE-mesh. Therewith, such enhanced AMG methods are able to reproduce the behavior of geometric MG methods even for Maxwell's equation, although the system matrices are not M-matrices here.

IV. Algebraic MultiGrid Methods

In order to outline the principles of the MultiGrid method we explain them by means of a two grid method (for an overview see [7]). For this purpose h and H describe the fine and coarse grids of an FE-discretization, respectively. The linear mappings (with $N_h > N_H$)

$$R_h : \mathbb{R}^{N_h} \mapsto \mathbb{R}^{N_H} \qquad \text{and} \qquad P_h : \mathbb{R}^{N_H} \mapsto \mathbb{R}^{N_h} \tag{12}$$

are called restriction and prolongation operators. Usually we choose $R_h = P_h^T$ for AMG. Therewith, the two grid algorithm is performed as presented in Alg. 1.

Algorithm 1 TwoGridMethod

Relax ν_1 times on the fine grid $K_h \underline{u}_h = \underline{f}_h$
(e.g. Gauß-Seidel forward)
Compute the defect $\underline{d}_h = \underline{f}_h - K_h \underline{u}_h$
Restrict the defect $\underline{d}_h$ onto the coarse grid
$\underline{d}_H = R_h \underline{d}_h$
Solve the coarse grid problem $K_H \underline{v}_H = \underline{d}_H$
Prolongate the coarse grid correction $\underline{v}_H$ onto the fine grid
$\underline{v}_h = P_h \underline{v}_H$
Update $\underline{u}_h$ by $\underline{v}_h$, i.e. $\underline{u}_h = \underline{u}_h + \underline{v}_h$
Relax ν_2 times on the fine grid $K_h \underline{u}_h = \underline{f}_h$
(e.g. Gauß-Seidel backward)

By replacing the exact solution of the coarse grid problem in Alg. 1 itself by a two grid approximation, we arrive

at the recursive definition of a MultiGrid cycle. The motivation for this approach comes from examing the error of the numerical solution in the frequency domain. High frequency errors, which include local variations in the solution, are well eliminated by simple iterative smoothing methods (e.g. Gauss-Seidel smoother). Once this is achieved, further fine-grid iterations would not improve significantly the convergence rate. Therefore, the solution is transferred to a coarser grid by using an appropriate restriction operator R_h. On this grid, the low frequency errors of the fine grid manifest themselves as relatively high frequency errors, and are thus eliminated efficiently using again simple iterative smoothing methods. If the coarsest grid has been reached, the equation has to be solved exactly (e.g. direct solver), which can be done with little computational effort due to the small numbers of unknowns. Consequently, each grid level is responsible for eliminating a particular frequency bandwidth of the error.

The subsequent discussion on AMG follows mainly [25]. First of all, we have to perform the coarsening process to extract from the given system matrix (arising form the FE-discretization) matrices with decreasing dimension. The key point of the coarsening process is to construct an auxiliary matrix on which the coarsening is performed. Therewith, we can always guarantee an appropriate coarsening and, in addition to be very fast. Furtheron, we have to define the smoothing operator and the restriction (prolongation) operator for the transfer of data between the different hierarchies based on the auxiliary matrix.

A. Auxiliary Matrix

Let us assume, that the system matrix K_h stems from an FE-discretization on the FE-mesh $\omega_h = (\omega_h^n, \omega_h^e)$, with ω_h^n, $|\omega_h^n| = M_h$ being the set of nodes and ω_h^e being the set of edges (see Fig. 1). An edge is defined as a pair of

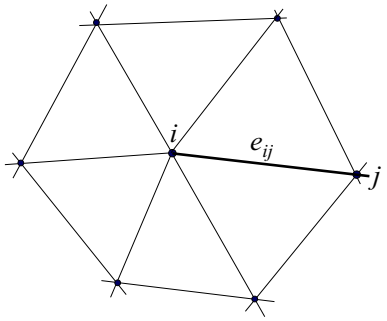

Fig. 1. Clipping of an FE-mesh in 2D.

indices for which the connection of the two points is a geometric edge. For instance, let $i,\, j \in \omega_h^n$ be the indices of the nodes $\mathbf{x}_i,\, \mathbf{x}_j \in \mathbb{R}^d$ then the edge is given by

$$e_{ij} = (i,j) \in \omega_h^e$$

and the corresponding geometric edge vector can be expressed by

$$\mathbf{a}_{ij} = \mathbf{x}_i - \mathbf{x}_j \in \mathbb{R}^d\,. \tag{13}$$

The first task we are concerned with is the construction of an auxiliary matrix $B_h \in \mathbb{R}^{M_h \times M_h}$ with the following properties:

$$(B_h)_{ij} = \begin{cases} b_{ij} \le 0 & \text{if } i \neq j, \\ 1 - \sum_{j \neq i} b_{ij} \ge 0 & \text{if } i = j\,. \end{cases} \tag{14}$$

The entries of B_h should be defined in such a way that the distance and parameter jumps of the variational forms are reflected. The matrix pattern of B_h can be constructed via different objectives: B_h reflects the geometric FE-mesh, which is of importance for an edge element discretization, or B_h reflects the matrix pattern of the system matrix K_h, which is useful for nodal FE-discretizations.

B. Coarsening Process

The auxiliary matrix B_h is a sparse M-matrix and therefore the coarsening process for B_h is straight forward and can be done in a robust way. We know that B_h represents a virtual FE-mesh $\omega_h = (\omega_h^n, \omega_h^e)$. Such a virtual FE-mesh can be split into two disjoint sets of nodes, i.e.,

$$\omega_h^n = \omega_C^n \cup \omega_F^n,\;\; \omega_C^n \cap \omega_F^n = \emptyset$$

with sets of coarse grid nodes ω_C^n and fine grid nodes ω_F^n. The splitting is usually performed such that no coarse grid nodes are connected directly and that the number of coarse grid nodes is as large as possible (see Fig. 2).

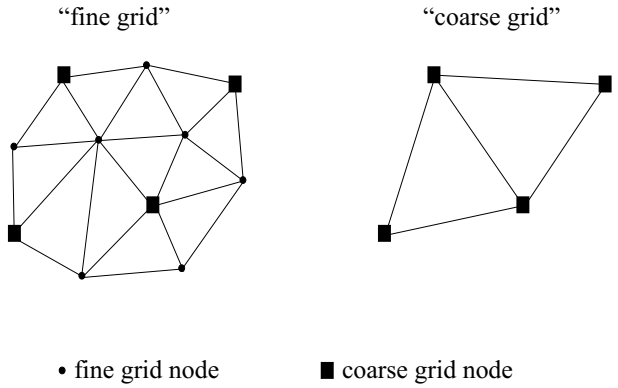

Fig. 2. Illustration of coarsening

In order to perform a coarsening algorithm, let us introduce the following sets:

$$\begin{aligned} N_h^i &= \{j \in \omega_h^n \,:\, |b_{ij}| \neq 0\,, i \neq j\}\,, \\ S_h^i &= \{j \in N_h^i \,:\, |b_{ij}| > \mathbf{coarse}\,(B_h, i, j)\,, i \neq j\}\,, \\ S_h^{i,T} &= \{j \in N_h^i \,:\, i \in S_h^j\}\,, \end{aligned}$$

where N_h^i is the set of neighbors, S_h^i denotes the set of 'strong connections' and $S_h^{i,T}$ is related to the set of nodes, which have a strong connection to node i, respectively. The cut-off (coarsening) function is chosen as, e.g.

$$\mathbf{coarse}\,(B_h, i, j) = \begin{cases} \theta \cdot \sqrt{|b_{ii}||b_{jj}|}\,, & \text{see [34]}\,, \\ \theta \cdot \max_{l \neq i} |b_{il}|\,, & \text{see [29]}\,, \\ \theta\,, & \text{see [20]}\,. \end{cases} \tag{15}$$

with an appropriate $\theta \in [0,1]$. In addition, we define the local sets

$$\omega_C^i = \omega_C^n \cap N_h^i\,, \quad \omega_F^i = \omega_F^n \cap N_h^i \tag{16}$$

and

$$E_h^i = \{(i,j) \in \omega_h^e \,:\, j \in N_h^i\}\,. \tag{17}$$

The coarsening algorithm is described in Alg. 2.

Algorithm 2 Coarsening phase

$\omega_C^n \leftarrow \emptyset, \quad \omega_F^n \leftarrow \emptyset$

while $\omega_C^n \cup \omega_F^n \neq \omega_h^n$ **do**
  $i \leftarrow \textsc{Pick}(\omega_h^n \setminus (\omega_C^n \cup \omega_F^n))$
  if $|S_h^{i,T}| + |S_h^{i,T} \cap \omega_F^n| = 0$ **then**
    $\omega_F^n \leftarrow \omega_h^n \setminus \omega_C^n$
  else
    $\omega_C^n \leftarrow \omega_C^n \cup \{i\}$
    $\omega_F^n \leftarrow \omega_F^n \cup (S_h^{i,T} \setminus \omega_C^n)$
  end if
end while

Therein the function

$$i \leftarrow \textsc{Pick}(\omega_h^n \setminus (\omega_C^n \cup \omega_F^n))$$

returns a node i where the number $|S_h^{i,T}| + |S_h^{i,T} \cap \omega_F^n|$ is maximal.

Example: Let us consider the FE-mesh of Fig. 3. The

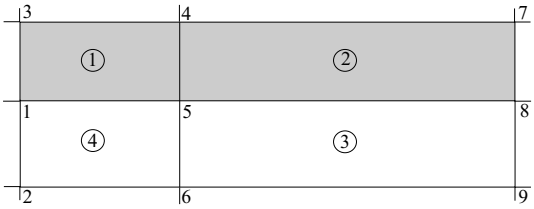

Fig. 3. Example with anisotropic mesh and parameter jump (material 1 in elements with number 1 and 2; material 2 in elements with numbers 3 and 4)

auxiliary matrix is defined on an FE-element r by the setting

$$b_{ij}^r = \frac{\nu_r}{\|\mathbf{a}_{ij}\|_2} \quad i \neq j$$

with ν_r the material parameter and $\mathbf{a}_{ij}$ the geometric edge vector (see (13)). Let us assume the following entries for row 5 of the assembled auxiliary matrix

$$\begin{array}{lllllllll} b_{51} & = & -202 & b_{54} & = & -400 & b_{57} & = & -100 \\ b_{52} & = & -2 & b_{55} & = & 1009 & b_{58} & = & -101 \\ b_{53} & = & -200 & b_{56} & = & -2 & b_{59} & = & -1 \end{array}$$

Using the coarsening function of [29] (see (15)) with $\theta = 0.25$, we obtain

$$\begin{aligned} N_h^5 &= \{1,\ldots,4,6,\ldots,9\} \\ S_h^5 &= \{1,3,4,7,8\} \\ S_h^{5,T} &= \{4,6,7\}\,. \end{aligned}$$

For the construction of set $S_h^{5,T}$ we assumed

$$\begin{array}{lllllll} S_h^1 & = & \{3\} & S_h^6 & = & \{1,2,5,8\} \\ S_h^2 & = & \{1\} & S_h^7 & = & \{4,5,8\} \\ S_h^3 & = & \{1\} & S_h^8 & = & \{7,9\} \\ S_h^4 & = & \{1,3,5,7,8\} & S_h^9 & = & \{7,8\}\,. \end{array}$$

A special coarsening algorithm is the agglomeration technique, where θ is set to 0. Consequently $N_h^i = S_h^i = S_h^{i,T}$ for all $i = 1,\ldots,M_h$. Further we call $(I_h^i)_{i=1}^{M_H}$ ($|\omega_C^n| = M_H < M_h$) a 'disjoint' splitting for the agglomeration method if

$$I_h^i \cap I_h^j = \emptyset, \quad \bigcup_{i=1}^{M_H} I_h^i = \omega_h^n\,,$$

is valid, see Fig. 4.

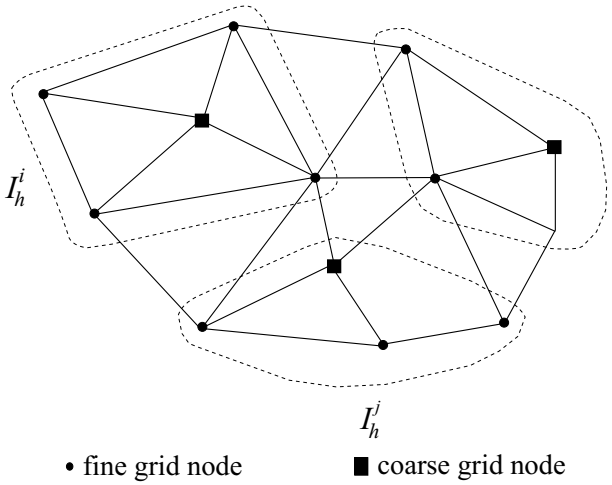

Fig. 4. 'Virtual' FE-mesh with a feasible agglomeration.

If an appropriate prolongation Q_h for B_h is defined then a coarse auxiliary matrix is computed by

$$B_H = (Q_h)^T B_h Q_h$$

and B_H represents again a virtual FE-mesh $\omega_H = (\omega_H^n, \omega_H^e)$, with $\omega_H^n = \omega_C^n$. It can be shown, that B_H is again an M-matrix if the prolongation operator Q_h fulfills certain criteria [29]. Thus the coarsening process can be applied recursively. Finally it is assumed that the degrees of freedom on the coarse grid are numbered first. For instance the nodes are reordered like

$$\omega_h^n = (\omega_C^n, \omega_F^n)$$

(similarly for edges) and as a consequence the system matrix can be written as

$$K_h = \begin{pmatrix} K_{CC} & K_{CF} \\ K_{CF}^T & K_{FF} \end{pmatrix}\,.$$

C. Prolongation Operators

For a given splitting $\omega_h^n = \omega_C^n \cup \omega_F^n$ the optimal prolongation operator is given by the Schur complement approach, i.e.,

$$K_H = K_{CC} - K_{FC}K_{FF}^{-1}K_{CF} = \tilde{P}_h^T K_h \tilde{P}_h$$

with

$$\tilde{P}_h = (I_H, -K_{CF}K_{FF}^{-1})^T\,.$$

The prolongation operator $\tilde{P}_h$ can hardly be realized in practice since $-K_{CF}K_{FF}^{-1}$ involves the inverse of K_{FF}, which in turn implies a global transport of information. In addition, the coarse grid operator K_H becomes dense. The goal of an AMG method is to approximate $\tilde{P}_h$ by some prolongation operator P_h which acts only locally and therewith produces a sparse coarse grid matrix.

D. Smoother and Coarse Grid Operator

An essential point in MG methods is the smoothing operator $S_h \in \mathbb{R}^{N_h \times N_h}$ which reduces the high frequency error components. Typically, a particular smoother works for certain classes of matrices. It is shown in [6] that a point Gauss-Seidel or point Jacobi smoother is appropriate for FE-discretizations with Lagrange FE-functions for scalar elliptic PDEs of second order. Analogously, the block Gauss-Seidel and block Jacobi smoother work well for the block counterpart, e.g. discretization of Maxwell's equation with nodal finite elements. For the edge FE-descritization we use a patch smoother.

The coarse grid operator K_H is usually constructed by Galerkin's method, i.e.,

$$K_H = P_h^T K_h P_h \,. \tag{18}$$

After a successful setup an AMG-cycle can be performed as usual (e.g. see [10]). For instance in Alg. 3 a $V(\nu_F, \nu_B)$-cycle with variable pre- and post-smoothing steps is described. The variable CoarseLevel stores the number of levels generated by the coarsening process until the size of the system is smaller than CoarseGrid.

Algorithm 3 V(ν_F, ν_B)-cycle AMGStep$(K, \underline{u}, \underline{f}, \ell)$

$K_\ell \leftarrow K, \quad \underline{f}_\ell \leftarrow \underline{f}, \quad \underline{u}_\ell \leftarrow \underline{u}$

if $\ell =$ CoarseLevel **then**
  $\underline{u}_\ell \leftarrow$ CoarseGridSolver $(L_\ell L_\ell^T, \underline{f}_\ell)$
  Return
else
  $\underline{d}_\ell \leftarrow 0, \underline{w}_{\ell+1} \leftarrow 0$
  $\underline{u}_\ell \leftarrow S_\ell^{\nu_F}(\underline{u}_\ell, \underline{f}_\ell)$
  $\underline{d}_\ell \leftarrow \underline{f}_\ell - K_\ell \underline{u}_\ell$
  $\underline{d}_{\ell+1} \leftarrow (P_\ell)^T \underline{d}_\ell$
  AMGStep$(K_{\ell+1}, \underline{w}_{\ell+1}, \underline{d}_{\ell+1}, \ell+1)$
  $\underline{w}_\ell \leftarrow P_\ell \underline{w}_{\ell+1}$
  $\underline{u}_\ell \leftarrow \underline{u}_\ell + \underline{w}_\ell$
  $\underline{u}_\ell \leftarrow S_\ell^{\nu_B}(\underline{u}_\ell, \underline{f}_\ell)$
end if

V. Realizations of AMG

In the following subsections we specialize the abstract algorithms, define the components for nodal and edge FE-discretization and additionally propose a method for complex symmetric systems. Prior to that we mention that static, transient and nonlinear analysis of a given problem results in the solution of linear systems (9). Therefore we restrict ourself to the linear analysis. Other applications can be found in [15], [16], [18], [19], [26].

A. AMG for nodal elements

First we consider equation (7) and use nodal elements for discretization. Note that for the following approach the scalar case ($p = 1$) is included (e.g. scalar potential equation).

Construction of 'virtual' FE-meshes: The definition of the auxiliary matrix B_h plays an important role for this problem class. The classical approach uses

$$(B_h)_{ij} = -\|k_{ij}\|_\infty \quad \text{for} \quad i \neq j \,.$$

with $\|\,\|_\infty$ the maximum norm. The diagonal entries of B_h are computed according to (14). Now the degrees of freedom per node of the system matrix have to be related to an entry in the auxiliary matrix, which in turn implies that the matrix pattern of K_h and of B_h has to be equal, i.e.,

$$\|k_{ij}\|_\infty \neq 0 \Leftrightarrow |b_{ij}| \neq 0 \,.$$

Construction of coarse FE-spaces: The simplest prolongation operator is given by

$$(P_h)_{ij} = \begin{cases} I_p & \text{if } i = j \in \omega_C^n \,, \\ \frac{1}{|S_h^{i,T} \cap \omega_C^n|} \cdot I_p & \text{if } i \in \omega_F^n,\ j \in S_h^{i,T} \cap \omega_C^n \,, \\ 0 & \text{else} \,, \end{cases} \tag{19}$$

with $I_p \in \mathbb{R}^{p \times p}$ the p-dimensional identity matrix. The AMG method shows a better convergence behavior as compared to (19) with the subsequent discrete harmonic extension, i.e.,

$$(P_h)_{ij} = \begin{cases} I_p & \text{if } i = j \in \omega_C^n \,, \\ -k_{ii}^{-1}\big(k_{ij} + c_{ij}\big) & \text{if } i \in \omega_F^n,\ j \in \omega_C^i \,, \\ 0 & \text{else} \,, \end{cases} \tag{20}$$

with

$$c_{ij} = \sum_{p \in \omega_F^i} \Big(\sum_{q \in \omega_C^i} k_{pq} \Big)^{-1} k_{ip} k_{pj} \,.$$

However, the increasing memory requirement and the slower application compared to the prolongation (19) is the major drawback of the discrete harmonic extension. Note, that the entries of the prolongation operators are matrix valued, e.g. $(P_h)_{ij} \in \mathbb{R}^{p \times p}$, like the entries of the system matrix K_h.

Smoothing operator: We use a block Gauss-Seidel method as smoothing operator, e.g. [6], or a patch-block Gauss-Seidel method, e.g. [21]. The latter should be used for anisotropic problems.

B. AMG for edge elements

The second class originates from an FE-discretization with edge FE-functions of the variational form (4).

Construction of 'virtual' FE-meshes: It was motivated by R. Hiptmair in [13] for geometric MG methods that for this class of problems a refinement of the FE-mesh can be performed on the nodes of an FE-mesh as it is usually done for Lagrange FE-functions. We use this fact and base our coarsening on an auxiliary matrix B_h which is constructed for instance by the FE-element wise setting

$$b_{ij}^r = -\frac{\nu_r}{\|\mathbf{a}_{ij}\|_2} \quad i \neq j \text{ and } (i,j) \in \omega_h^e \,.$$

with ν_r the reluctivity of the material. Again the diagonal elements are computed via (14).

Example: Let us consider the FE-mesh of Fig. 3 and choose element $r = 1$. We get the following element matrix

$$B_h^1 = 100 \cdot \begin{pmatrix} 2.5 & -0.5 & -1 & 0 \\ -0.5 & 2.5 & 0 & -1 \\ -1 & 0 & 2.5 & -0.5 \\ 0 & -1 & -0.5 & 2.5 \end{pmatrix} .$$

The entries $(B_h^1)_{14}$, $(B_h^1)_{23}$, $(B_h^1)_{41}$ and $(B_h^1)_{32}$ are zero, i.e., there is no diagonal edge in the virtual FE-mesh.

Let us recall that an FE-mesh is represented by

$$\omega_h = (\omega_h^n, \omega_h^e)\,,$$

i.e., the set of nodes ω_h^n and the set of edges ω_h^e. The coarse grid is defined by identifying each coarse grid node $j \in \omega_C^n$ with an index $k \in \omega_H^n$. This is expressed by the index map $\mathbf{ind}\,(.)$ as

$$\omega_H^n = \mathbf{ind}\,(\omega_C^n)\,.$$

A 'useful' set of coarse grid edges ω_H^e can be constructed if we invest in a special prolongation operator Q_h for the auxiliary matrix B_h. The prolongation operator Q_h is constructed such that each fine grid node prolongates exactly from one coarse grid node, so that one arrives at a partition of ω_h^n into clusters, each of them being represented by a coarse grid variable. We extend the index map $\mathbf{ind}\,: \omega_C^n \mapsto \omega_H^n$ defined above onto the whole fine set ω_h^n by assigning to all fine grid nodes of a cluster the coarse grid index of the representative.

$$\mathbf{ind}\,: \omega_h^n \to \omega_H^n\,.$$

A consequence is that $\mathbf{ind}\,(i) = \mathbf{ind}\,(j)$ iff $i, j \in \omega_h^n$ prolongate from the same coarse grid variable. We define an agglomerate (cluster) I_h^i of a grid point $i \in \omega_h^n$ by (see Fig. 5)

$$I_h^i = \{j \in \omega_h^n \,|\, \mathbf{ind}\,(j) = \mathbf{ind}\,(i)\} \subset N_h^i$$

and hence the set of coarse grid nodes can be written as

$$\omega_H^n = \{\mathbf{ind}\,(i) \,|\, i \in \omega_h^n\}\,.$$

The prolongation operator Q_h has only 0 and 1 entries

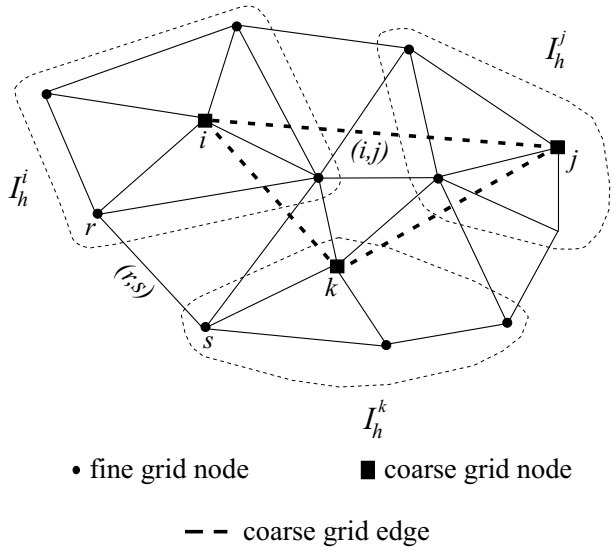

Fig. 5. Virtual FE-mesh with a feasible agglomeration and coarse grid edges

by construction, i.e.,

$$(Q_h)_{ij} = \begin{cases} 1 & i \in \omega_h^n,\ j = \mathbf{ind}\,(i) \\ 0 & \text{otherwise}\,. \end{cases} \qquad (21)$$

Now, a coarse grid edge only exists if there is at least one fine edge connecting the agglomerates I_h^i and I_h^k with $i \neq k$ (see Fig. 5), i.e.,

$$\exists r \in I_h^i, \exists s \in I_h^k \text{ such that } (r,s) \in \omega_h^e\,.$$

Note that a decrease of the number of edges in the coarsening process is not proofed in general, but a decrease is heuristically given, if the average number of nonzero entries of B_h does not grow too fast.

Construction of coarse FE-spaces: The construction of the prolongation operator $P_h : \mathbb{R}^{N_H} \mapsto \mathbb{R}^{N_h}$, is delicate because of the kernel of the **curl**-operator, i.e. all gradient fields. P_h is defined for $i = (i_1, i_2) \in \omega_h^e, j = (j_1, j_2) \in \omega_H^e$ as

$$(P_h)_{ij} = \begin{cases} 1 & \text{if } j = (\mathbf{ind}\,(i_1), \mathbf{ind}\,(i_2)), \\ -1 & \text{if } j = (\mathbf{ind}\,(i_2), \mathbf{ind}\,(i_1)), \\ 0 & \text{otherwise}\,, \end{cases} \qquad (22)$$

by assuming a positive orientation of an edge $j = (j_1, j_2)$ from j_1 to j_2 if $j_1 < j_2$ holds. The constructed prolongation operator P_h has full rank, because the coarse grid edges prolongate to N_H distinct fine grid edges by construction. For a detailed discussion see [27].

Smoothing operator: To complete the components for an AMG method for edge element FE-discretizations, we need an appropriate smoother. We consider two different types of smoothers for K_h. The first one was suggested by D. Arnold, R. Falk and R. Winther in [2]. This is a block Gauss-Seidel smoother where all edges that belong to E_h^i (see (17)) are smoothed simultaneously for all $i \in \omega_h^n$ (see Fig. 6).

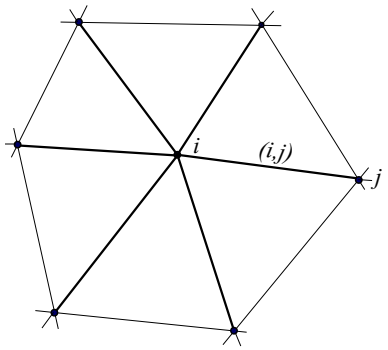

Fig. 6. Detail view of a virtual FE-mesh.

Another kind of smoother was suggested by R. Hiptmair in [13]. A mathematically equivalent formulation is outlined in Alg. 4. Therein the vector $\underline{g}_h^{e,i} \in \mathbb{R}^{N_h}$ is

Algorithm 4 Hybrid smoother of Hiptmair

$\underline{u}_h \leftarrow \textsc{GaussSeidel}(K_h, \underline{f}_h, \underline{u}_h)$

for all $i \in \omega_h^n$ **do**

$\underline{u}_h \leftarrow \underline{u}_h + \frac{\left((\underline{f}_h - K_h \underline{u}_h), \underline{g}_h^{e,i}\right)}{\left(K_h \underline{g}_h^{e,i}, \underline{g}_h^{e,i}\right)} \cdot \underline{g}_h^{e,i}$

end for

defined by

$$\underline{g}_h^{e,i} = \mathbf{grad}_h \underline{g}_h^{n,i} = \begin{cases} 1 & \text{if } j < i \quad (i,j) \in E_h^i\,, \\ -1 & \text{if } j > i \quad (i,j) \in E_h^i\,, \\ 0 & \text{otherwise}\,, \end{cases}$$

with a vector $\underline{g}_h^{n,i} \in \mathbb{R}^{M_h}$, $(\underline{g}_h^{n,i})_j = \delta_{ij}$.

It should be noted, that the proposed AMG solver has been adopted by Mifune et. al. and successfully implemented [22].

C. AMG for Time Harmonic Case

In the harmonic case the time derivative of the magnetic vector potential is substituted by

$$\frac{\partial \mathbf{A}}{\partial t} \rightarrow j\omega \hat{\mathbf{A}}$$

with j the complex number, ω the angular frequency and $\hat{\mathbf{A}}$ the complex magnetic vector potential. Therewith, we have to apply the AMG method to a complex valued and symmetric algebraic system of equations with system matrix

$$K_h = K_h^{re} + jK_h^{im} \,. \tag{23}$$

In (23) K_h^{re} denotes the real part and K_h^{im} the complex part of the system matrix.
The application to scalar potential equations has been presented in [28], and adaption to the magnetic vector potential formulation is straight forward as shown below.

Construction of 'virtual' FE-meshes: The auxiliary matrix is defined to be real valued. This means, that we setup B_h for an edge element discretization as defined in Sec. V-B. For a nodal element discretization we can use the procedure described in Sec. V-A for K_h^{re}.

Construction of coarse FE-spaces: For the construction of a coarse grid operator K_H we define the system prolongation to be real valued and computed as defined in Sec. V-A as well as Sec. V-B. Therefore we get

$$K_H = P_h^T K_h P_h = P_h^T K_h^{re} P_h + jP_h^T K_h^{im} P_h = K_H^{re} + jK_H^{im} \,.$$

The prolongation Q_h is also taken from the real valued realizations correspondingly.

Smoothing operator: In the case of an algebraic system of equations arising from a nodal FE-discretization we apply a block Jacobi or Gauss-Seidel smoother in the complex variant. The complex version of the smoother proposed by D. Arnold, R. Falk and R. Winther in [2] is used for an edge FE-discretization.

VI. Case Studies

In order to gain robustness and efficiency, the proposed AMG methods were used as a preconditioner in the conjugate gradient (CG) method for the static and eddy current case (in time domain) and the Quasi Minimal Residual (QMR) method for the time harmonic case. The iteration was stopped as soon as an error reduction in the preconditioner energy norm has been reduced by a factor 10^{-6} for the PCG method. In the time harmonic case (QMR solver) we use

$$\|\underline{f}_h - K_h \underline{u}_h\|_2 \leq 10^{-6} \, \|\underline{f}_h\|_2 \,.$$

For all calculations a $V(2,2)$-cycle has been applied and the coarsest matrix equation is solved by a Cholesky factorization (degrees of freedom ≤ 500). All computations were done on an PC Pentium 1.7 GHz.

A good measure for the speed of coarsening is the so called **grid complexity**, which is given by

$$GC(K_h) = \frac{\sum_{i=1}^{L} M_i}{M_1} \,, \tag{24}$$

with L the number of levels and M_i the number of nodes (edges) for level i. This number is close to 1, if the reduction of unknowns is done very fast. If the number is very large then the coarsening is usually very slow. A second measure which is more related to the memory consumption and arithmetic costs is the **operator complexity**, i.e.,

$$OC(K_h) = \frac{\sum_{i=1}^{L} NME_i \cdot N_i}{NME_1 \cdot N_1} \,, \tag{25}$$

where NME_i denotes the average number of non-zero entries on level i and N_i the number of unknowns on this level. This number gives an idea of how much memory is used with respect to the finest grid. The same applies for the arithmetic costs. Again this number is close to 1 if only a small amount of memory is required. The abbreviation MB denotes the amount of MegaByte used.
The computations with Lagrange and Nédélec FE-functions were always done on the same FE-mesh. We want to emphasize, that $N_h = p\,|\omega_h^n|$ for node (static case: $p = 3$; eddy current case: $p = 4$) and $N_h = |\omega_h^e|$ for edge FE-discretization.

A. Static Analysis

For the computational domain we consider the geometry of TEAM 20 (see Fig. 7), which has been discretized

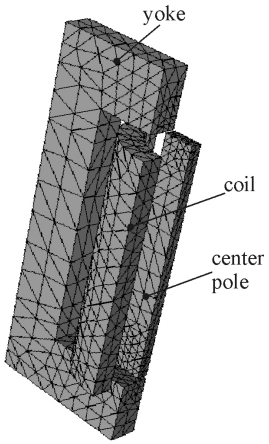

Fig. 7. FE-mesh of TEAM 20 (without air region)

by tetrahedra elements.

Table I displays the evaluated grid complexity GC and operator complexity OC as well as required memory for the nodal (defined by n) and edge (defined by e) case. N_h defines the number of unknowns. It can be clearly seen, that the required memory scales optimal with the number of unknowns and the values for OC and GC are close to 1. The number of iterations as well as elapsed CPU-times are shown in Tab. II. The short time for performing the setup makes the AMG solvers very attractive for any nonlinear problem.

N_h		GC		OC		MB	
n	e	n	e	n	e	n	e
1.263	2.253	1.4	1.2	1.07	1.02	1.5	3
8.022	16.217	1.3	1.2	1.06	1.02	10	24
56.673	122.762	1.3	1.2	1.07	1.03	65	173

TABLE I
TEAM 20: Complexities and memory requirement

N_h		Setup [s]		Solve [s]		Iter	
n	e	n	e	n	e	n	e
1.263	2.253	0.2	0.2	0.2	0.2	18	9
8.022	16.217	0.4	0.5	2.0	2.8	27	16
56.673	122.762	1.7	2.9	30.7	35.3	48	24

TABLE II
TEAM 20: CPU times and number of iterations

B. Transient Analysis

In order to show the performance of the proposed enhanced AMG methods for an eddy current problem, we present results of 3D magnetic field computations for a simplified MRI scanner with z-gradient coil, as shown in Fig. 8 [24]. Here, gradient and magnet coils are assumed as smeared cylindrical coils. Furthermore, only the three inner cryostat cylinders are modeled. Table III displays

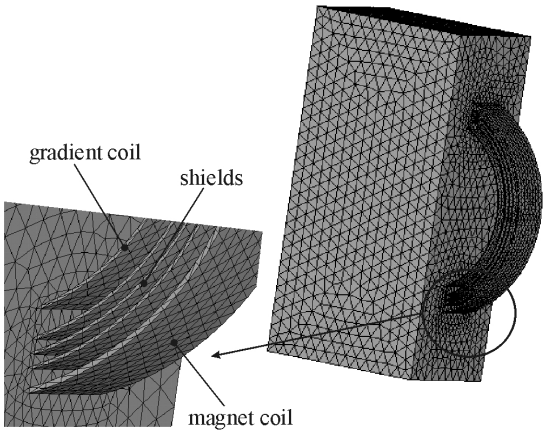

Fig. 8. FE-mesh of a simplified MRI scanner (not the full air region is displayed)

the values for the grid complexity GC, the operator complexity OC and the required memory. Again a very good performance with optimal memory requirement can be found.

N_h		GC		OC		MB	
n	e	n	e	n	e	n	e
27.834	61.342	1.2	1.2	1.06	1.03	75	94
88.053	197.375	1.2	1.2	1.06	1.03	250	330
162.882	368.131	1.2	1.2	1.05	1.03	510	639

TABLE III
MRI-scanner: Complexities and memory requirement

The performance of the proposed AMG solvers concerning number of iterations and CPU time is shown in Tab. IV. Since in this example we have no parameter

N_h		Setup [s]		Solve [s]		Iter	
n	e	n	e	n	e	n	e
27.834	61.342	2.5	1.5	10.3	7.6	15	10
88.053	197.375	7.7	5.8	31.2	38.2	15	15
162.882	368.131	14.6	10.1	64.1	75.2	16	15

TABLE IV
MRI-scanner: CPU times and number of iterations

jump in the reluctivity, the number of iterations keeps quite constant, which results in an optimal convergence rate.

C. Time Harmonic Analysis

Finally we show the performance of the AMG method for the harmonic analysis. We present results of 3D magnetic field computations for an electric transformer as shown in Fig. 9. In Tab. V the values for the grid com-

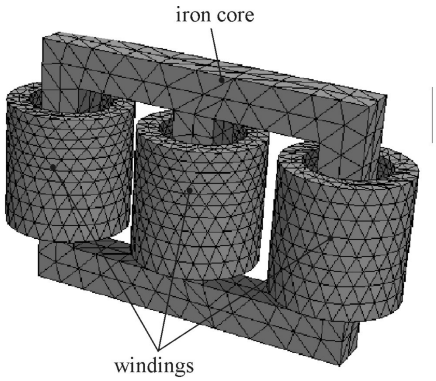

Fig. 9. FE-mesh of an electric transformer (no air region is displayed)

plexity GC, the operator complexity OC, the required memory, number of iterations and the CPU-times can be found. For this example we only used a edge discretization with tetrahedral elements.

N_h	GC	OC	MB	Iter	Setup [s]	Solver [s]
4.786	1.2	1.02	9	19	0.1	1.5
37.390	1.2	1.02	72	30	1.1	22.5
296.064	1.2	1.03	580	51	9.6	319

TABLE V
Electric Transformer: Performance for different FE-meshes

VII. Conclusion

The presented AMG solvers are well suited for the efficient solution – both concerning CPU time and memory requirements – of algebraic systems of equations arising from nodal as well as edge FE discretizations of Maxwell's equation. Especially the presented algorithms for the coarsening strategy make the solvers very attractive also

for nonlinear electromagnetic field problems, since the setup time can be kept very small.

Further research will concentrate on improvements of the prolongation operators to obtain even better convergence rates. One possible method was proposed in [34] (so called smoothed aggregation), which could be applied for our problem classes.

If even more speedup is required for practical applications, then the presented AMG methods can be parallelized on distributed memory computers. First promising results can be found in [11], [12].

VIII. Acknowledgment

This work was supported by 'Deutsche Forschungsgemeinschaft, Sonderforschungsbereich 603 and the Austrian Science Foundation - 'Fonds zur Förderung der wissenschaftlichen Forschung (FWF)' - under the grant P14953 'Robust Algebraic MultiGrid Methods and their Parallelization'.

References

[1] R.A. Adams, *Sobolev Spaces*, Pure and Applied Mathematics, Academic Press, 1975.

[2] D. Arnold, R. Falk, and R. Winther, *Multigrid in H(div) and H(curl)*, Numer. Math. **85** (2000), 197–218.

[3] O. Biró and K. Preis, *On the use of the magnetic vector potential in the finite element analysis of three-dimensional eddy currents*, IEEE Transactions on Magnetics **25** (July 1989), no. 4, 3145–3159.

[4] A. Bossavit and Verite, *The TRIFOU Code: Solving 3-D Eddy Currents Problem by Using H as State Variable*, IEEE Transactions on Magnetics **19** (1983), no. 6, 2465–2470.

[5] D. Braess, *Finite Elemente*, Springer Verlag, Berlin, Heidelberg, New York, 1997.

[6] A. Brandt, *Algebraic multigrid theory: The symmetric case*, Appl. Math. Comput. **19** (1986), 23–56.

[7] W.L. Briggs, V.E. Henson, and S.F. McCormick, *A Multigrid Tutorial*, 2 ed., SIAM, 2000.

[8] M. Costabel and M. Dauge, *Weighted regularization of Maxwell equations in polyhedral domains*, Tech. Report IRMAR 01-26, IRMAR, Institut Mathematique Universite de Rennes 1, 2001.

[9] V. Girault and P.-A. Raviart, *Finite element approximation of the Navier-Stokes equations*, Springer Verlag, Berlin Heidelberg New York, 1979.

[10] W. Hackbusch, *Multigrid methods and application*, Springer Verlag, Berlin, Heidelberg, New York, 1985.

[11] G. Hasse, M. Kuhn, U. Langer, S. Reitzinger, and J. Schöberl, *Parallel Maxwell Solvers*, Scientific Computiong in Electrical Engineering (U. Van Rienen, M. Günther, and D. Hecht, eds.), Lecture Notes in Computational Science and Engineering, vol. 18, Springer, 2000.

[12] G. Hasse, M. Kuhn, and S. Reitzinger, *Parallel AMG on Distributed Memory Computers*, SIAM J. Sci. Comput. **24** (2002), no. 2, 410–427.

[13] R. Hiptmair, *Multigrid method for Maxwell's equations*, SIAM J. Numer. Anal. **36** (1999), no. 1, 204–225.

[14] M. Jung, U. Langer, A. Meyer, W. Queck, and M. Schneider, *Multigrid preconditioners and their application*, Proceedings of the 3rd GDR Multigrid Seminar held at Biesenthal, Karl-Weierstraß-Institut für Mathematik, May 1989, pp. 11–52.

[15] M. Kaltenbacher, H. Landes, S. Reitzinger, and R. Peipp, *3D Simulation of Electrostatic-Mechanical Transducers using Algebraic Multigrid*, IEEE Transactions on Magnetics **38** (2002), no. 2, 985–988.

[16] M. Kaltenbacher and S. Reitzinger, *Algebraic Multigrid for Solving Electromechanical Problems*, Lecture Notes in Computational Science and Engineering, Multigrid Methods VI, Springer-Verlag, September 1999, pp. 129–135.

[17] ———, *Appropriate Finite Element Formulations for 3D Electromagnetic Field Problems*, IEEE Transactions on Magnetics **38** (2002), no. 2, 513–516.

[18] ———, *Nonlinear 3D Magnetic Field Computations using Lagrange FE-functions and Algebraic Multigrid*, IEEE Transaction on Magnetics **38** (2002), no. 2, 1489, 1496.

[19] M. Kaltenbacher, S. Reitzinger, and J. Schöberl, *Algebraic Multigrid Method for Solving 3D Nonlinear Electrostatic and Magnetostatic Field Problems*, IEEE Transactions on Magnetics **36** (2000), no. 4, 1561–1564.

[20] F. Kickinger, *Algebraic multigrid for discrete elliptic second-order problems*, Multigrid Methods V. Proceedings of the 5th European Multigrid conference (W. Hackbusch, ed.), Springer Lecture Notes in Computational Science and Engineering, vol. 3, 1998, pp. 157–172.

[21] F. Kickinger and U. Langer, *A note on the global extraction element-by-element method*, ZAMM (1998), no. 78, 965–966.

[22] T. Mifune, T. Iwashita, and M. Shimasaki, *New Algebraic Multigrid Preconditioner for Iterative Solvers in Electromagnetic Finite Edge-Element Analysis*, Proceedings of the CEFC 2002, 2002.

[23] J. Nédélec, *A new family of mixed finite elements in R^3*, Numer. Math. **50** (1986), 57–81.

[24] M. Rausch, M. Gebhardt, , Kaltenbacher, and H. Landes, *Magnetomechanical Field Computation of a Clinical Magnetic Resonance Imaging (MRI) Scanner*, IGTE Proceedings, 2002.

[25] S. Reitzinger, *Algebraic Multigrid Methods for Large Scale Finite Element Equations*, Reihe C - Technik und Naturwissenschaften, no. 36, Universitätsverlag Rudolf Trauner, 2001.

[26] S. Reitzinger and M. Kaltenbacher, *Algebraic Multigrid Methods for Magnetostatic Field Problems*, IEEE Transactions on Magnetics **38** (2002), no. 2, 477–480.

[27] S. Reitzinger and J. Schöberl, *An Algebraic Multigrid Method for Finite Element Discretizations with Edge Elements*, Numer. Linear Algebra Appl. **31** (2002), no. 3, 223 – 238.

[28] S. Reitzinger, U. Schreiber, and U. van Rienen, *Algebraic Multigrid for Complex Symmetric Matrices: Numerical Study*, Tech. Report 02-01, Johannes Kepler University Linz, SFB "Numerical and Symbolic Scientific Computing", 2002.

[29] J. W. Ruge and K. Stüben, *Algebraic multigrid (AMG)*, Multigrid Methods (S. McCormick, ed.), Frontiers in Applied Mathematics, vol. 5, SIAM, Philadelphia, 1986, pp. 73–130.

[30] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publ. Company, 1996.

[31] M. Schinnerl, J. Schöberl, and M. Kaltenbacher, *Nested Multigrid Methods for the Fast Numerical Computation of 3D Magnetic Fields*, IEEE Transactions on Magnetics **36** (2000), no. 4, 1557–1560.

[32] M. Schinnerl, J. Schöberl, M. Kaltenbacher, and R. Lerch, *Multigrid Methods for the 3D Simulation of Nonlinear Magneto-Mechanical Systems*, IEEE Transactions on Magnetics **38** (2002), no. 3, 1497–1511.

[33] J. A. Stratton, *Electromagnetic Theory*, McGraw-Hill, Inc., 1941.

[34] P. Vanek, J. Mandel, and M. Brezina, *Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems*, Computing **56** (1996), 179–196.

[35] B. Weiß and O. Biró, *Edge Element Multigrid Solution of Nonlinear Magnetostatic Problems*, COMPEL - The International Journal for Computation and Mathematics in Electrical and Electronic Engineering **20** (2001), no. 2, 357–364.

Manfred Kaltenbacher
Department of Sensor Technology
University of Erlangen, Germany
manfred@lse.e-technik.uni-erlangen.de

Stefan Reitzinger
Institute of Computational Mathematics
University of Linz, Austria
reitz@numa.uni-linz.ac.at