Algebraic Multigrid for Static Nonlinear 3D Electromagnetic Field Computations *

Manfred Kaltenbacher[†] Stefan Reitzinger[‡]

20th April 2000

Abstract

In this report the efficiency of algebraic multigrid is shown for static nonlinear electromagnetic field problems, using algebraic multigrid as a preconditioner in the preconditioned conjugate gradient method. The linearization of the nonlinear equation is performed by the fixed-point-scheme.

Because of the relative high setup time of algebraic multigrid the preconditioner is attempted to be kept constant for several nonlinear iteration steps. A practical control mechanism is given such that the condition number of the overall preconditioned system is as small as possible. As a consequence the total computational costs are reduced in terms of CPU time. Numerical studies involving a piezoelectric transducer and a magnetic assembly demonstrate the efficiency of the proposed technique. For comparison the standard incomplete Cholesky preconditioner is used.

Keywords algebraic multigrid, nonlinear electromagnetic field computation, preconditioned conjugate gradient method

1 Introduction

In this paper we are concerned with the efficient solution of static nonlinear electromagnetic field problems. For the electrostatic and the magnetostatic field

*This work has been supported by the Austrian Science Foundation - 'Fonds zur Förderung der wissenschaftlichen Forschung (FWF)' - SFB F013 'Numerical and Symbolic Scientific Computing'

[†]Department of Sensor Technology, University of Erlangen, manfred.kaltenbacher@lse.e-technik.uni-erlangen.de

[‡]Spezialforschungsbereich SFB F013, "Numerical and Symbolic Scientific Computing", Johannes Kepler University Linz, reitz@sfb013.uni-linz.ac.at

problem Maxwell's equations reduce to a scalar, nonlinear, elliptic partial differential equation of second order

$$\operatorname{div}(\nu(u) \cdot \operatorname{grad} u) = f \qquad \text{in } \Omega\,, \tag{1}$$

with appropriate boundary conditions on $\partial\Omega$. Therein $\nu(u) = \nu(|\operatorname{grad} u|)$ is a strictly positive, scalar, bounded function in the computational domain $\Omega \subset \mathbb{R}^d$ $(d = 2, 3)$, u is the solution, and f the right-hand side. A weak formulation of (1) can be formulated in $H^1(\Omega)$ and by using appropriate FE functions we are concerned with the discrete problem

$$K_h(u_h)u_h = f_h\,. \tag{2}$$

Therein $K_h(\cdot) \in \mathbb{R}^{N_h \times N_h}$ denotes the nonlinear stiffness matrix, $f_h \in \mathbb{R}^{N_h}$ the load vector and $u_h \in \mathbb{R}^{N_h}$ the solution vector of the unknown nodal parameters. The number N_h is related to the usual discretization parameter h by the relation $N_h = O(h^{-d})$. The stiffness matrix K_h is supposed to be sparse and symmetric positive definite (SPD). Equation (2) can be solved on the one hand by a Newton type method [9, 10], or on the other hand by the use of a simple fixed-point-scheme [24]. For simplicity we restrict ourself to the fixed-point-scheme in this paper. For both methods a sequence of SPD equations

$$\tilde{K}_h \tilde{u}_h = \tilde{f}_h \tag{3}$$

has to be solved. In the case of the fixed-point-scheme $\tilde{K}_h$ reads as

$$\tilde{K}_h = K_h(u_h^i) \tag{4}$$

where i is the nonlinear iteration index. The spectral condition number $\kappa(\tilde{K}_h)$ of the stiffness matrix $\tilde{K}_h$ typically behaves like $O(h^{-2})$ as the discretization parameter h tends to zero. This is the reason for exhibiting slow convergence of standard iterative solvers.

One way to solve (3) in an efficient way is to use a preconditioned conjugate gradient method (PCG). As it was shown in [11, 12] the most robust solution strategy for SPD problems (3) is PCG with geometric multigrid preconditioner. Geometric multigrid is known to be an iterative solver with optimal time and memory complexity (see e.g. [7]). Unfortunately geometric multigid needs a hierarchical grid and in many cases the grid should be structured. In [9, 10] a Newton type method is suggested (theory and applications) and geometric multigrid is used as a preconditioner for the PCG method. Other applications using geometric multigrid for nonlinear magnetic field problems are given in [21].

In contrast to geometric multigrid algebraic multigrid (AMG) needs no FE-discretization with hierarchical grids but preserves the most advantages of geometric multigrid. That means, AMG is an iterative solver with optimal time and memory complexity which is especially suited for large linear systems with large condition number and a sparse matrix structure. AMG can be applied easily and effectively to problems with complicated geometries and parameter jumps. In addition AMG can be used as a solver in FE-codes which support no hierarchical grid structure. For theory and applications on AMG we refer to [2, 3, 4, 5, 6, 15, 19, 20, 23].

Using AMG as a preconditioner we have to expect relative high setup time (i.e. the construction of the preconditioner, see [4]). As we mentioned above we have to solve several linear equations in order to solve the nonlinear equation (2). According to [8] it is sufficient to use a spectrally equivalent matrix (for a definition on spectral equivalent see Sec. 4) $\tilde{B}_h$ w.r.t. the original system matrix $\tilde{K}_h$ for the PCG method. Thus for solving (3), we apply the preconditioner constructed from $\tilde{B}_h$ for several nonlinear steps i. The convergence rate of the PCG method depends on the spectral equivalence constants which can be controlled via the nonlinearity $\nu(u_h)$. In other words, the management of a new setup is done automatically. In order to get a good AMG preconditioner we use the *element preconditioning technique* introduced in [17]. This technique constructs an M-matrix out of a stiffness matrix by knowing the element stiffness matrices.

The report is organized as follows: After the short introduction the problems to be considered are presented in Sec. 2. In Sec. 3 the fixed-point-scheme is discussed. In every nonlinear step a linear equation has to be solved. This is done by the preconditioned conjugate gradient method with AMG as a preconditioner. In Sec. 4 we show, that a setup has not to be performed in every nonlinear step and therefore saves CPU time. Finally, the results of the numerical studies are presented in Sec. 5. Conclusions and further remarks are drawn in Sec. 6.

2 Problem Formulation

2.1 Electrostatic Field

The electrostatic field is mathematically described by Maxwell's equations for the static case by

$$\begin{aligned} \operatorname{rot} \vec{E} &= 0 \\ \operatorname{div} \vec{D} &= q \\ \vec{D} &= \varepsilon(\vec{E})\vec{E}\,, \end{aligned} \tag{5}$$

with appropriate boundary conditions. In (5), $\vec{E}$ denotes the electric field strength, $\vec{D}$ the electric displacement, q the electric volume charge and ε the electric permittivity. By introducing the scalar electric potential ϕ according to

$$\vec{E} = -\operatorname{grad} \phi\,, \tag{6}$$

the electrostatic field problem can be formulated as follows

$$
\begin{aligned}
-\operatorname{div}(\varepsilon(\vec{E}) \operatorname{grad} \phi) &= q \\
\phi &= \phi_g \text{ on } \Gamma_{\mathrm{e}} \\
-\varepsilon \frac{\partial \phi}{\partial n} &= \sigma_g \text{ on } \Gamma_{\mathrm{n}} .
\end{aligned}
\tag{7}
$$

The boundary conditions are given by

$$
\phi_g = \begin{cases} \phi_0 \text{ on } \Gamma_{\mathrm{e}1} \\ 0 \text{ on } \Gamma_{\mathrm{e}2} \end{cases} \qquad \sigma_g = \begin{cases} \sigma \text{ on } \Gamma_{\mathrm{n}1} \\ 0 \text{ on } \Gamma_{\mathrm{n}2} . \end{cases} \tag{8}
$$

In (7) and for further discussion $\Omega \subset \mathbb{R}^d$ denotes the whole domain with the unit outward vector $\vec{n}$ and the surface $\Gamma = \Gamma_{\mathrm{e}1} \cup \Gamma_{\mathrm{e}2} \cup \Gamma_{\mathrm{n}1} \cup \Gamma_{\mathrm{n}2}$. Therein $\Gamma_{\mathrm{e}1}$ is the surface of given potential ϕ_0 (e.g. potential of voltage loaded capacitor), $\Gamma_{\mathrm{e}2}$ is the surface of zero potential (grounded electrode), $\Gamma_{\mathrm{n}1}$ is the surface of prescribed surface charge σ (on an electrode of a charge loaded capacitor), and $\Gamma_{\mathrm{n}2}$ is the surface of zero normal electric displacement (see Fig. 1). Consequently, we derive

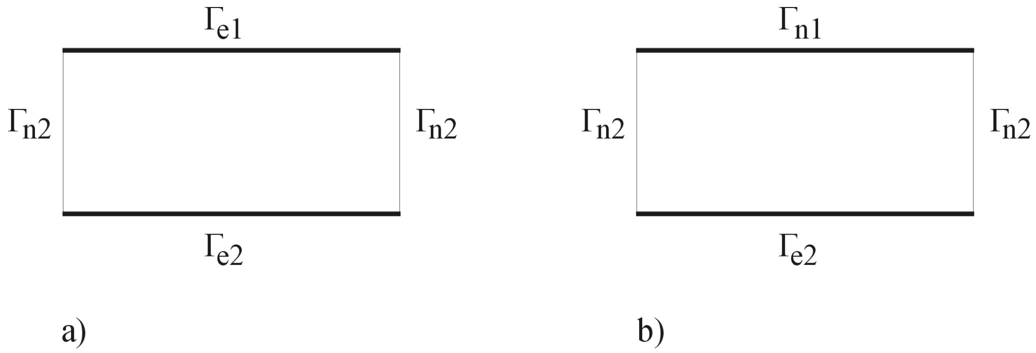

Figure 1: Principle structure of the considered problems.

the variational formulation of (6) in the Sobolev space $H^1(\Omega)$: *Find* $\phi \in H^1(\Omega)$ *such that* $\phi|_{\Gamma_{\mathrm{e}}} = \phi_g$ *and*

$$
a(\phi, v) = \int_{\Omega} q v \, dx \tag{9}
$$

for all $v \in V$, *where*

$$
a(\phi, v) = \int_{\Omega} (\varepsilon(|\nabla \phi|) \ \operatorname{grad} \phi \ \operatorname{grad} v) \ dx + \int_{\Gamma_{\mathrm{n}1}} \sigma v \, ds , \tag{10}
$$

$$
V := \left\{ v \in H^1(\Omega) : v|_{\Gamma_{\mathrm{e}}} = 0 \right\} . \tag{11}
$$

Applying the FE-discretization to (9) leads to the following nonlinear matrix equation

$$K_h^{\phi}(\phi_h)\phi_h = f_h^{\phi} \tag{12}$$

with $K_h^{\phi}(\cdot) \in \mathbb{R}^{N_h \times N_h}$ be SPD, $\phi_h \in \mathbb{R}^{N_h}$ the solution vector of the unknown nodal values and $f_h^{\phi} \in \mathbb{R}^{N_h}$ the load vector. The terms of the electric stiffness matrix $K_h^{\phi}(\cdot)$ are computed by using standard nodal FE-functions.

2.2 Magnetostatic Field

The total magnetic field strength $\vec{H}$ can be defined by

$$\vec{H} = \vec{H}_{\mathrm{s}} + \vec{H}_{\mathrm{m}}\,, \tag{13}$$

where $\vec{H}_{\mathrm{s}}$ describes the magnetic field due to current sources (e.g. calculated according to Biot-Savarts law) and $\vec{H}_{\mathrm{m}}$ denotes the induced magnetism in ferromagnetic materials. Since $\operatorname{rot}\vec{H}_{\mathrm{m}} = 0$ holds, (13) changes to

$$\vec{H} = \vec{H}_{\mathrm{s}} - \operatorname{grad}\psi \tag{14}$$

by introducing the reduced scalar magnetic potential ψ. Using Maxwell's equations for the magnetostatic field results in (see [22])

$$\begin{aligned} \operatorname{div}\left(\mu(\vec{H})\operatorname{grad}\psi\right) &= \operatorname{div}\left(\mu(\vec{H})\vec{H}_{\mathrm{s}}\right) + \operatorname{div}\vec{B}_{\mathrm{r}} \\ \psi &= 0 \text{ on } \Gamma_{\mathrm{e}} \\ \mu\frac{\partial\psi}{\partial n} &= 0 \text{ on } \Gamma_{\mathrm{n}} \end{aligned} \tag{15}$$

with μ being the permeability of the material and $\vec{B}_{\mathrm{r}}$ the remanent flux density (nonzero just in permanent magnets). In addition the following interface condition has to be fulfilled

$$\mu_2\frac{\partial\psi_2}{\partial n} - \mu_1\frac{\partial\psi_1}{\partial n} = \mu_2\vec{H}_{\mathrm{s}}^2\cdot\vec{n} - \mu_1\vec{H}_{\mathrm{s}}^1\cdot\vec{n}\,. \tag{16}$$

Therewith, the variational formulation of the problem in the Sobolev space $H^1(\Omega)$ is derived: *Find* $\psi \in H^1(\Omega)$ *such that* $\psi|_{\Gamma_{\mathrm{e}}} = 0$ *and*

$$a(\psi, v) = \int_{\Omega} \mu(\vec{H})\vec{H}_{\mathrm{s}}\cdot\operatorname{grad}v\,dx + \int_{\Omega}\vec{B}_{\mathrm{r}}\cdot\operatorname{grad}v\,dx \tag{17}$$

for all $v \in V$, *where*

$$a(\psi, v) = \int_{\Omega} \left(\mu(|\nabla\psi|)\ \operatorname{grad}\psi\ \operatorname{grad}v\right)\,dx \tag{18}$$

$$V := \left\{v \in H^1(\Omega) : v|_{\Gamma_{\mathrm{e}}} = 0\right\}\,. \tag{19}$$

The FE-discretization of the magnetostatic field problem yields the following nonlinear matrix equation

$$K_h^{\psi}(\psi_h)\psi_h = f_h^{\psi} \tag{20}$$

with the magnetic stiffness matrix $K_h^{\psi}(\cdot) \in \mathbb{R}^{N_h \times N_h}$ and the load vector $f_h^{\psi} \in \mathbb{R}^{N_h}$. Again the FE-discretization can be done by standard nodal FE-functions.

3 The Fixed-Point-Scheme

The FE-discretization of the nonlinear electrostatic and the magnetostatic problem results in the same equation structure, i.e.

$$K_h(u_h)u_h = f_h\,. \tag{21}$$

Thus the indices ϕ and ψ are omitted for further discussion and the solution vector of unknown nodal values is changed to u_h (see (12) and (20)). Additionally we use the abbreviation $K_h(u_h^i) = K_h^i$. Since the material parameters of the electrostatic (permittivity ε) as well as magnetostatic field equation (permeability μ) depend on the corresponding field quantity (electric field strength $\vec{E}$, magnetic field strength $\vec{H}$), the resulting stiffness matrices given in (12) and (20) are functions of the solution vector. Therefore, we use a standard fix-point-iteration for the solution of (12) and (20)

$$\begin{aligned} K_h^i \Delta u_h &= f_h - K_h^i u_h^i = r(u_h^i) \\ u_h^{i+1} &= u_h^i + \eta \Delta u_h\,. \end{aligned} \tag{22}$$

The nodal vector $r(u_h^i)$ is known as the residual of the problem and a solution is given by the set of nodal values u_h, for which the residual is zero (see [1, 9, 10]). The scalar parameter $\eta > 0$ is introduced to control the convergence during early steps of the iteration process or due to non-monotonic material relations. A common algorithm to compute η is a *line search* (see, i.e., [24]) defined by

$$|G(\eta)| \to \min \tag{23}$$

with

$$G(\eta) = \Delta u_h \cdot r(u_h^i + \eta \Delta u_h)\,. \tag{24}$$

For our problem class it is sufficient to compute the relaxation parameter η as follows:

1. evaluate $g_1 = G(0.1)$ and $g_2 = G(1.0)$,
2. calculate the straight line $l(g_1, g_2)$ between g_1 and g_2,
3. calculate the value $\eta = \frac{10 \cdot g_2 - g_1}{10 \cdot (g_2 - g_1)}$ for which. $l(g_1, g_2) = 0$ holds

4 Preconditioned Conjugate Gradient Method with Algebraic Multigrid

4.1 Notations and Algorithm

We call matrices A and $B \in \mathbb{R}^{m \times m}$ spectrally equivalent if there exists lower and upper constants $\underline{\gamma}$, $\bar{\gamma} \in \mathbb{R}^+$, respectively, such that

$$\underline{\gamma} \cdot \langle Bu, u \rangle \leq \langle Au, u \rangle \leq \bar{\gamma} \cdot \langle Bu, u \rangle \qquad \forall u \in \mathbb{R}^n \tag{25}$$

with $\langle \cdot, \cdot \rangle$ the Euclidian inner product. This is abbreviated by $\underline{\gamma} \cdot B \leq A \leq \bar{\gamma} \cdot B$. Further $\langle Au, u \rangle$ is called the energy inner product for SPD matrices A, and $\|u\|_A^2 = \langle Au, u \rangle$ denotes the energy norm introduced by the energy inner product. The transposed of a matrix A is abbreviated by A^T. Finally the condition number of a SPD matrix A is defined by

$$\kappa(A) = \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)}\,, \tag{26}$$

where $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$ are the largest and the smallest eigenvalues of A, respectively. We use the fact that the condition number $\kappa(B^{-1}A)$ is bounded by

$$\kappa(B^{-1}A) \leq \frac{\bar{\gamma}}{\underline{\gamma}}\,, \tag{27}$$

for SPD matrices $A, B \in \mathbb{R}^{n \times n}$.

As it was mentioned in the last section, the problems to be considered are SPD and therefore a PCG method can be applied. A more detailed description of the PCG method can be found in [8]. The PCG algorithm performs as in Alg. 1. Therein the linear equation

$$K_h u_h = f_h \tag{28}$$

is solved by the PCG method with a SPD preconditioner C_h. The statement "apply q-preconditioning steps" means that q iteration steps of a preconditioner C_h are applied in order to solve the residual equation approximately (typically $q = 1$). It is shown in [11, 12] that multigrid methods applied as preconditioner to the PCG method perform most robust.

Because AMG is a multigrid method we can heuristically conclude that it is a good preconditioner for the PCG method. Unfortunately it can not be proofed in the general case, but many numerical examples showed that AMG is a fast and robust preconditioner (see [2, 6, 15, 19, 23]).

Algorithm 1 Preconditioned conjugate gradient method

$k = 0$
$r^0 = K_h u_h^0 - f_h$
Solve $C_h d^0 = -r^0$,
i.e., apply q-preconditioning steps with an initial zero guess
$s^0 = -d^0$
$r^1 = r^0$
while $\|r^{k+1}\|_{(K_h C_h^{-1} K_h)} > \epsilon \cdot \|r^0\|_{(K_h C_h^{-1} K_h)}$ **do**
  $\alpha^k = \frac{(r^k)^T s^k}{(d^k)^T K_h d^k}$
  $u_h^{k+1} = u_h^k + \alpha^k \cdot d^k$
  $r^{k+1} = r^k + \alpha^k \cdot K_h d^k$
  Solve $C_h d^{k+1} = r^{k+1}$,
  i.e., apply q-preconditioning steps with an initial zero guess
  $\beta^k = \frac{(r^{k+1})^T s^{k+1}}{(r^k)^T s^k}$
  $d^{k+1} = -s^{k+1} + \beta^k \cdot d^k$
  $k = k + 1$
end while

4.2 Solution Strategy

Let us consider a sequence of SPD matrices $(K_h^i)_{i=1,...,m}$ arising during a nonlinear iteration, where m denotes the actual nonlinear iteration step. Further let C_h be an SPD preconditioner for K_h^1 (e.g. $C_h = AMG(K_h^1)$). So far we know that all K_h^i are spectrally equivalent to K_h^1 with lower and upper spectral constants $\underline{\alpha}_i$ and $\bar{\alpha}_i$ $(i = 1, ..., m)$, respectively. The first lemma shows that the preconditioner C_h is also spectrally equivalent to K_h^i $(i = 1, ..., m)$, with a condition number which is a multiple of $\kappa_1 = \kappa(C_h^{-1} K_h^1)$.

Lemma 4.1. *Let $K_h^i \in \mathbb{R}^{N_h \times N_h}$ be SPD for $i = 1, ..., m$, and each K_h^i is spectrally equivalent to K_h^1 with spectral constants $\underline{\alpha}_i$ and $\bar{\alpha}_i$ $(i = 1, ..., m)$, respectively. Further let $C_h \in \mathbb{R}^{N_h \times N_h}$ be a SPD preconditioner for K_h^1 with lower and upper spectral constants $\underline{\gamma}_1$ and $\bar{\gamma}_1$, respectively. Then the following holds*

$$\underline{\alpha}_i \cdot \underline{\gamma}_1 \cdot C_h \leq K_h^i \leq \bar{\gamma}_1 \cdot \bar{\alpha}_i \cdot C_h \tag{29}$$

with $\kappa_i = \kappa(C_h^{-1} K_h^i) = \frac{\bar{\gamma}_1 \cdot \bar{\alpha}_i}{\underline{\gamma}_1 \cdot \underline{\alpha}_i} = \kappa_1 \cdot \frac{\bar{\alpha}_i}{\underline{\alpha}_i}$

Proof. From the assumptions we get

$$\underline{\alpha}_i \cdot K_h^1 \leq K_h^i \leq \bar{\alpha}_i \cdot K_h^1 \tag{30}$$

and straight forward it follows (by the use of spectral equivalence of C_h w.r.t. K_h^1)

$$\underline{\alpha}_i \cdot \underline{\gamma}_1 \cdot C_h \leq K_h^i \leq \bar{\gamma}_1 \cdot \bar{\alpha}_i \cdot C_h\,, \tag{31}$$

which is the desired result. $\square$

Remark 4.2. *In the last lemma the preconditioner is constructed from* K_h^1*, i.e. from the matrix arising in the first nonlinear step. Obviously the preconditioner could also be constructed from* K_h^j $(j > 1)$ *where in such a case the index* j *is redefined to* $j = 1$.

In order to keep the setup costs as small as possible we try to keep the actual preconditioner C_h as long as possible, i.e. m should be large, while the condition number κ_i is bounded by a given constant α.

Next we investigate on the constants $\bar{\alpha}_m$ and $\underline{\alpha}_m$. Thus we assume on the nonlinearity $\nu(x)$ the following properties.

1. ν_h^i is the discrete nonlinearity in the i^{th} nonlinear step
2. $\nu_h^{i,r} \equiv const$ on every element $r \in \tau_h$ where τ_h is the index set of all finite elements

With these assumptions on $\nu(x)$, we get for the spectral constants

$$\underline{\alpha}_i = \min\left\{\min_{r\in\tau_h} \frac{\nu_h^{1,r}}{\nu_h^{i,r}}, 1\right\} \tag{32}$$

$$\bar{\alpha}_i = \max\left\{\max_{r\in\tau_h} \frac{\nu_h^{1,r}}{\nu_h^{i,r}}, 1\right\} \tag{33}$$

and therewith the condition number $\kappa_i = \kappa_1 \cdot \frac{\bar{\alpha}_i}{\underline{\alpha}_i}$. This is easy to see in the case of the considered FE-discretization. The stiffness matrix can be expressed in terms of element stiffness matrices

$$K_h^i = \sum_{r\in\tau_h} A_r^T K_h^{i,r} A_r \tag{34}$$

where $K_h^{i,r} \in \mathbb{R}^{n_r \times n_r}$ is the element stiffness matrix, and $A_r \in \mathbb{R}^{N_h \times n_r}$ is the element connectivity matrix of the finite element $r \in \tau_h$. Because of (9) and (17) the element stiffness matrix has the form

$$K_h^{i,r} = \nu_h^{i,r} \cdot K_h^r\,. \tag{35}$$

As a consequence the spectral constants $\bar{\alpha}_i$ and $\underline{\alpha}_i$ follows immediately. This can be used to control the PCG method. A new setup is performed if the factor $\frac{\bar{\alpha}_i}{\underline{\alpha}_i} > \alpha$, i.e., a new setup is performed if the spectral constants get too large.

Remark 4.3.

1. *The generalization in the case of a tensor valued $\nu(u)$ is straight forward. We assume*

$$\nu_h^{i,r} = \begin{pmatrix} \nu_{11} & \nu_{12} & \nu_{13} \\ \nu_{21} & \nu_{22} & \nu_{23} \\ \nu_{31} & \nu_{32} & \nu_{33} \end{pmatrix} \qquad \forall r \in \tau_h \tag{36}$$

to be SPD. The constants (32) and (33) changes therfore to

$$\underline{\alpha}_i = \min\left\{ \min_{r\in\tau_h} \frac{\lambda_{\min}(\nu_h^{1,r})}{\lambda_{\min}(\nu_h^{i,r})}, 1 \right\} \tag{37}$$

$$\bar{\alpha}_i = \max\left\{ \max_{r\in\tau_h} \frac{\lambda_{\max}(\nu_h^{1,r})}{\lambda_{\max}(\nu_h^{i,r})}, 1 \right\}. \tag{38}$$

2. *In practice $\alpha \in (1,3)$ if the element preconditioning method is additionally used. Otherwise we can take $\alpha \in (1,30)$.*

3. *An important feature of the PCG method is that the error in the l_i^{th} ($i = 1,...,m$) iteration step of the solution can be bounded by*

$$\|x_i^{l_i} - x_i^*\|_{B_h^i} \leq 2 \cdot \left(\frac{\sqrt{\kappa_i} - 1}{\sqrt{\kappa_i} + 1} \right)^{l_i} \cdot \|x_i^0 - x_i^*\|_{B_h^i} \tag{39}$$

with $B_h^i = K_h^i C_h^{-1} K_h^i$, $x_i^{l_i}$ the approximate solution vector of the i^{th} nonlinear step at the l_i^{th} PCG iteration step, x_i^* the exact solution vector of the i^{th} nonlinear step and x_i^0 the start solution of the i^{th} nonlinear step. Thus for a fixed error level ϵ the iteration is stopped if

$$2 \cdot \left(\frac{\sqrt{\kappa_i} - 1}{\sqrt{\kappa_i} + 1} \right)^{l_i} \leq \epsilon\,. \tag{40}$$

In (39), $x_i^0 \equiv 0$ is normally taken as initial guess. But because we are dealing with nonlinear problems we can assume that the iterative solution $x_{i-1}^{l_{i-1}}$ of nonlinear step $i-1$ is a good initial guess for the i^{th} nonlinear step. In practice we see a better performance.

4. *It is important to notice that the convergence rate of the PCG method depends strongly on κ_i and therefore κ_i should not be too large (see [8]).*

5 Numerical Studies

In order to show the performance of the proposed method we present one typical example of each problem class by using the FE-package CAPA [16] and the AMG-package PEBBLES [18]. All calculations where done on a SGI Octane 300 MHz

R12000 workstation. The outer nonlinear iteration stopped if the relative error of the residuum $r(u_h^i)$ is smaller than 10^{-4}. The PCG iteration stopped if the error in the preconditioner energy norm was reduced by a factor of $\epsilon = 10^{-5}$. In each AMG preconditioning step a $V(1,1)$ cycle has been used. For comparison the incomplete Cholesky (IC) preconditioner is presented. Similar computations can be found in [13, 14]. All calculations have been performed by using the element preconditioning method (see [17]). For the following discussion the following short cuts are used.

1. "number of unknowns": size of the stiffness matrix to be solved, i.e. this number is equal to N_h.
2. "CPU time solution": the CPU time for the solution of all nonlinear iteration steps. Therein the CPU time for the assembling process etc. is not included.
3. "linear iterations": the sum of all linear iterations of the PCG method for solving the nonlinear equation.
4. "nonlinear iterations": the number of fixed-point iterations.
5. "number of setups": the number of constructed preconditioners.
6. "AMG: no EP": AMG with no element preconditioning.

5.1 Electrostatic

To study the computational complexity of the developed AMG solver, the numerical calculation of the electric field of a piezoelectric transducer (material PZT 5A) with partial electrodes in the linear and nonlinear case is studied. The electrodes of the transducer are squares with length 10 mm and have a distance of 0.5 mm. The FE-discretization has been performed by brick elements (Fig. 2). The nonlinear function considered for the numerical studies is given in Fig. 3. In the linear case the top electrode of the piezoelectric actuator has been loaded by a voltage. Table 1 presents the CPU-time (seconds) of the overall solution time for the AMG preconditioner and the IC preconditioner as a function of unknowns. We see that already for about 20000 unknowns the AMG preconditioner is faster than the IC preconditioner. Additionally the number of iterations needed for the AMG-PCG is only slowly growing while for the IC-PCG the number of iterations grows quite fast.

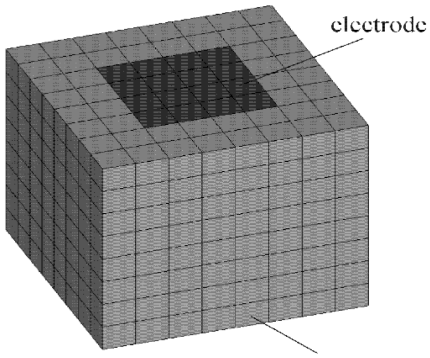

Figure 2: FE-discretization of the piezoelectric transducer

number of unknowns	CPU time solution (sec)			linear iterations		
	AMG	IC	AMG: no EP	AMG	IC	AMG: no EP
18965	5.75	7.73	12.07	18	59	44
67088	23.95	43.01	43.59	21	91	42
147930	69.42	124.73	102.49	28	118	44

Table 1: Electrostatic: Linear case.

number of unknowns	nonlinear iterations	number of setups	CPU time solution (sec)		linear iterations	
			AMG	IC	AMG	IC
18964	5	5	33.57	35.28	111	271
67088	5	5	136.39	197.89	125	418
147930	5	5	348.11	565.42	142	536

Table 2: Electrostatic: nonlinear case with $\alpha = 1$.

In the nonlinear case the top electrode is loaded by an electric charge. To define an equipotential surface constraints are applied to the finite element nodes within the area of the top electrode. The nonlinear case reflects the results of the linear case. This is illustrated in Tab. 2 and Tab. 3. Moreover AMG-PCG is faster if we use the "setup control" by the nonlinearity. In other words, if we can reduce the number of setups in such a way that the overall condition number of the PCG gets not too large, then we end up with a faster method.

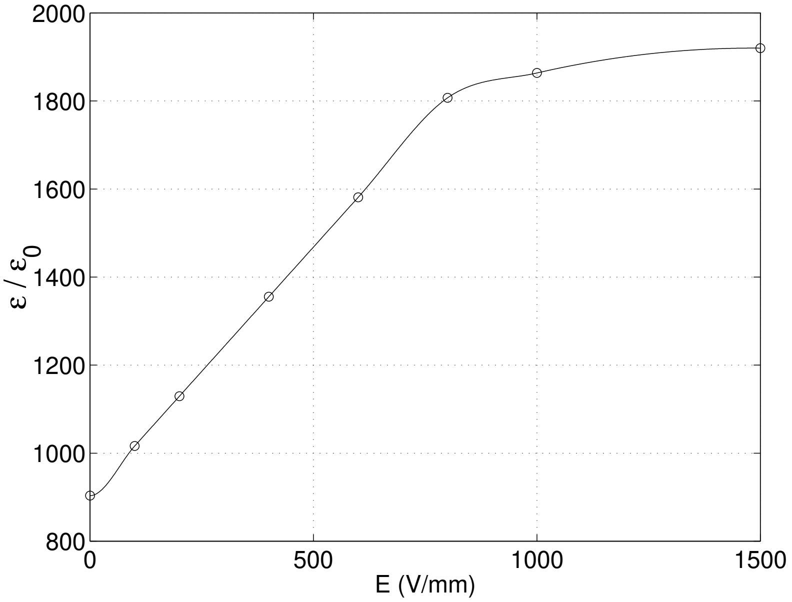

Figure 3: Nonlinear function of the relative permittivity $\varepsilon_r = \frac{\varepsilon}{\varepsilon_0}$.

number of unknowns	nonlinear iterations	number of setups	CPU time solution (sec)		linear iterations	
			AMG	IC	AMG	IC
18965	5	1	30.77	37.88	119	324
67088	5	2	124.28	187.42	125	419
147930	5	2	338.55	546.34	151	541

Table 3: Electrostatic: nonlinear case with $\alpha = 2$.

5.2 Magnetostatic

In the magnetostatic case a magnetic assembly with an air-gap, driven by a permanent magnet, has been considered (Fig. 4). As well as for the piezoelectric transducer brick elements are used for discretization. The considered nonlinearity is given in Fig. 5. Compared to the nonlinearity for the piezoelectric transducer this one is more difficult to handle. As we will see later, the number of nonlinear iterations is larger than for the first example. In the linear case the performance of the AMG and IC preconditioner is shown in Tab. 4. Up to about 50000 unknowns the IC-PCG is faster than the AMG-PCG. However, practical important discretizations are up to 10^6 number of unknowns. The results of the nonlinear case are given in Tab 5, Tab 6 and Tab 7. It is important to notice that the fastest strategy for the nonlinear case is given for $\alpha = 2$.

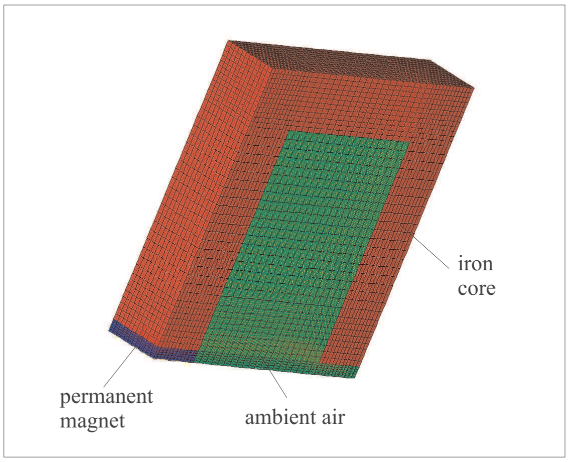

Figure 4: FE-discretization of the magnetic circuit

number of unknowns	CPU time solution (sec)			linear iterations		
	AMG	IC	AMG: no EP	AMG	IC	AMG: no EP
9102	3.86	2.64	3.33	29	42	24
65846	37.03	37.74	39.84	35	82	38
149625	97.00	118.85	88.64	40	112	36

Table 4: Magnetostatic: Linear case.

number of unknowns	nonlinear iterations	number of setups	CPU time solution (sec)		linear iterations	
			AMG	IC	AMG	IC
9102	14	14	43.05	35.01	309	575
65846	15	15	525.15	544.31	503	1194
149625	18	18	1683.18	2021.15	705	1944

Table 5: Magnetostatic: nonlinear case with $\alpha = 1$.

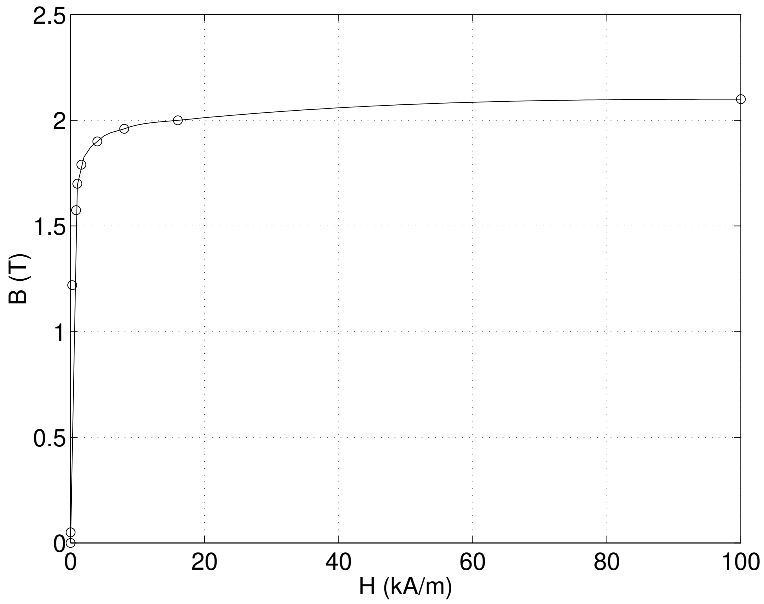

Figure 5: Nonlinear function of the permeability.

number of unknowns	nonlinear iterations	number of setups	CPU time solution (sec)		linear iterations	
			AMG	IC	AMG	IC
9102	14	5	39.73	32.96	315	598
65846	15	5	478.98	532.32	498	1237
149625	18	5	1570.17	2046.85	706	2053

Table 6: Magnetostatic: nonlinear case with $\alpha = 2$.

number of unknowns	nonlinear iterations	number of setups	CPU time solution (sec)		linear iterations	
			AMG	IC	AMG	IC
9102	14	4	41.58	34.10	336	610
65846	15	4	500.49	558.90	524	1294
149625	18	4	1638.70	2249.87	747	2264

Table 7: Magnetostatic: nonlinear case with $\alpha = 4$.

6 Conclusions and Further Remarks

In this paper we have introduced a solution strategy for the numerical simulation of electrostatic and magnetostatic nonlinear field problems. In order to solve a sequence of SPD matrices which results in the fix-point-scheme, the PCG method is used with a new special AMG preconditioner introduced by the second author in [17]. We have shown that the preconditioner for the PCG method can be held constant for several nonlinear iteration steps and this reduces the CPU time compared to conventional preconditioner (e.g. IC). This is of great importance for unstructured grids, parameter jumps, etc.

Further this approach has been realized for time-dependent problems and moving body problems [13].

References

[1] O. Biro, *Numerische Aspekte von Potentialformulierungen in der Elektrodynamik*, Habilitation, TU-Graz, 1992.

[2] D. Braess, *Towards algebraic multigrid for elliptic problems of second order*, Computing **55** (1995), 379–393.

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

[4] A. Brandt, S. McCormick, and J. W. Ruge, *Algebraic multigrid (AMG) for automatic multigrid solution with application to geodetic computations*, Report, Inst. Comp. Studies Colorado State Univ., 1982.

[5] ———, *Algebraic multigrid (AMG) for sparse matrix equations*, Sparsity and its Application (D.J. Evans, ed.), 1984, pp. 257–284.

[6] T. Grauschopf, M. Griebel, and H. Regler, *Additive multilevel preconditioners based on bilinear interpolation, matrix dependent geometric coarsening and algebraic multigrid coarsening for second order elliptic PDEs*, SFB Bericht 342/02/96 A, Technische Universität München, Institut für Informatik, 1996.

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

[8] ———, *Iterative Lösung großer schwachbesetzter Gleichungssysteme*, B.G. Teubner, Stuttgart, 1991.

[9] B. Heise, *Mehrgitter-Newton-Verfahren zur Berechnung Nichtlinearer Magnetischer Felder*, Dissertation, TU-Chemnitz, 1991.

[10] ———, *Analysis of a fully discrete finite element method for a nonlinear magnetic fields problem*, SIAM J. Numer. Anal. **31** (1994), no. 3, 745–759.

[11] M. Jung and U. Langer, *Applications of multilevel methods to practical problems*, Surveys Math. Indust. **1** (1991), 217–257.

[12] 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.

[13] M. Kaltenbacher and S. Reitzinger, *Algebraic multigrid for solving electromechanical problems*, European Multigrid Meeting 1999, Lecture Notes in Computer Science and Engineering, submitted.

[14] M. Kaltenbacher, S. Reitzinger, and J. Schöberl, *Algebraic multigrid for solving 3D nonlinear electrostatic and magnetostatic field problems*, COMPUMAG 1999, IEEE Trans. on Magnetics, submitted.

[15] F. Kickinger, *Algebraic multigrid for discrete elliptic second-order problems*, Technical Report 96-5, Johannes Kepler Universität Linz, Institut für Mathematik, 1997, www.nathan.numa.uni-linz.ac.at/pub/technical_reports/.

[16] R. Lerch, H. Landes, and M. Kaltenbacher, *CAPA User's Guide*, 1999, www.lse.e-technik.uni-erlangen.de/CAPA/index.htm.

[17] S. Reitzinger, *Algebraic mutigrid and element preconditioning II*, Tech. Report 99-18, SFB F013, 1999, submitted.

[18] ———, *PEBBLES - User's Guide*, Special research program SFB F013, 1999, www.sfb013.uni-linz.ac.at.

[19] J. W. Ruge and K. Stüben, *Efficient solution of finite difference and finite element equations*, Multigrid Methods for integral and differential equations (D. Paddon and H. Holstein, eds.), 3, Clarendon Press, Oxford, 1985, pp. 169–212.

[20] ———, *Algebraic multigrid (AMG)*, Multigrid Methods (S. McCormick, ed.), Frontiers in Applied Mathematics, vol. 5, SIAM, Philadelphia, 1986, pp. 73–130.

[21] M. Schinnerl, J. Schöberl, and M. Kaltenbacher, *Nested multigrid methods for the fast numerical computation of 3D magnetic fields*, IEEE Trans. on Magnetics, 2000, submitted.

[22] J. Simkin and C.W. Trowbridge, *On the use of the total scalar potential in the numerical solution of field problems in electromagnetics*, Internat. J. Numer. Methods Engrg. **14** (1979), 423–440.

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

[24] O. C. Zienkiewicz and R.L. Taylor, *The finite element method*, 4 ed., vol. 2, McGraw-Hill, 1991.