Algebraic Multigrid and Element Preconditioning II

Stefan Reitzinger *
Johannes Kepler University Linz
Spezialforschungsbereich SFB F013
"Numerical and Symbolic Scientific Computing"
Freistädter Str. 313, first floor
A-4040 Linz, Austria
e-mail: reitz@sfb013.uni-linz.ac.at

20th October 1999

Abstract

This paper presents a solution strategy for large linear systems with a sparse matrix arising from a finite element discretization of some second order scalar elliptic selfadjoint partial differential equation. Especially if anisotropic elements are used then robustness has to be gained.
To achieve this, the algebraic multigrid method of Ruge/Stüben [14] is used in this paper. This algorithm is robust for M-matrices, but unfortunately the 'region of robustness' between symmetric positive definite M-matrices and general symmetric positive definite matrices is very fuzzy. For this reason the so-called element preconditioning technique is introduced to obtain a spectrally equivalent M-matrix with respect to the original stiffness matrix. This is done by solving a restricted optimization problem. Algebraic multigrid applied to the spectrally equivalent M-matrix instead of the original stiffness matrix is then used as a preconditioner in the conjugate gradient method for solving the original problem.
Numerical studies are presented for a magnetic shielding problem, a boundary layer problem and a loudspeaker which show the numerical robustness of the new preconditioning method.

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

Keywords algebraic multigrid, anisotropic elements, preconditioned conjugate gradient method

AMS Subject Classification 65N22, 65F10

1 Introduction

We will solve a linear system

$$A_h u_h = f_h \,, \tag{1}$$

where we assume $A_h \in \mathbb{R}^{n\times n}$ to be symmetric, positive definite, sparse and A_h has a large condition number. This system matrix A_h arises from a finite element (f.e.) discretization τ_h of some second order scalar elliptic selfadjoint partial differential equation in a domain $\Omega \subset \mathbb{R}^3$. Moreover $u_h \in \mathbb{R}^n$ is the solution vector and $f_h \in \mathbb{R}^n$ is the right hand side.
Equation (1) can be solved on one hand by a simple conjugate gradient method or on the other hand in optimal complexity $O(n)$ by using multigrid (multilevel) methods. As shown in [9, 10] the fastest solution algorithm for (1) uses a conjugate gradient iteration with multigrid preconditioning.
Geometric multigrid methods, see [7, 8, 15], are known to be asymptotically optimal, but they require a hierarchical grid structure which may not always be available. Additionally, the implementation of a geometric multigrid method for anisotropic problems is not that easy. Nevertheless, a multigrid cycle can be set up if algebraic multigrid (AMG) is used. AMG is a technique which preserves most advantages of geometric multigrid methods and needs no hierarchical grid structure. Actually, AMG just needs the matrix A_h and the right hand side f_h. This makes AMG very attractive as black-box solver with the features of geometric multigrid cycling processes.
The first serious approach to AMG was made in 1982 by Brandt et.al. [4] and an improved version can be found in [5]. The algorithm described therein behaves robust against parameter jumps, geometry of the domain, etc., if the underlying matrix is a positive definite M-matrix or if it is at least very close to an M-matrix (i.e. there are only few relatively small positive off-diagonal entries). AMG is very simple in principle. To solve equation (1) by means of a multigrid cycle the interpolation, restriction, coarse grid and smoothing operator have to be defined in an algebraic way. We explain this methodology by means of a two grid method, where h and H denote the fine and coarse grid, respectively. First of all, the smoothing operator is fixed and usually a simple one is taken, e.g. Gauß-Seidel. Next the coarse grid matrix A_H is defined via the Galerkin-approach, i.e. $A_H = I_h^H A_h I_H^h$. In the last formula, the restriction and interpolation operators are denoted by I_h^H and I_H^h, respectively. The additional assumption $I_h^H = (I_H^h)^T$ means that only the interpolation operator I_H^h has to be defined. An appropriate

method for computing the interpolation weights is given in [14]. If the operators are set up, a multigrid cycle can be constructed in the usual way (see [8]). For theory on AMG and numerical studies see [6, 14, 13, 3]. Other interesting AMG approaches can be found in [2, 12, 16].
Sometimes it is favorable to derive a preconditioner from a nearby spectrally equivalent matrix B_h for a multilevel preconditioner. Especially in our case we need an M-matrix for applying AMG efficiently. Thus if we construct B_h such that B_h is an M-matrix then we have a good preconditioner for A_h by applying AMG to B_h instead of A_h. For the construction of a spectrally equivalent M-matrix B_h the element preconditioning method is proposed. Therefore we need access to the element stiffness matrices, and the null-space of all element matrices has to be known.
AMG applied to B_h and used as a preconditioning step in the conjugate gradient method results in a robust and efficient solution strategy.
The paper is organized as follows: First, we present a method to obtain a matrix with M-matrix property which is as close as possible to the original stiffness matrix in the spectral sense (see below the definition of spectrally equivalent). For this purpose we have to solve a restricted optimization problem and therefore the gradient of the objective function and a good initial guess is constructed. This method is called element preconditioning. For the element preconditioning technique, numerical examples are presented which prove the numerical robustness of this method. At the end some conclusions are drawn and further remarks are given.

Notation Throughout the paper s.p.(s)d. is the abbreviation for symmetric positive (semi)definite. Further, $\langle \cdot, \cdot \rangle$ denotes the Euclidian inner product. $A = \{a_{ij}\}_{i,j=1,\dots,n} \in \mathbb{R}^{n \times n}$ s.p.sd. and $B \in \mathbb{R}^{n \times n}$ s.p.sd. are called spectrally equivalent if

$$\exists c_1, c_2 \in \mathbb{R}^+ : c_1 \cdot \langle Au, u \rangle \leq \langle Bu, u \rangle \leq c_2 \cdot \langle Au, u \rangle \qquad \forall u \in \mathbb{R}^n \tag{2}$$

which is denoted by $A \sim B$. Additionally the generalized condition number $\kappa(A^\dagger B)$ is bounded by $\kappa(A^\dagger B) \leq \frac{c_2}{c_1}$, where $A^\dagger$ is the pseudo-inverse of A. The i^{th} entry of a vector $u \in \mathbb{R}^n$ is abbreviated by u_i. A^T denotes the transpose matrix of A and $|A|$ is the rank of the matrix A. Moreover we define the set $Z = \{A \in \mathbb{R}^{n \times n} : a_{ii} > 0, a_{ij} \leq 0\}$. Finally, the null-space of a matrix A is denoted by $N(A)$.

2 Element Preconditioning

In this section we present a method to construct a spectrally equivalent M-matrix, such that the generalized condition number is as small as possible. Previous to

that,we present a method to reduce a s.p.sd. matrix A to a regular one A_{11} to cope with the null-space in the generalized eigenvalue problem and the classification of element matrices introduced later.
A s.p.sd. element matrix can be transformed easily into a regular system if we know its null-space analytically.

Lemma 2.1. *Let $A \in \mathbb{R}^{n\times n}$ s.p.sd., $|N(A)| = m$, $n > m$ and*

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} = Q^{-1} \cdot \begin{pmatrix} A_{11} & 0 \\ 0 & 0 \end{pmatrix} \cdot Q^{-T} \tag{3}$$

where $A_{11} \in \mathbb{R}^{(n-m)\times(n-m)}$, $A_{12} = A_{21}^T \in \mathbb{R}^{m\times(n-m)}$ and $A_{22} \in \mathbb{R}^{m\times m}$. Further $Q^T \in \mathbb{R}^{n\times n}$ is a matrix of the form

$$Q^T = \begin{pmatrix} I & \phi_1 \\ 0 & \phi_2 \end{pmatrix} \tag{4}$$

with $I \in \mathbb{R}^{(n-m)\times(n-m)}$ the identity matrix and

$$\phi = \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} \in \mathbb{R}^{m\times n} \tag{5}$$

a basis of $N(A)$. Then for every s.p.sd. matrix $B \in \mathbb{R}^{n\times n}$ with $N(B) = N(A)$ the generalized eigenvalue problem

$$Au = \lambda Bu \tag{6}$$

can be written as

$$A_{11}v_1 = \lambda B_{11}v_1, \qquad v_2 = 0 \tag{7}$$

with the notation

$$\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = Q^{-T}u \tag{8}$$

and regular submatrices A_{11}, B_{11}.

Proof. If $N(A) = \emptyset$ then choose $Q^T = I$ to finish the proof. If $N(A) \neq \emptyset$ then the transformation (3) changes (6) into

$$Q^{-1} \cdot \begin{pmatrix} A_{11} & 0 \\ 0 & 0 \end{pmatrix} \cdot Q^{-T}u = \lambda Q^{-1} \cdot \begin{pmatrix} B_{11} & 0 \\ 0 & 0 \end{pmatrix} \cdot Q^{-T}u. \tag{9}$$

With notation (8) the result (7) follows immediately. A_{11} and B_{11} are s.p.d. by basic algebra results. $\square$

2.1 The element preconditioning method

In order to get a robust AMG method we propose a technique to produce a spectrally equivalent M-matrix B_h from A_h. This is important because if the non-negative values in matrix A_h get too big, then AMG will lose robustness and efficiency. The M-matrix property of A_h can be lost if quadratic f.e. functions are used instead of linear ones, if long thin rectangles or flat prisms are used etc.. A method to get a spectrally equivalent M-matrix is to exploit more information from the element matrices. A part of the following lemma can already be found in [1].

Lemma 2.2. *Let the f.e. matrix $A_h \in \mathbb{R}^{n\times n}$ be s.p.sd.. Moreover let $A^{(r)}$ be the corresponding s.p.sd. element matrices such that*

$$A_h = \sum_{r\in\tau_h} C_r^T A^{(r)} C_r \,, \tag{10}$$

with C_r connectivity matrices, holds. If we have additionally s.p.sd. matrices $B^{(r)} \in Z$ with

$$B^{(r)} \sim A^{(r)} \qquad \forall r \in \tau_h \tag{11}$$

which are spectrally equivalent with lower and upper spectral constants $c_1^{(r)}$ and $c_2^{(r)}$ in (11), respectively. (Notice: Because of the definition of spectrally equivalent it follows that $N(B^{(r)}) = N(A^{(r)}.)$) Then we get

$$B_h \sim A_h \tag{12}$$

where $B_h = \sum_{r\in\tau_h} C_r^T B^{(r)} C_r$. The lower and upper spectral constants of (12) are $c_1 = \min_{r\in\tau_h} c_1^{(r)}$ and $c_2 = \max_{r\in\tau_h} c_2^{(r)}$, respectively. Additionally, $B_h \in Z$ s.p.sd. and $N(B_h) = N(A_h)$ because of (12).

Proof. It is easy to see that

$$\langle \sum_{r\in\tau_h} C_r^T A^{(r)} C_r u, u\rangle = \sum_{r\in\tau_h} \langle A^{(r)} C_r u, C_r u\rangle \tag{13}$$

and because of (11) we achieve

$$\sum_{r\in\tau_h} c_1^{(r)} \cdot \langle B^{(r)} C_r u, C_r u\rangle \le \sum_{r\in\tau_h} \langle A^{(r)} C_r u, C_r u\rangle \le \sum_{r\in\tau_h} c_2^{(r)} \cdot \langle B^{(r)} C_r u, C_r u\rangle \tag{14}$$

By taking the minimum $c_1 = \min_{r\in\tau_h} c_1^{(r)}$ on the left side and the maximum $c_2 = \max_{r\in\tau_h} c_2^{(r)}$ on the right side, the result follows immediately, i.e.

$$c_1 \cdot \langle B_h u, u\rangle \le \langle A_h u, u\rangle \le c_2 \cdot \langle B_h u, u\rangle \,. \tag{15}$$

Finally, every spectrally equivalent element matrix $B^{(r)} \in Z$ and thus $B_h \in Z$. $\square$

Lemma 2.2 is the theoretical background for Algorithm 1 which returns the best s.p.sd. $B^{(r)}$ from the element matrix $A^{(r)}$ w.r.t. the ratio $\frac{c_2^{(r)}}{c_1^{(r)}}$ of the spectral constants.

Algorithm 1 GeneralSpectralMatrix ($A^{(r)}$)

get an element matrix $A^{(r)}$
if $A^{(r)} \notin Z$ **then**
  $A^{(r)}u = \lambda B^{(r)}u$
  $\lambda_{max} = \max\{\lambda\}$
  $\lambda_{min} = \min\{\lambda : \lambda \neq 0\}$
  calculate $B^{(r)}$ with
  $\frac{\lambda_{max}}{\lambda_{min}} \to \min$
  such that $B^{(r)} \in Z$ and s.p.sd.
else
  $B^{(r)} = A^{(r)}$
end if
return $B^{(r)}$

2.2 Practical usage

In Algorithm 1 a restricted optimization problem has to be solved. Therefore we should find a good initial guess for the solution of the optimization problem and determine the gradient of the objective function.
For reasons of simplicity we restrict ourself to the practical important cases $N(A) = \{\emptyset\}$ and $N(A) = \operatorname{span}\{u \in \mathbb{R}^n : u_i = 1\}$.

Gradient of the objective function

Lemma 2.3. *Let $A \in \mathbb{R}^{n\times n}$ s.p.d. be given, $B \in \mathbb{R}^{n\times n}$ s.p.d. and*

$$Au = \lambda Bu \tag{16}$$

with the maximal and minimal eigenvalue λ_{max} and λ_{min} of the generalized eigenvalue problem (16) and the corresponding eigenvectors u^{max} and u^{min}. The condition number of the preconditioned system is

$$\kappa(A^{-1}B) = \frac{\lambda_{max}}{\lambda_{min}}. \tag{17}$$

Then the gradient of (17) is given by

$$\frac{\partial\kappa(A^{-1}B)}{\partial b_{ij}} = \frac{\lambda_{max}}{\lambda_{min}} \cdot \left(u_j^{max} \cdot u_i^{max} \cdot \lambda_{min} - u_j^{min} \cdot u_i^{min} \cdot \lambda_{max} \right) \quad \forall i,j = 1,\dots,n\,. \tag{18}$$

Proof. First of all, equation (16) is transformed into

$$Cv = \mu v \tag{19}$$

with $C = L^{-1}BL^{-T}$, $\mu = \frac{1}{\lambda}$, $A = LL^T$ and $v = L^T u$. Then equation (19) is multiplied with v^T and so we get

$$\mu = \frac{\langle Cv, v\rangle}{\langle v, v\rangle}. \tag{20}$$

By using the chain rule, (19), and (20) we get

$$\dot{\mu} = \frac{(\langle \dot{C}v, v\rangle + 2\cdot\langle Cv, \dot{v}\rangle)\cdot\langle v, v\rangle - 2\cdot\langle v, \dot{v}\rangle\cdot\langle Cv, v\rangle}{\langle v, v\rangle^2} = \langle \dot{C}v, v\rangle = v_i \cdot v_j \tag{21}$$

with $\dot{\mu} = \frac{\partial\mu(B)}{\partial b_{ij}}$. In the last formula μ is an eigenvalue of (19) and v is the corresponding eigenvector with $\langle v, v\rangle = 1$.
Next we observe (by the definition of μ)

$$\kappa(A^{-1}B) = \frac{\lambda_{max}}{\lambda_{min}} = \frac{\frac{1}{\mu_{min}}}{\frac{1}{\mu_{max}}} = \frac{\mu_{max}}{\mu_{min}} \tag{22}$$

and therefore

$$\frac{\partial\kappa(A^{-1}B)}{\partial b_{ij}} = \frac{\partial}{\partial b_{ij}}\left(\frac{\mu_{max}}{\mu_{min}}\right). \tag{23}$$

With a straight forward calculation by using (21), and (19) we get

$$\frac{\partial\kappa(A^{-1}B)}{\partial b_{ij}} = \frac{\lambda_{max}}{\lambda_{min}}\cdot\left(u_j^{max}\cdot u_i^{max}\cdot\lambda_{min} - u_j^{min}\cdot u_i^{min}\cdot\lambda_{max}\right) \tag{24}$$

which is the desired result. □

Remark 2.4. *Lemma 2.3 gives a result for the gradient of the objective function if the underlying matrices A and B are s.p.d.. If A and B are s.p.sd. with $N(A) = N(B)$ then Lemma 2.1 has to be applied first.*

Initial guess for the optimization problem A good choice for the initial guess is very important for the complexity of the optimization problem. We propose two techniques to get a rather good M-matrix which can be used as initial guess in the optimization problem. First, the case $A \in \mathbb{R}^{n\times n}$, $N(A) = \emptyset$ results in Algorithm 2 and Lemma 2.5 gives the spectral equivalence constants.

Lemma 2.5. *Let $A \in \mathbb{R}^{n\times n}$ s.p.d., $A \notin Z$ and $B \in \mathbb{R}^{n\times n}$ the matrix produced by Algorithm 2. Then $B \in Z$, B is s.p.d. and $B = A + M$ with*

$$M = \begin{cases} m_{ii} \geq 0 & i = 1, ..., n \\ m_{ij} \leq 0 & i \neq j \end{cases} \tag{25}$$

and M is weakly diagonally dominant, i.e. $\sum_{j=1,...,n} m_{ij} \geq 0$ $\forall i = 1, ..., n$. Moreover B is spectrally equivalent to A with lower and upper constants $c_1 = \frac{\mu_{min}(A)}{\mu_{min}(A)+\mu_{max}(M)}$ and $c_2 = 1$, respectively. Again, $\mu_{max}(A)$ and $\mu_{min}(A)$ is the maximal and minimal eigenvalue of A.

Algorithm 2 StartMatrix for s.p.d. $(A^{(r)})$

get the element matrix $A^{(r)} \in \mathbb{R}^{n\times n}$
initialize $A_{start} = 0$
for all i=1,n **do**
  for all j=i,n **do**
    if $a_{ij} > 0$ and $i \neq j$ **then**
      update $a_{ii}^{start} = a_{ii}^{start} + \frac{a_{ij}}{2}$
      update $a_{jj}^{start} = a_{jj}^{start} + \frac{a_{ij}}{2}$
    else
      update $a_{ij}^{start} = a_{ij}$
    end if
  end for
end for
return $B = A_{start}$

Proof. First of all it is clear that $B \in Z$ and B is symmetric. Next, we investigate the spectrally equivalent constants. Because $A \notin Z$ there exists at least one positive off diagonal element. Therefore $M \neq 0$, i.e. M can not be the zero matrix. Additionally, the matrix B produced by Algorithm 2 can be written as $B = A + M$, with (25) and by construction M is weakly diagonally dominant. This is easy to see because for every positive off diagonal element a_{ij} Algorithm 2 updates the diagonal elements m_{ii} and m_{jj} with $\frac{a_{ij}}{2}$ and the off diagonal entries m_{ij} and m_{ji} with the value $-\frac{a_{ij}}{2}$. So we get by construction $\sum_{j=1,\dots,n} m_{ij} = 0$ $\forall i = 1, ..., n$. Therefore

$$\langle Mu, u\rangle \geq 0 \qquad \forall u \in \mathbb{R}^n . \tag{26}$$

Thus, on one hand we get

$$\langle Au, u\rangle \leq \langle Au, u\rangle + \langle Mu, u\rangle = \langle Bu, u\rangle \qquad \forall u \in \mathbb{R}^n \tag{27}$$

and on the other hand

$$\langle Bu, u\rangle = \langle Au, u\rangle + \langle Mu, u\rangle = \langle Au, u\rangle \cdot \left(1 + \frac{\langle Mu, u\rangle}{\langle Au, u\rangle}\right) \qquad \forall u \in \mathbb{R}^n \tag{28}$$

resulting in

$$\langle Bu, u\rangle \leq \langle Au, u\rangle \cdot \left(1 + \frac{\mu_{max}(M)}{\mu_{min}(A)}\right) \qquad \forall u \in \mathbb{R}^n . \tag{29}$$

By defining $c_1 = \frac{\mu_{min}(A)}{\mu_{min}(A)+\mu_{max}(M)}$ and $c_2 = 1$ the result follows. Actually we have shown that B is s.p.d. and thus B is an admissible initial guess for the optimization problem. □

The second case, i.e. $A \in \mathbb{R}^{n\times n}$ s.p.sd., $N(A) = \text{span}\{u \in \mathbb{R}^n : u_i = 1\}$, requires some extra work. We define $\psi^{ij} = \{\psi_k^{ij}\}_{k=1,\dots,n} \in \mathbb{R}^n$ by

$$\psi_k^{ij} = \begin{cases} 1 & \text{if k = i} \\ -1 & \text{if k = j} \\ 0 & \text{else}\,. \end{cases} \tag{30}$$

Actually ψ^{ij} and $\psi^{ij} \cdot \psi^{ij^T}$ look like

$$\psi^{ij} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ -1 \\ 0 \end{pmatrix} \begin{matrix} \\ \leftarrow i \\ \\ \leftarrow j \\ \\ \end{matrix} \in \mathbb{R}^n \qquad \psi^{ij} \cdot \psi^{ij^T} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{n\times n}\,. \tag{31}$$

Next we define

$$A_{ij} = \begin{pmatrix} a_{ii} & a_{ij} \\ a_{ij} & a_{jj} \end{pmatrix} \qquad \forall i,j = 1,\dots n \quad i \neq j \tag{32}$$

with $A_{ij} \in \mathbb{R}^{2\times 2}$ and a permutation matrix $P_{ij} \in \mathbb{R}^{n\times n}$ such that

$$\tilde{A} = P_{ij}^T A P_{ij} = \begin{pmatrix} A_{ij} & C_{ij}^T \\ C_{ij} & D_{ij} \end{pmatrix} \tag{33}$$

where $C_{ij} \in \mathbb{R}^{2\times(n-2)}$ and $D_{ij} \in \mathbb{R}^{(n-2)\times(n-2)}$. Moreover, a Schur complement $S_{ij} \in \mathbb{R}^{2\times 2}$ is defined which is given by

$$S_{ij} = A_{ij} - C_{ij}^T D_{ij}^{-1} C_{ij} = \begin{pmatrix} \beta_{ij} & -\beta_{ij} \\ -\beta_{ij} & \beta_{ij} \end{pmatrix} \tag{34}$$

with a value $\beta_{ij} \in \mathbb{R}^+$. The regularity of D_{ij} follows by basic algebra results.
With those definitions Algorithm 3 can be written and Lemma 2.6 gives the spectral constants.

Lemma 2.6. *Let $A \in \mathbb{R}^{n\times n}$ s.p.sd., $A \notin Z$ and $B \in \mathbb{R}^{n\times n}$ the matrix produced by Algorithm 3. Then $B \in Z$ and B is s.p.sd.; moreover $B \sim A$ with lower and upper constants $c_1 = \frac{1}{n(n-1)}$ and $c_2 = \frac{1}{\min_{i,j;i\neq j}\{\frac{\beta_{ij}}{\alpha_{ij}}:\beta_{ij}<\alpha_{ij}\}}$, respectively, and $\alpha_{ij} = \frac{a_{ij}}{2}$.*

Proof. Algorithm 3 results in

$$B = \sum_i \sum_{j\neq i} \beta_{ij} \cdot \psi_{ij} \cdot \psi_{ij}^T \tag{35}$$

Algorithm 3 StartMatrix for s.p.sd. $(A^{(r)})$

get the element matrix $A^{(r)} \in \mathbb{R}^{n\times n}$
initialize $A_{start} = 0$
for all i=1,n **do**
  for all j=1,n **do**
    if $i \neq j$ **then**
      calculate the Schur complement S_{ij} and get parameter β_{ij}
      update matrix $A_{start} = A_{start} + \beta_{ij} \cdot \psi^{ij} \cdot \psi^{ij^T}$
    end if
  end for
end for
return $B = A_{start}$

and it follows that

$$\langle Bu, u\rangle = \sum_i \sum_{j\neq i} \beta_{ij} \cdot (u_i - u_j)^2 \geq 0 \qquad \forall u \in \mathbb{R}^n \tag{36}$$

because of $\beta_{ij} > 0$. Moreover it is easy to see that

$$N(B) = \text{span}\{u \in \mathbb{R}^n : u_i = 1\} = N(A) \tag{37}$$

Therefore B is an admissible initial guess. Now, we determine the spectrally equivalence constants. We observe, that

$$\langle \beta_{ij} \cdot \psi_{ij} \cdot \psi_{ij}^T u, u\rangle \leq \langle Au, u\rangle \qquad \forall u \in \mathbb{R}^n\ . \tag{38}$$

This is because the Schur-complement produces an orthogonal splitting of $\mathbb{R}^n$ w.r.t. to the energy inner product, i.e. $\langle Au, u\rangle$. The necessary transformation is given by

$$T_{ij} = \begin{pmatrix} I \\ -D_{ij}^{-1} C_{ij} \end{pmatrix} \in \mathbb{R}^{2\times n} \tag{39}$$

and therefore the Schur-complement (34) can be written as

$$S_{ij} = T_{ij}^T \tilde{A} T_{ij}\ . \tag{40}$$

Now, every vector $u \in \mathbb{R}^n$ can be expressed as $u = T_{ij}w + v$ with $w \in \mathbb{R}^2$ and $v \in \mathbb{R}^n$, such that $\langle AT_{ij}w, v\rangle = 0$. By using $\langle \tilde{A}u, u\rangle = \langle Au, u\rangle$ we get

$$\langle \tilde{A}u, u\rangle = \langle \tilde{A}(T_{ij}w + v), T_{ij}w + v\rangle = \langle T_{ij}^T \tilde{A} T_{ij} w, w\rangle + \langle \tilde{A}v, v\rangle + 2 \cdot \langle \tilde{A} T_{ij} w, v\rangle \tag{41}$$

and with the orthogonality and the semidefinitness of A it follows

$$\langle Au, u\rangle = \langle \tilde{A}u, u\rangle \geq \langle T_{ij}^T \tilde{A} T_{ij} w, w\rangle = \langle S_{ij} w, w\rangle \tag{42}$$

which is the desired result. Finally we get with (35), (38), and (42)

$$\langle Bu, u\rangle \leq n(n-1) \cdot \langle Au, u\rangle \qquad \forall u \in \mathbb{R}^n . \tag{43}$$

Next, we can express $A = \sum_i \sum_{j \neq i} \alpha_{ij} \cdot \psi_{ij} \cdot \psi_{ij}^T$, with $\alpha_{ij} = \frac{a_{ij}}{2}$. Thus we get by a component comparison

$$\langle \beta_{ij} \cdot \psi_{ij} \cdot \psi_{ij}^T u, u\rangle \geq c_h \cdot \langle \alpha_{ij} \cdot \psi_{ij} \cdot \psi_{ij}^T u, u\rangle \qquad \forall u \in \mathbb{R}^n \tag{44}$$

with $c_h = \min_{i,j;i\neq j}\{\frac{\beta_{ij}}{\alpha_{ij}} : \beta_{ij} < \alpha_{ij}\}$. The proof ends by defining $c_1 = \frac{1}{n(n-1)}$ and $c_2 = \frac{1}{\min_{i,j;i\neq j}\{\frac{\beta_{ij}}{\alpha_{ij}} : \beta_{ij} < \alpha_{ij}\}}$. □

2.3 How good is this technique?

It is clear that every s.p.sd. matrix $A \in \mathbb{R}^{n\times n}$ has a s.p.sd. spectrally equivalent matrix $B \in \mathbb{R}^{n\times n}$, but how close is the ratio $\frac{c_2}{c_1}$ of spectral constants to 1? Unfortunately there exists an example where the generalized condition number $\kappa(A^\dagger B)$ is not uniformly bounded.
Let us consider the Laplace operator with linear f.e. functions on the triangle in Figure 1. The element stiffness matrix has the form

$$A = \frac{1}{4q}\begin{pmatrix} 1+q^2 & 1-q^2 & -2 \\ 1-q^2 & 1+q^2 & -2 \\ -2 & -2 & 4 \end{pmatrix} . \tag{45}$$

For $0 < q < 1$ $A \notin Z$ and the best matrix $B \in Z$ in the sense of Algorithm 1 is

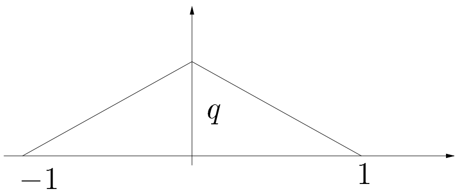

Figure 1: A long thin triangle.

$$B = \frac{1}{4q}\begin{pmatrix} 2 & 0 & -2 \\ 0 & 2 & -2 \\ -2 & -2 & 4 \end{pmatrix} . \tag{46}$$

Solving the generalized eigenvalue problem we get

$$\kappa(A^{\dagger}B) \geq \frac{1}{q^2}. \tag{47}$$

Thus, if q gets small enough then the generalized condition grows very fast. Anyway, in practice it is not advisable to use such elements for an f.e. discretization because of their poor approximation property.
Nevertheless this technique is successful in a lot of important cases, where normal AMG breaks down (see Section 4).

3 Classification

In many practical cases the element matrices $A^{(r)} \in \mathbb{R}^{n\times n}$, $r \in \tau_h$, are, up to a factor, similar in the sense of spectral equivalence; i.e. only some of the element matrices have to be optimized. Practically the following has to be done: First, we classify which element matrices are similar in the sense of spectral equivalence. Then one matrix of each class is optimized and additionally the remaining matrices of this class get the same spectrally equivalent matrix. This trick saves a lot of computational work. The next lemma shows that this technique works.
First, we rescale every element matrix in the following way

$$\tilde{A}^{(r)} = \frac{1}{\max_{i=1,\dots,n}\{a_{ii}\}} \cdot A^{(r)} \qquad \forall r \in \tau_h. \tag{48}$$

Lemma 3.1. *Let $A_1, A_2 \in \mathbb{R}^{n\times n}$ s.p.d. scaled as in (48) with*

$$A_2 = A_1 + \Delta \tag{49}$$

where $\Delta \in \mathbb{R}^{n\times n}$ is symmetric. Moreover let $B \in \mathbb{R}^{n\times n}$, $B \in Z$ s.p.d and $B \sim A_1$ with lower and upper constants c_1 and c_2, respectively. If there exist $\delta, \lambda_1 \in \mathbb{R}^+$

$$-\delta \leq \frac{\langle \Delta u, u\rangle}{\langle u, u\rangle} \leq \delta \qquad \forall u \in \mathbb{R}^n \tag{50}$$

$$\lambda_1 \leq \frac{\langle Bu, u\rangle}{\langle u, u\rangle} \qquad \forall u \in \mathbb{R}^n \tag{51}$$

such that $c_1 > \frac{\delta}{\lambda_1}$ holds, then we get

$$\left(c_1 - \frac{\delta}{\lambda_1}\right) \cdot \langle Bu, u\rangle \leq \langle A_2 u, u\rangle \leq \left(c_2 + \frac{\delta}{\lambda_1}\right) \cdot \langle Bu, u\rangle \qquad \forall u \in \mathbb{R}^n . \tag{52}$$

Proof. Starting with

$$c_1 \cdot \langle Bu, u\rangle \le \langle A_1 u, u\rangle \le c_2 \cdot \langle Bu, u\rangle \qquad \forall u \in \mathbb{R}^n \tag{53}$$

we get

$$c_1 \cdot \langle Bu, u\rangle + \langle \Delta u, u\rangle \le \langle A_2 u, u\rangle \le c_2 \cdot \langle Bu, u\rangle + \langle \Delta u, u\rangle \qquad \forall u \in \mathbb{R}^n \tag{54}$$

or equivalently

$$\langle Bu, u\rangle \cdot \left(c_1 + \frac{\langle \Delta u, u\rangle}{\langle Bu, u\rangle}\right) \le \langle A_2 u, u\rangle \le \langle Bu, u\rangle \cdot \left(c_2 + \frac{\langle \Delta u, u\rangle}{\langle Bu, u\rangle}\right) \qquad \forall u \in \mathbb{R}^n . \tag{55}$$

With assumptions (50), (51) we can conclude the proof. □

Remark 3.2. *Lemma 3.1 gives a result for the classification of s.p.d. matrices. Dealing with s.p.sd. matrices A and B, then A and B can only be in the same class if $N(A) = N(B)$. If this is the case, A and B are transformed into (3) and hence we can work with the regular parts of A and B.*
Notice: With the conditions $c_1 > \frac{\delta}{\lambda_1}$ and (51) we get with (50) a necessary condition which $\Delta \in \mathbb{R}^{n\times n}$ is admissible.

4 Numerical Studies

In this section the efficiency and robustness of the new technique is shown. Therefore the code was included in the program packages FEPP and CAPA. Therein AMG is used as a preconditioner for the conjugate gradient method.
A potential equation, typically arising in magnetic field problems, a boundary layer problem and a loudspeaker are treated as examples. The overall abbreviations for this section are listed in Table 1.

PCAMG	preconditioned CG method with AMG
ModPCAMG	preconditioned CG method with element preconditioning and AMG
n	number of unknowns
level	number of levels used in the calculation
κ	condition number of $C_h^{-1}A_h$,
	where C_h is the preconditioner, i.e. AMG applied to B_h
iter	number of iterations of the preconditioned CG method

Table 1: Abbreviations for the numerical studies.

We used in an AMG step a generalized V-cycle for the shielding problem and for the boundary layer problem. For the loudspeaker a V-cycle is used. For the shielding and boundary layer examples the iteration was terminated if an error reduction in preconditioner energy norm by a factor of 10^{-8} has been achieved. These calculations were done on an SGI Octane workstation R10000, 225 MHz. For the electrodynamic loudspeaker the iteration was terminated if an error reduction in preconditioner energy norm by a factor of 10^{-6} has been achieved. Additionally this problem was calculated on an SGI Octane workstation R10000, 195 MHz.

4.1 Shielding problem in 2D

Let us consider the following problem

$$
\begin{aligned}
-\operatorname{div}(\mu_i \operatorname{grad} u(x)) &= 0 \qquad \forall x \in \Omega_i \qquad i = 1,2 && (56)\\
u(x) &= -\mu_1 \cdot (B_x x_1 + B_y x_2) \qquad \forall x \in \partial\Omega_1 \setminus \partial\Omega_2 && (57)\\
-\mu_1 \frac{\partial u_1(x)}{\partial n} &= -\mu_2 \frac{\partial u_2(x)}{\partial n} \qquad \forall x \in \partial\Omega_2 && (58)
\end{aligned}
$$

with parameter settings $\mu_1 = 1$, $\mu_2 = 1000$, $B_x = 21$, $B_y = 45$ and domain Ω as shown in Figure 2. As we can see in Figure 2, the domain has a geometric anisotropy, which is discretized by long thin quadrilaterals. If we would discretize the shielding unit with a graded mesh, e.g. with triangles, then the number of unknowns would explode be very large. Thus we keep the number of unknowns small by using long thin quadrilaterals.

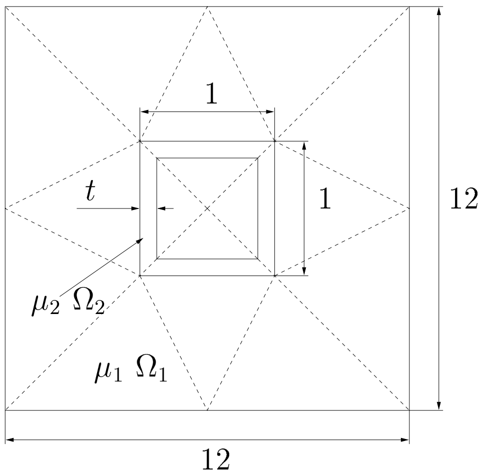

Figure 2: Shielding problem with two materials.

In Table 2, the condition numbers and the number of iterations are shown for a 2D problem using element preconditioning. For comparison the non robust PCAMG method is presented for $t = 10^{-3}$ in Table 3.

	ModPCAMG			
t	n	level	κ	iter
10^{-2}			5.59	25
10^{-3}	2369	3	5.42	24
10^{-4}			4.93	22
10^{-5}			4.31	20
10^{-2}			6.58	29
10^{-3}	9345	4	6.57	28
10^{-4}			6.05	27
10^{-5}			5.14	24
10^{-2}			8.91	33
10^{-3}	37121	6	9.26	34
10^{-4}			7.96	31
10^{-5}			6.60	28

Table 2: Shielding problem in 2D with element preconditioning.

	PCAMG			
t	n	level	κ	iter
	2369	3	1401	>100
10^{-3}	9345	4	927	>100
	37121	6	625	>100

Table 3: Shielding problem in 2D without element preconditioning.

The numerical work for the element preconditioning part is negligible for the shielding problem because of the classification strategy we have to find a spectrally equivalent matrix in the set Z only for ~ 16 element stiffness matrices.

4.2 Boundary layer problem in 2D

Second, we consider a boundary layer problem

$$
\begin{aligned}
-\epsilon^2 \Delta u + u &= 1 \qquad \forall x \in \Omega && (59)\\
u &= 0 \qquad \forall x \in \Gamma && (60)
\end{aligned}
$$

with an appropriate discretization presented in Figure 3 (bilinear f.e. functions). Again, if we use a graded mesh, e.g. with triangles, for resolving the boundary layer then the number of unknowns would be very large.

In Table 4 the numerical robustness of the ModPCAMG can be seen. If we use normal AMG, then κ grows very fast if ϵ gets small.

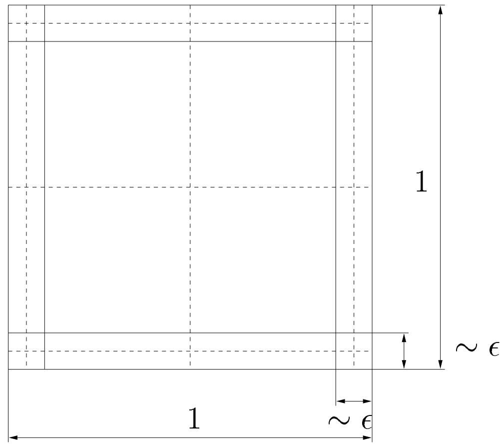

Figure 3: Boundary layer problem.

In the case of the boundary layer problem the numerical work for the element preconditioning method is to construct a spectrally equivalent matrix for 3 different element matrices.

4.3 The loudspeaker

Finally, we present the electrodynamic loudspeaker (see Figure 4). Actually this is a time dependent coupled field problem, but we solve only the magnetic part with ModPCAMG. The mechanical-acoustic part is solved via a direct method, because the stiffness matrix remains constant during the time steps. The magnetic part is changing in every time-step. For a detailed discussion see [11].

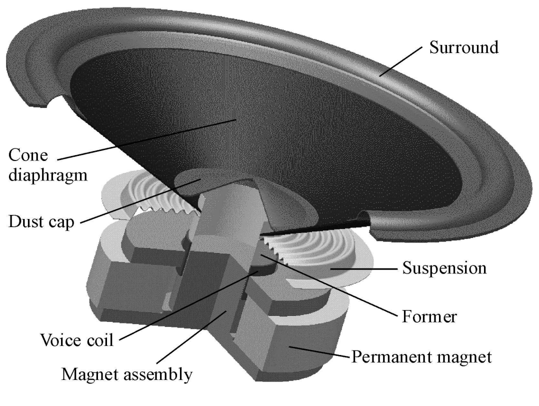

Figure 4: The principle structure of the electrodynamic loudspeaker.

The discretization of the computational domain was done by rectangles, with maximal ratio of the longest to the shortest side of about 20 (see Figure 5). The

	ModPCAMG			
ϵ	n	level	κ	iter
10^{-2}			2.98	13
10^{-3}	625	2	3.05	12
10^{-4}			2.94	11
10^{-5}			2.92	11
10^{-2}			3.08	13
10^{-3}	2401	4	4.61	13
10^{-4}			3.11	12
10^{-5}			3.14	12
10^{-2}		5	3.18	13
10^{-3}	9409	8	3.17	13
10^{-4}		8	4.24	15
10^{-5}		8	3.25	11

Table 4: Boundary layer problem.

number of unknowns for the magnetic, mechanical and acoustic parts are 14738, 31071 and 154191, respectively. Moreover quadratic f.e. functions are used.

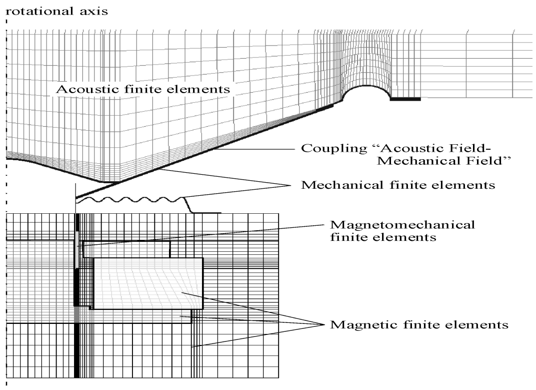

Figure 5: The f.e. discretization of the electrodynamic loudspeaker.

Table 5 presents the convergence rate ρ_{PCG} of the PCG-method with and without element preconditioning. In addition the total CPU-time t_{CPU} for the PCG-method and a direct solver is given.

We have to perform 5000 time steps for the transient analysis of the electrodynamic loudspeaker (one time step 20 ns) in order to get the desired accuracy; the preconditioner is kept constant throughout all 5000 time steps.
In the case of the loudspeaker we have to produce about 190 spectrally equivalent matrices, but this computational work is negligible compared to the total CPU time.

	ρ_{PCG}	t_{CPU}
ModPCAMG	0.75	7.5 h
PCAMG	0.84	8.9 h
direct solver	-	20 h

Table 5: Convergence rates and CPU-time for the electrodynamic loudspeaker.

5 Conclusions and further Remarks

The numerical results show the robustness of the proposed technique if anisotropic elements are used for the discretization. First, the element preconditioning technique can also be applied to 3D problems. Second, if we have an f.e. discretization of a scalar elliptic selfadjoint equation then this strategy can be used for the arising s.p.d. matrices. This can be done if we have access to the element stiffness matrices. However it might be a large numerical effort to produce an M-matrix in the general case even with the classification strategy. Additionally this method was introduced for scalar elliptic equations. A straight forward approach to systems is not obvious and maybe not possible.
Another way to overcome the 'M-matrix property' is to seek for a new interpolation method and/or for a new coarsening strategy. This means, if the interpolation could be improved, then we should end up with a better convergence rate (even if just s.p.d. matrices are considered). Of course this is closely related to the coarsening strategy because positive off-diagonal entries have to be treated in the right way.

Acknowledgment I would like to give special thanks to Prof. Ulrich Langer and Gundolf Haase for their great help writing this paper. Moreover special thanks to Joachim Schöberl for his f.e. code FEPP and a lot of helpful discussions, Manfred Kaltenbacher for the program package CAPA, and Wolfram Mühlhuber for his optimization code.

References

[1] R. E. Bank and T. F. Dupont, *Analysis of a two-level scheme for solving finite element equations*, Report CNA-159, Center for Numerical Analysis, University of Texas at Austin, May 1980.

[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).

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

[5] ______, *Algebraic multigrid (amg) for sparse matrix equations*, Sparsity and it's 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] M. Griebel, *Multilevelmethoden als Iterationsverfahren über Erzeugendensystemen*, B.G.Teubner, Stuttgart, 1994.

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

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

[10] 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.

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

[12] 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/.

[13] 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.

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

[15] J. Schöberl, *Robust multigrid preconditioning for parameter-dependent problems I: The stokes-type case*, Technical Report 97-2, Johannes Kepler Universität Linz, Institut für Mathematik, 1997, www.nathan.numa.uni-linz.ac.at/pub/technical_reports/.

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