AN ALGEBRAIC MULTILEVEL ITERATION METHOD FOR FINITE ELEMENT MATRICES*

O. Axelsson† and M. Larin‡

Abstract

To solve a sparse linear system of equations resulting from the finite element approximation of elliptic self-adjoint second order boundary value problems an algebraic multilevel iteration method is presented. The new method can be considered as an extension of methods, which have been defined by Axelsson and Eijkhout [4] for nine-point matrices and later generalized by Axelsson and Neytcheva [6] for the Stieltjes matrices, to a wider class of sparse symmetric positive-definite matrices. The rate of convergence and the computational complexity of the method are analyzed. Experimental results on some standard test problems are presented and discussed.

Keywords: algebraic multilevel iterative method, preconditioned conjugate gradient method, generalized eigenvalue problem.

1 Introduction

Many problems in science and engineering can be reduced to solving an algebraic linear system of equations

$$Ax = b, \tag{1.1}$$

where A is a sparse symmetric positive-definite matrix of order N, which arises from finite element approximations of elliptic self-adjoint second order boundary value problems.

In recent time much interest has been devoted to the construction of preconditioners of optimal order of computational complexity for the solution of system (1.1). For such preconditioners one can find an iterative method which converges in a number of iterations

*This work is part of the Joint Research Project "High performance computing in numerical simulation" founded by INTAS within the framework of Cooperation Agreement INTAS-93-377.

†Department of Mathematics, Catholic University of Nijmegen, Toernooiveld 1, 6525 ED Nijmegen, The Netherlands

‡Department of Problems of Physics and Chemistry, Computing Center SD RAS, Lavrentiev ave. 6, 630090 Novosibirsk, Russia

independent of N, and a computational cost per iteration which is proportional to N. In particular, algebraic multilevel and multigrid methods allow us to construct preconditioning matrices with these properties, see [2, 3, 4, 6, 10, 11, 12], for instance.

In the present paper we will use the algebraic multilevel iteration (AMLI) method to define the preconditioners for the preconditioned conjugate gradient (PCG) method. The preconditioning matrix M can be recursively defined as a sequence of preconditioning matrices $M^{(k)}$ corresponding to a sequence of matrices $\{A^{(k)}\}$ of an increasing order n_k, $k = k_0, k_0 + 1, \ldots, L$, where k_0 is the level number of the coarsest mesh used. Note that $A = A^{(L)}$ and $M = M^{(L)}$.

There are two different ways to define the sequence $\{A^{(k)}\}$. The first approach is based on the finite element approximation of reduced matrices for a sequence of nested meshes, for details, see [2, 3, 12]. The second one is to form the reduced matrices explicitly as Schur's complements of intermediate matrices of the previous level, which are constructed by some technique of deletion of certain matrix entries and diagonal compensation of those entries, see [1, 4, 6].

The present paper belongs to the second class and can be considered as an extension of methods, which have been defined by Axelsson and Eijkhout [4] for nine-point matrices (see also Brand [7]) and later generalized by Axelsson and Neytcheva [6] for the Stieltjes matrices. These methods are methods of deletion of matrix entries and diagonal compensation of them. The first method is based on the recursive red-black ordering of mesh nodes, aproximating the nine-point stencils by five-point ones at the red points and then forming the reduced system explicitly as the Schur's complement system of the previous level, which is then a nine-point stencil on a skew oriented mesh. The second one is an extension of the first method to more general classes of Stieltjes matrices.

The main difference between the method presented here and those used previously is that, instead of the recursive red-black ordering on a rectangular mesh we make use of a special recursive one on a triangular mesh, and in contrast to the five-point approximation of nine-point stencils in some nodes we apply a four-point approximation of the standard seven-point finite element one. In addition, we will show that when choosing a proper recursive ordering, the presented method is readily applicable also when applied for irregular finite element meshes.

The paper is organized as follows. In Section 2 the background for the AMLI method is recalled. Section 3 is devoted to the description of the algorithm to approximate the first pivot block of the $A^{(k)}$ matrices. The computational complexity and the rate of convergence will be discussed in Sections 4 and 5. The local analysis of the quality of the preconditioning matrix is given in Section 6. In the final section, performance results on standard test problems are presented and discussed.

2 The AMLI method

To create the preconditioning matrix M we have to construct a sequence of matrices $\{A^{(k)}\}$, $k = k_0, k_0 + 1, \ldots, L - 1, L$ of an increasing order n_k, where $A^{(L)} = A$ and A is assumed to

be a sparse symmetric positive definite matrix.

Let $\{G_k\}$ be a sequence of matrix graphs corresponding to the sequence of matrices $\{A^{(k)}\}$, i.e., $G_k = (X_k, E_k) = G(A^{(k)})$, where $X_k = X(A^{(k)})$ is a set of vertices and $E_k = E(A^{(k)})$ is a set of edges in the matrix graph for A^k, where $(i,j) \in E_k$ if and only if $a_{ij}^{(k)} \neq 0$. We assume that the following conditions are satisfied:

(1) $\{X_k\}$ is a sequence of nested meshes, i.e.,

$$X_{k_0} \in X_{k_0+1} \in \ldots \in X_k \in X_{k+1} \in \ldots \in X_L, \tag{2.1}$$

(2) the number of vertices increase in a geometric ratio, i.e.,

$$\frac{n_{k+1}}{n_k} = \rho_k \geq \rho > 1, k = k_0, k_0+1, \ldots, L-1. \tag{2.2}$$

Now we define the sequence of matrices $\{A^{(k)}\}$ by recursion for decreasing values of k: Each matrix $A^{(k+1)}, k > 0$ is partitioned in a certain manner into a two by two block matrix form

$$A^{(k+1)} = \begin{bmatrix} A_{11}^{(k+1)} & A_{12}^{(k+1)} \\ \\ A_{21}^{(k+1)} & A_{22}^{(k+1)} \end{bmatrix}, \tag{2.3}$$

where the first pivoting block corresponds to the vertices in X_{k+1}, which are not in X_k, and the second one corresponds to those in X_k. The selection of sets of the vertices X_k will be discussed in detail in Section 3. It is assumed that $A_{11}^{(k+1)}$ is diagonally dominant.

Now define a symmetric and positive definite matrix $\overline{A_{11}^{(k+1)}}$ as an approximation of $A_{11}^{(k+1)}$, which will be discussed in detail below (see (2.6)).

Next we consider an intermediate matrix

$$\widetilde{A}^{(k+1)} = \begin{bmatrix} \overline{A_{11}^{(k+1)}} & A_{12}^{(k+1)} \\ \\ A_{21}^{(k+1)} & A_{22}^{(k+1)} \end{bmatrix}, \tag{2.4}$$

which is also a symmetric positive definite matrix. This result will be proven in Section 5.

We define now the matrix $A^{(k)}$ as the Schur's complement of $\widetilde{A}^{(k+1)}$, i.e.,

$$A^{(k)} = A_{22}^{(k+1)} - A_{21}^{(k+1)} \left[\overline{A_{11}^{(k+1)}}\right]^{-1} A_{12}^{(k+1)}. \tag{2.5}$$

Note that $A^{(k)}$ is a symmetric positive definite matrix being a Schur's complement of the symmetric positive definite matrix [8]. In order to preserve some sparsity pattern in the matrix sequence one must choose $\overline{A_{11}^{(k+1)}}$ properly. Typically, it is diagonal or block-diagonal.

Here we approximate $A_{11}^{(k+1)}$ by a diagonal and positive matrix $\overline{A_{11}^{(k+1)}}$, defined by

$$A_{11}^{(k+1)} e^{(k+1)} = \overline{A_{11}^{(k+1)}} e^{(k+1)}, \tag{2.6}$$

where $e^{(k+1)} = (1, ..., 1)^T$ is a column-vector of the order $n_{k+1} - n_k$.

To proceed, we apply a similar operation to each matrix $A^{(k)}$ and repeat this process until the matrix corresponding to a coarse mesh is obtained.

The preconditioning matrix M is defined recursively, based on a sequence of matrices $\{M^{(k)}\}$, which are preconditioners for the sequence $A^{(k)}$, $k = k_0, k_0 + 1, \ldots, L$. Here they are defined as follows:

$$
\begin{aligned}
& M^{(k_0)} = A^{(k_0)} \\
& \text{for } k = k_0 + 1, \ldots, L \\
& M^{(k+1)} = \begin{bmatrix} \overline{A_{11}^{(k+1)}} & 0 \\ A_{21}^{(k+1)} & I \end{bmatrix} \begin{bmatrix} I & \left[\overline{A_{11}^{(k+1)}}\right]^{-1} A_{12}^{(k+1)} \\ 0 & S^{(k)} \end{bmatrix},
\end{aligned}
\tag{2.7}
$$

where the matrix $\overline{A_{11}^{(k)}}$ is defined in (2.6) and

$$
S^{(k)} = A^{(k)} \left[I - P_{\nu_k}(\left[M^{(k)}\right]^{-1} A^{(k)}) \right]^{-1}, \tag{2.8}
$$

where $P_{\nu_k}(x)$ is a polynomial of degree ν_k, normalized by $P_{\nu_k}(0) = 1$, and which is small in the interval $I_k = [\underline{t}_k, \overline{t}_k]$ containing the eigenvalues of $\left[M^{(k)}\right]^{-1} A^{(k)}$. Let in addition $P_{\nu_k}(x) > 0,\ x \in I_k$ hold. The choice of the polynomials will be discussed in detail in Sections 5 and 6. Note that both $A^{(k)}$ and $M^{(k)}$ are symmetric and positive definite so the eigenvalues of $\left[M^{(k)}\right]^{-1} A^{(k)}$ are real and positive.

3 Description of sets X_k

At first we make an important remark. If we delete some properly chosen couplings for a group of nodes(unknowns) before their elimination, then the resulting Schur's complement will have a simple structure as compared to the structure of the exact Schur's complement, which would have arisen if no initial deletions were done.

This fact is used to preserve the sparsity structure of the Schur's complement of the intermediate matrix $\widetilde{A}^{(k+1)}$ in the process of the factorization. This will be discussed now.

Let a matrix $A^{(k+1)}$, which corresponds to the structure of a triangular finite element mesh/matrix, be given on each level, i.e., the graph $G_{k+1} = (X_{k+1}, E_{k+1})$ is defined.

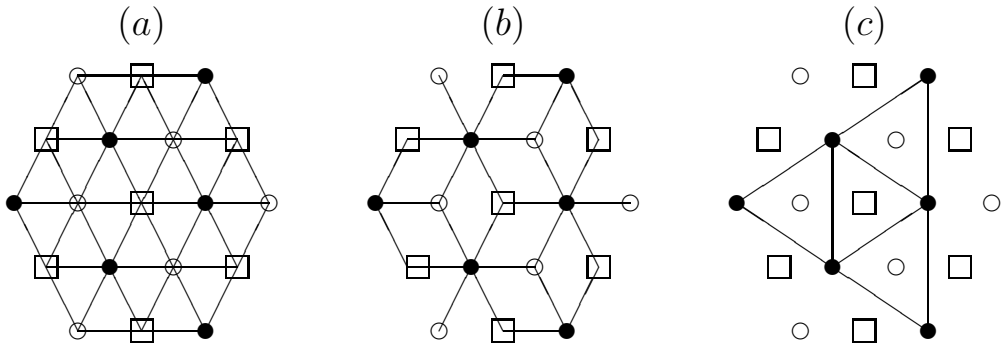


Fig.1. Transformation of the graph G_{k+1} into the graph G_k
(a) Example of coloring of a triangular mesh
(b) The triangular mesh after deleting certain couplings
(c) The new triangular mesh

First step (Partitioning the set of vertices X_{k+1}): Color the nodes of the graph G_{k+1} in three colors: red($\circ$), blue($\square$) (the set $X_{k+1}\backslash X_k$) and green($\bullet$) (the set X_k) by a certain principle: nodes which have the same color are not connected, i.e., there are no couplings between the nodes of the same color (Fig.1(a)). A classical result states that for an arbitrary planar graph it suffices with four colors. For graphs with some special property fewer colors may suffice. For instance, for an equilateral triangulation, it suffices with three and the familiar five point difference rectangular mesh requires only two colors.

Second step (Approximation of the submatrix $A_{11}^{(k+1)}$): By definition, the approximation process is a deletion of all off-diagonal entries and their diagonal compensation, i.e., we have to delete all couplings between red and blue nodes. It leads to the mesh corresponding to the intermediate matrix $\widetilde{A}^{(k+1)}$, which is shown in Fig.1(b).

Third step (Calculation of the new matrix $A^{(k)}$): Make a graph reconstruction, which corresponds to Gaussian elimination of all red and blue nodes, for example, see [9]. It generates a new triangular finite element mesh G_k for the green nodes, which has the same structure and much fewer number of nodes (Fig.1(c)), than G_{k+1}.

As the corresponding matrix has again the same finite element matrix structure, the process can be repeated until a sufficiently coarse grid is reached, where the linear system of equations can be solved by a direct method with computational cost sufficiently small in relation to n_L. For example, if the coarse mesh problem is solved by a direct method one can use the following criterion

$$n_{k_0} \approx (n_L)^{1/2}. \tag{3.1}$$

In this case the total computational cost to solve the system with matrix $A^{(k_0)}$ has the order $O(n_L^{2/3})$ and the cost to factor $A^{(k_0)}$ has the order $O(n_L)$. We have here assumed that the nodes of the coarse mesh have been ordered such that the bandwidth of $A^{(k_0)}$ is $O(\sqrt{n_{k_0}})$, or less.

Alternatively, one can use a simple iterative solution such as the Jacobi PCG method to solve the coarse mesh matrix problems. This can enable the use of finer meshes for the

coarsest mesh. In this case the number of iterations depend on the condition number of this matrix. Typically one has $O(\sqrt{n_{k_0}})$ iterations, in which case the cost of the coarse mesh solver is balanced to the finest mesh if we let $n_{k_0} \approx (n_L)^{2/3}$. For a further discussion of this approach, see [6].

4 Computational complexity

At each level we have to solve a system with the matrix $M^{(k+1)}$, which is a preconditioner to the matrix $A^{(k+1)}$. By the choice of $S^{(k)}$ and the structure of the matrix sequences $\{A^{(k)}\}$ and $\{M^{(k)}\}$ it requires only some matrix-vector multiplications and vector additions. More precisely, solution of a system with matrix $M^{(k+1)}$ breaks up into two subproblems: forward

$$\begin{aligned} z_1 &= \left[\overline{A_{11}^{(k+1)}}\right]^{-1} y_1, \\ z_2 &= y_2 - A_{21}^{(k+1)} z_1, \end{aligned} \tag{4.1}$$

and back substitutions

$$\begin{aligned} x_2 &= {S^{(k)}}^{-1} z_2, \\ x_1 &= z_1 - \left[\overline{A_{11}^{(k+1)}}\right]^{-1} A_{12}^{(k+1)} x_2. \end{aligned} \tag{4.2}$$

Here we have to solve two systems with the diagonal matrix $\overline{A}_{11}^{(k+1)}$ and the system with $S^{(k)}$, which as was earlier suggested in [2], can be written in the following way

$$\begin{aligned} &\text{Solve } M^{(k)} x = a_{\nu_k} z_2. \\ &\text{For } r = 1 \text{ step 1 until } \nu_k - 1 : \\ &\qquad \text{solve } M^{(k)} x = A^{(k)} x + a_{\nu_k - r} z_2, \end{aligned}$$

where

$$P_{\nu_k}(t) = 1 - a_1 t - \ldots - a_{\nu_k} t^{\nu_k}.$$

Hence to solve the system with $M^{(k+1)}$ we need to solve ν_k systems with $M^{(k)}$. Each of these systems requires ν_{k-1} solutions with $M^{(k-1)}$ and so on. To estimate the total computational complexity we can use a recursive technique, which has been already discussed in [6].

Denote by w_L the computational complexity of the solution of the system with preconditioning matrix $M^{(L)}$ and recall that n_k is the number of points of the mesh at level k. Let ν, $0 \le \nu \le L$ be an integer parameter and let the polynomial degrees ν_k be chosen as 1 for every μ consecutive values of k, i.e.,

$$\begin{aligned} &\nu_L = 1, \ \nu_{L-1} = 1, \ \ldots, \ \nu_{L-\mu+1} = 1, \ \nu_{L-\mu} = \nu, \\ &\nu_{L-\mu-1} = 1, \ \nu_{L-\mu-2} = 1, \ \ldots, \ \nu_{L-2\mu} = 1, \ \nu_{L-2\mu-1} = \nu, \\ &\nu_{L-2\mu-2} = 1, \ \nu_{L-2\mu-3} = 1, \ \ldots, \ \nu_{k_0} = 1. \end{aligned} \tag{4.3}$$

Here ν is the number of iterations of the above polynomial type iterative process, which solves the system with the matrix $\left[M^{(k)}\right]^{-1} A^{(k)}$, and which is used for obtaining a spectrally equivalent preconditioner, and μ is the number of levels, after which we repeat this process.

Let the number of nodes n_{k_0} at the coarse level be defined by criterion (3.1) and the amount of work for solution at the coarsest level be proportional to $n_{k_0}\sqrt{n_{k_0}}$. Recall that the latter holds when we solve the coarse mesh problem by a direct method and the bandwidth of the corresponding matrix is $O(\sqrt{n_{k_0}})$. Then

$$w_L = C(n_L + \ldots + n_{L-\mu}) + C\nu(n_{L-\mu-1} + \ldots + n_{L-2\mu})$$

$$+ C\nu^2(n_{L-2\mu-1} + \ldots + n_{L-3\mu}) + \ldots + C\nu^{\frac{L-k_0}{\mu+1}} n_{k_0}^{3/2}$$

$$\leq Cn_L \cdot \left[1 + \tfrac{1}{\rho} + \ldots + \left(\tfrac{1}{\rho}\right)^{\mu}\right] \cdot \frac{1}{1 - \nu\rho^{-(\mu+1)}} + C\nu^{\frac{L-k_0}{\mu+1}} n_{k_0}^{3/2} \tag{4.4}$$

$$\leq Cn_L \cdot \frac{\rho^{\mu+1} - 1}{\rho^{\mu}(\rho - 1)} \cdot \frac{1}{1 - \nu/\rho^{\mu+1}} + Cn_L \cdot \left(\frac{\nu}{\rho^{\mu+1}}\right)^{\frac{L-k_0}{\mu+1}} \rho^{L-k_0} \frac{n_{k_0}^{3/2}}{n_L},$$

where ρ is the constant from (2.2) and C is an upper bound of the arithmetic work per meshpoint defined by two vector additions and three matrix-vector multiplications in (4.1) and (4.2). Due to the deletion of certain matrix entries as discussed in Section 3, C does not depend on L. The inequality for the first term in (4.4) is satisfied provided that

$$\nu < \rho^{\mu+1}, \tag{4.5}$$

and from (4.4) one can see that its coefficient of n_L is independent of the number of levels. To analyze the second term in (4.4), we assume that

$$\nu \leq (\theta\rho)^{\mu+1} \tag{4.6}$$

for some $\theta < 1$. Then, using (2.2) we find that the second term in (4.4) is bounded by $O(n_L)$ if

$$\theta^{L-k_0}\sqrt{n_{k_0}} \leq 1,$$

that is,

$$\theta \leq n_{k_0}^{-0.5/(L-k_0)}. \tag{4.7}$$

Here, $n_{k_0}^{-0.5/(L-k_0)} \to 1$, as $L \to \infty$, if k_0 is fixed, i.e., if the size of the coarse mesh problem is fixed. Hence, in this case we can take θ very close to unity. If we permit n_{k_0} to grow with n_L, for instance as $n_{k_0} = n_L^{\xi}$ for some $\xi < 1$, then it follows from (4.7) that $\theta \leq n_L^{-0.5\xi/(L-k_0)}$. Assume that $n_L = \rho^L$ (cf.(2.2)). Then $\xi = \dfrac{k_0}{L}$ and the above condition implies $\theta \leq \rho^{-0.5\xi/(1-\xi)}$. Hence θ can be taken arbitrary close to unity, if ξ is

sufficiently small. For the choice made in (3.1), $\theta \leq \rho^{-1/2}$. As a conclusion, the coefficient of n_L of the second term in (4.4) is independent of the number of nodes on the fine mesh if (4.6) and (4.7) holds. It follows that the inequalities of (4.6) and (4.7) define the upper bound of the degrees of polynomials, under which the total computational complexity for one application of the preconditioner is proportional to the number of nodes on the fine mesh, and that conditions (4.6) and (4.7) are only slightly stricter than (4.5).

5 Rate of convergence

In this section we will analyze the condition number of the matrix A to the preconditioner M and discuss some conditions for an optimal rate of convergence. This will be done by comparing the condition numbers on two adjacent levels.

For the further analysis we recall that the intermediate matrices $\{\widetilde{A}^{(k)}\}$ arise from deleting certain couplings in the first pivoting block $A_{11}^{(k)}$ of the original matrix $A^{(k)}$. The values of these couplings can be both negative and positive, because the matrices $A_{11}^{(k)}$ are not M-matrices in general. Now if we will try to find its approximation from (2.6), it may happen that the resulting matrix $\overline{A_{11}^{(k)}}$ is either non-positive definite or a singular matrix. To avoid these possibilities assume that on each level all positive off-diagonal entries are located outside $A_{11}^{(k)}$ (in $A_{12}^{(k)}$, for instance), i.e., $A_{11}^{(k)}$ are M-matrices.

Now to estimate the condition number we can use the following results, which were proven earlier for Stieltjes matrices, since in our case the value of constants β_{k+1} (see (5.1)), which arise as a result of the approximation of the first pivoting blocks, is changed only. Here we rewrite them in our notation.

The following inequalities for the condition numbers on each level

$$(1 - P_{\nu_k}(\underline{t}_k)) \leq \frac{x^T A^{(k+1)} x}{x^T M^{(k+1)} x} \leq \beta_{k+1} \tag{5.1}$$

hold, where we have assumed that

$$\max_{t \in I_k} P_{\nu_k}(t) = P_{\nu_k}(\underline{t}_k),\ I_k = [\underline{t}_k, \overline{t}_k],$$

is valid. Here we choose the boundary points of the interval I_{k+1} to be

$$\underline{t}_{k+1} = (1 - P_{\nu_k}(\underline{t}_k)), \quad \overline{t}_{k+1} = \beta_{k+1}. \tag{5.2}$$

The definition of the constants β_{k+1} will be discussed in detail in the next section. The polynomials $P_{\nu_k}(t)$ are chosen as

$$P_{\nu_k}(t) = \frac{T_{\nu_k}\left(\frac{\overline{t}_k + \underline{t}_k - 2t}{\overline{t}_k - \underline{t}_k}\right) + 1}{T_{\nu_k}\left(\frac{\overline{t}_k + \underline{t}_k}{\overline{t}_k - \underline{t}_k}\right) + 1}, \tag{5.3}$$

where $T_m(t)$ are the Chebyshev polynomials of degree m,

$$T_0 = 1,\ T_1 = t,\ T_{\nu+1} = 2tT_\nu - T_{\nu-1}.$$

To obtain the final condition on the lower bound of degrees of polynomials we use the standard technique, which is described in [6].

The condition of the optimal rate of convergence is

$$\nu > \left(\max_{\xi=1,2,\dots L/\mu} \prod_{s=L-\xi\mu}^{L-(\xi-1)\mu} \beta_s \right)^{\frac{1}{2}}, \tag{5.4}$$

where ν_k is chosen as in (4.3).

We collect the final results (4.6), (4.7) and (5.4) in the following theorem.

Theorem 1. *The algebraic multilevel iteration method for the finite element matrices, based on the sequence of the matrices $\{A^{(k)}\}$ defined with (2.5) and (2.6), and the sequence of their preconditioners $\{M^{(k)}\}$, recursively defined with (2.7), (2.8) and (5.3), has an optimal order of computational complexity if*

$$(\theta\rho)^{\mu+1} > \nu > \left(\max_{\xi=1,2,\dots L/\mu} \prod_{s=L-\xi\mu}^{L-(\xi-1)\mu} \beta_s \right)^{\frac{1}{2}}$$

where β_s is the above defined constants by (5.1) and (5.2), ρ is a coefficient of a geometric progression from (2.2), θ is a parameter, which depends on the number of nodes on the coarse mesh and defined by (4.7), ν is the degree of the matrix polynomials used in (2.8) and $\nu > 1$ at every $(\mu+1)$th step.

Thus, choosing proper degrees of the polynomial ν_k we have an optimal rate of convergence, i.e., the condition number of $M^{-1}A$ has magnitude $O(1)$, and the whole computational complexity is $O(n_L)$, i.e., is proportional to the number of nodes on the fine mesh.

6 Local analysis

To estimate the quality of a constructed preconditioning matrix we solve the following generalized eigenvalue problem on each level

$$A^{(k)}v = \lambda M^{(k)}v. \tag{6.1}$$

To do this we construct element matrices which can be associated with the global matrices and analyze their condition number, i.e., we reduce the analysis of the global generalized eigenvalue problem to a local, element by element analysis.

Lemma 1. *Let $\{A_i\}_{i=1}^n$ and $\{M_i\}_{i=1}^n$ be sequences of symmetric positive semidefinite matrices, $A = \sum_{i=1}^n A_i$, $M = \sum_{i=1}^n M_i$. Then, if for some positive constants α_i and $\beta_i \geq \alpha_i$ and for all $x \in R^n$*

$$\alpha_i x^T M_i x \leq x^T A_i x \leq \beta_i x^T M_i x$$

holds, then

$$\alpha x^T M x \le x^T A x \le \beta x^T M x,$$

where $\alpha = \min \alpha_i,\ \beta = \max \beta_i$.

Proof. Follows directly from the definition of parameters of α and β.

□

Lemma 1 shows that the global analysis can be reduced to a local analysis. In finite element applications, A_i usually correspond to the elementary finite element stiffness matrices, and M_i are their preconditioners. In this case A_i and M_i are typically singular except, possibly, a few matrices such as associated with boundary elements. However, the spectral relation between A and M is still valid if one of them is positive definite on the orthogonal complement to their common nullspace.

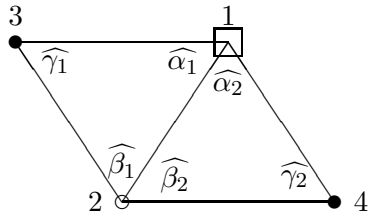


Fig.2. The superelement

Consider two triangular elements, which have a common edge (see Fig.2). It is well known that the corresponding finite element matrices for linear basis functions on arbitrary triangles, which are shown in Fig.2, are

$$K_1 = \frac{1}{2}\begin{bmatrix} \beta_1+\gamma_1 & -\gamma_1 & -\beta_1 \\ -\gamma_1 & \alpha_1+\gamma_1 & -\alpha_1 \\ -\beta_1 & -\alpha_1 & \beta_1+\alpha_1 \end{bmatrix},$$

$$K_2 = \frac{1}{2}\begin{bmatrix} \beta_2+\gamma_2 & -\gamma_2 & -\beta_2 \\ -\gamma_2 & \alpha_2+\gamma_2 & -\alpha_2 \\ -\beta_2 & -\alpha_2 & \beta_2+\alpha_2 \end{bmatrix},$$

where $\alpha_1, \beta_1, \gamma_1, \alpha_2, \beta_2, \gamma_2$ are cotangents of $\widehat{\alpha_1}, \widehat{\beta_1}, \widehat{\gamma_1}, \widehat{\alpha_2}, \widehat{\beta_2}, \widehat{\gamma_2}$, respectively. Note that this relation is valid only for the original grid since on the following grids the values of $\alpha_i,\ \beta_i$ and γ_i are not cotangents of suitable angles. However, for each triangular element of the graph $G_k, 0 < k < L$ one can assign a similar elementary matrix, where $\alpha_i,\ \beta_i,\ \gamma_i$ are values of off-diagonal entries of $A^{(k)}$. For example,

$$\alpha_1 = -a_{23}^{(k+1)}, \beta_1 = -a_{13}^{(k+1)}, \gamma_1 = -a_{12}^{(k+1)}.$$

After having assembled these elementary matrices, we derive the following superelement matrix

$$K = \frac{1}{2}\begin{bmatrix} \beta_1+\gamma_1+\beta_2+\gamma_2 & -\gamma_1-\gamma_2 & -\beta_1 & -\beta_2 \\ -\gamma_1-\gamma_2 & \alpha_1+\gamma_1+\alpha_2+\gamma_2 & -\alpha_1 & -\alpha_2 \\ -\beta_1 & -\alpha_1 & \beta_1+\alpha_1 & 0 \\ -\beta_2 & -\alpha_2 & 0 & \beta_2+\alpha_2 \end{bmatrix}. \tag{6.2}$$

Two nodes, which lay on the common edge, are red and blue, the other nodes are green. Deleting the coupling between red(1) and blue(2) nodes, we obtain

$$B = \begin{bmatrix} \beta_1+\beta_2 & 0 & -\beta_1 & -\beta_2 \\ 0 & \alpha_1+\alpha_2 & -\alpha_1 & -\alpha_2 \\ -\beta_1 & -\alpha_1 & \beta_1+\alpha_1 & 0 \\ -\beta_2 & -\alpha_2 & 0 & \beta_2+\alpha_2 \end{bmatrix}. \tag{6.3}$$

In what follows for a correct definition of the Schur complement matrix of B (see (6.5)) we assume that

$$\alpha_1+\alpha_2 \neq 0, \text{ or } \beta_1+\beta_2 \neq 0. \tag{6.4}$$

Now we have to solve the local generalized eigenvalue problem

$$Kw = \lambda Bw$$

or, by partitioning the matrices into two by two blocks,

$$\begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \lambda \begin{bmatrix} B_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}. \tag{6.5}$$

The solution of this problem is as follows:

1. $\lambda = 1$, $w_1 = 0$, $w_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$;
2. $\lambda = 1$, $w_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$, $w_2 = 0$;
3. $\lambda \neq 1$, $w_2 = -K_{22}^{-1}K_{21}w_1$ and $(K_{11} - K_{12}K_{22}^{-1}K_{21})w_1 = \lambda(B_{11} - K_{12}K_{22}^{-1}K_{21})w_1$, where w_1 is not in the nullspace of the two matrices.

An elementary calculation shows that in the latter case the generalized eigenvalue problem reduces to

$$(\gamma_1+\gamma_2+\frac{\alpha_1\beta_1}{\beta_1+\alpha_1}+\frac{\alpha_2\beta_2}{\beta_2+\alpha_2}) \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} w_1 = \lambda(\frac{\alpha_1\beta_1}{\beta_1+\alpha_1}+\frac{\alpha_2\beta_2}{\beta_2+\alpha_2}) \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} w_1,$$

which has the unique solution

$$\lambda = 1 + \frac{(\gamma_1 + \gamma_2)(\alpha_1 + \beta_1)(\alpha_2 + \beta_2)}{\alpha_1\beta_1(\alpha_2 + \beta_2) + \alpha_2\beta_2(\alpha_1 + \beta_1)}. \tag{6.6}$$

The latter is true if the numerator of the fraction is not equal to zero, i.e.,

$$\alpha_1\beta_1(\alpha_2 + \beta_2) + \alpha_2\beta_2(\alpha_1 + \beta_1) \neq 0. \tag{6.7}$$

Theorem 2. *The upper bound of the maximum eigenvalue of the generalized eigenvalue problem (6.1) on level k is*

$$\overline{t}_k < 1 + \max_{e \in E_k} \left(\frac{(\gamma_1 + \gamma_2)(\alpha_1 + \beta_1)(\alpha_2 + \beta_2)}{\alpha_1\beta_1(\alpha_2 + \beta_2) + \alpha_2\beta_2(\alpha_1 + \beta_1)} \right),$$

where $\alpha_i, \beta_i, \gamma_i, i = 1, 2$, are the above-defined entries of the superelement matrix and E_k is the set of superelement matrices on level k.

To investigate the function $\lambda(\alpha_1, \beta_1, \alpha_2, \beta_2)$ assume that condition (6.7) holds. In this case the conditions of extremum of the function $\lambda(\alpha_1, \beta_1, \alpha_2, \beta_2)$

$$\frac{\partial\lambda}{\partial\alpha_1} = 0, \quad \frac{\partial\lambda}{\partial\beta_1} = 0, \quad \frac{\partial\lambda}{\partial\gamma_1} = 0,$$

$$\frac{\partial\lambda}{\partial\alpha_2} = 0, \quad \frac{\partial\lambda}{\partial\beta_2} = 0, \quad \frac{\partial\lambda}{\partial\gamma_2} = 0,$$

reduce to the following system

$$\begin{aligned}
\beta_1^2(\alpha_2 + \beta_2)^2(\gamma_1 + \gamma_2) &= 0,\\
\beta_2^2(\alpha_1 + \beta_1)^2(\gamma_1 + \gamma_2) &= 0,\\
\alpha_1^2(\alpha_1 + \beta_1)^2(\gamma_1 + \gamma_2) &= 0,\\
\alpha_2^2(\alpha_1 + \beta_1)^2(\gamma_1 + \gamma_2) &= 0,\\
(\alpha_1 + \beta_1)(\alpha_2 + \beta_2) &= 0,
\end{aligned} \tag{6.8}$$

which has the solutions

1. $\gamma_1 + \gamma_2 = 0,\ \alpha_1 + \beta_1 = 0,\ \alpha_1 \neq 0,\ \beta_1 \neq 0;$
2. $\gamma_1 + \gamma_2 = 0,\ \alpha_2 + \beta_2 = 0,\ \alpha_2 \neq 0,\ \beta_2 \neq 0;$
3. $\alpha_1 + \beta_1 = 0,\ \alpha_2 + \beta_2 = 0,\ \alpha_1 \neq 0,\ \beta_1 \neq 0,\ \alpha_2 \neq 0,\ \beta_2 \neq 0;$
4. $\alpha_1 = 0,\ \beta_1 = 0;$
5. $\alpha_2 = 0,\ \beta_2 = 0.$

In what follows we need the next lemma.

Lemma 2. *Assume that no entries α_i, i=1,2, and β_i, i=1,2, of the matrix B, defined by (6.3), are equal to zero. Then all the entries of its Schur's complement are nonzero too.*

Proof. Since B is an elementary matrix of $\widetilde{A}^{(k+1)}$, then its Schur's complement is defined from the new entries of the matrix $A^{(k)}$. By hypothesis of the lemma we have

$$\alpha_1 \neq 0,\ \beta_1 \neq 0,\ \alpha_2 \neq 0,\ \beta_2 \neq 0. \tag{6.9}$$

Then after elimination of red(1) and blue(2) nodes (see (6.3) and Fig.3) we obtain a new coupling between the remaining green nodes (3 and 4)

$$S = K_{22} - K_{12}B_{11}^{-1}K_{21} = \left(\frac{\alpha_1\alpha_2}{\alpha_1+\alpha_2} + \frac{\beta_1\beta_2}{\beta_1+\beta_2}\right)\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \tag{6.10}$$

which is not equal to zero by (6.9).

□

Lemma 2 shows that if the matrix graph of $A^{(L)}$ on the level L is identical to the triangular finite element grid, then the matrix graphs on the remaining levels are also identical to the triangular finite element grids. Hence, solutions 4–5 of system (6.8) are beyond reach, i.e., these extremal points cannot occur.

Moreover, due to the fact that on the level L the values of $\widehat{\alpha_i}, \widehat{\beta_i}, \widehat{\gamma_i}, i = 1, 2$ are angles of triangles the following expressions

$$\alpha_1 + \beta_1 \neq 0,\ \alpha_2 + \beta_2 \neq 0, \tag{6.11}$$

hold. By (6.11) the solutions 1–3 of system (6.8) are also beyond reach. Thus, the function $\lambda(\alpha_1, \beta_1, \alpha_2, \beta_2)$ has no extremum points for arbitrary values of α_i and β_i, $i = 1, 2$.

However, there is a possibility that condition (6.7) is not valid. For example, if we have $\widehat{\alpha_1} = \widehat{\beta_2} = \frac{\pi}{2}$, i.e., two isoscele right-angled triangles, then $\alpha_1 = \beta_2 = 0$. Note that in this case the coupling between 1 and 2 is equal to zero and therefore the matrix graph is not triangular. Thus, (6.7) is the first condition on the original grid. The second one is (6.4), which may not hold either. For example, if we have $\widehat{\alpha_1} = x, \widehat{\alpha_2} = \pi - x$, then

$$\alpha_1 + \alpha_2 = \cot x + \cot(\pi - x) = \cot x + (-\cot x) = 0.$$

The analysis made shows that the suggested method is applicable for an arbitrary triangulation for which conditions (6.4) and (6.7) hold. On the other hand, the method is not applicable to the isosceles right triangular mesh. However, these conditions can be always satisfied by a local modification of the original grid with the help of selected conditions (6.4) and (6.7), but as we shall see in the numerical tests, the resulting condition number can be big.

Remarks. From the first point of view it seems that there is a contradiction between the results of local and global analyses. There are two conditions (6.4) and (6.7) in the local analysis and there are none in the global one. However, it is not true!

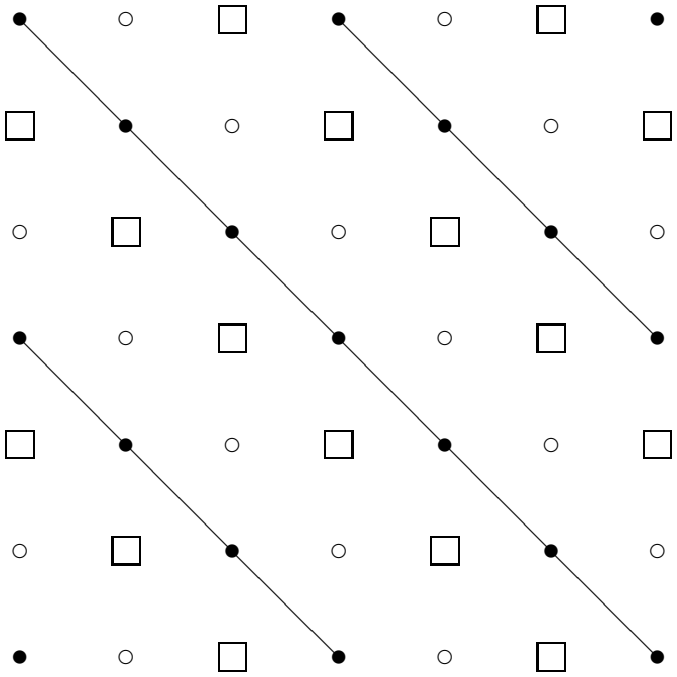


Fig.3. Example of the graph G_{L-1} in case of regular square mesh

The first condition arises as a result of the difference between the estimates obtained in the global and the local analyses. From proof of Lemma 1 it is readily seen that the estimates of the local analysis are stronger than the global one. As a result of a more detailed analysis of (6.4) one can see that if either of $\alpha_1 + \alpha_2 = 0$ or $\beta_1 + \beta_2 = 0$ is equal to zero, then the corresponding diagonal entry of the local preconditioning matrix B (see (6.3)) is equal zero too. However, the whole contribution of all superelements, based on the vertex, cannot be equal to zero. Thus, condition (6.4) may be omitted.

The second condition arises as a result of the difference between the notions "matrix graph" and "grid". They are not identical! And the example with the isosceles right triangular mesh shows in this case that the matrix graph is the square regular mesh and in contrast the grid is the standard seven-point regular mesh. Hence the deletion of couplings on the rectangular grid by the rules for a triangular grid may cause an undesirable surprise. It is readily seen that after the first reduction we obtain the graph G_{L-1} (see Fig.3), which corresponds to the block-diagonal matrix with tridiagonal blocks. If we continue our procedure now, then we obtain the following undesirable situation: for an vertex of G_{L-1}, which has only two real couplings, we must delete them(!). Thus, the matrix block $A_{11}^{(k)}$ becomes indefinite.

7 Numerical experiments

In this section we will present and discuss numerical experiments for various type of meshes. We will test the method described above on the Laplace equation

$$-\Delta u = 0 \tag{7.1}$$

on a domain Ω with Dirichlet boundary conditions

$$u\Big|_{\Gamma} = 1, \quad \text{where } \Gamma = \partial\Omega. \tag{7.2}$$

The coefficient matrix on the finest level was derived using standard piecewise linear finite elements on an triangular mesh. The right-hand side in the system of equations was choosen so that the solution has the form

$$u = x(1-x)y(1-y)e^{xy}. \tag{7.3}$$

The domains were chosen either as a regular hexagon or a unit square.

The solution method is the preconditioned conjugate gradient method with a preconditioner M defined by (2.7), (2.8) and (5.3). The initial approximation was always taken as the zero vector. The following stopping criterion was used

$$\frac{r_i^T M^{-1} r_i}{r_1^T M^{-1} r_1} < 10^{-12}, \tag{7.4}$$

where r_1 and r_i are the initial and the current residuals, respectively.

At first let Ω be a unit square $\Omega = (0,1)^2$ and let the mesh on the finest level be a isosceles right triangular mesh. In Section 4 was noted that isosceles right triangular mesh is bad mesh for this method. It is confirmed by numerical experiments. After two reductions we derived a new rectangular $(m \times m)$-mesh, for which the matrix $B_{11}^{(k+1)}$ or $B_{22}^{(k+1)}$ is singular. Step by step modification of original mesh in the case (20×20)-mesh is illustrated in Appendix A.

There are two possibilities to solve this problem. First we can use another approximation of the first pivoting block on each level. Another possibility is to use another triangular (general) mesh. In the present paper we shall investigate the second method.

For example, in the case of regular isosceles triangular mesh we can make some perturbation of coordinates of interior nodes in one direction by some small parameter $\widetilde{\varepsilon} = \varepsilon h$, where h is character mesh size and $\varepsilon << 1$. Let the interior nodes of original isosceles triangular mesh have the coordinates

$$\begin{aligned} x_{ij} &= (j-1)h, \;\; i = 2, ..., n+1, \\ y_{ij} &= (i-1)h, \;\; j = 2, ..., n+1, \end{aligned} \tag{7.5}$$

then after perturbation they have the following coordinates

$$\begin{aligned} x_{ij} &= (j-1)h + \varepsilon h, \;\; j = 2, ..., n+1, \\ y_{ij} &= (i-1)h, \;\; i = 2, 4, 6, ..., 2\lceil \tfrac{n+1}{2} \rceil, \\ x_{ij} &= (j-1)h, \;\; j = 2, ..., n+1, \\ y_{ij} &= (i-1)h, \;\; i = 3, 5, 7, ..., 2\lceil \tfrac{n+1}{2} \rceil + 1, \end{aligned} \tag{7.6}$$

where $\lceil x \rceil$ is integer part of x, i.e., we shifted on $\widetilde{\varepsilon} = \varepsilon h$ only nodes on even numbered coordinate lines. It allow us to avoid right angles in the elements(triangles).

Table 1. PCG method, $\varepsilon = 0.01$

7^2	μ	ν	Number of iterations	15^2	μ	ν	Number of iterations
	1	2	17		1	2	195
	2	3	25		2	3	223
	0	1	15		0	1	57
	0	2	17		0	2	145
	0	3	23		0	3	225

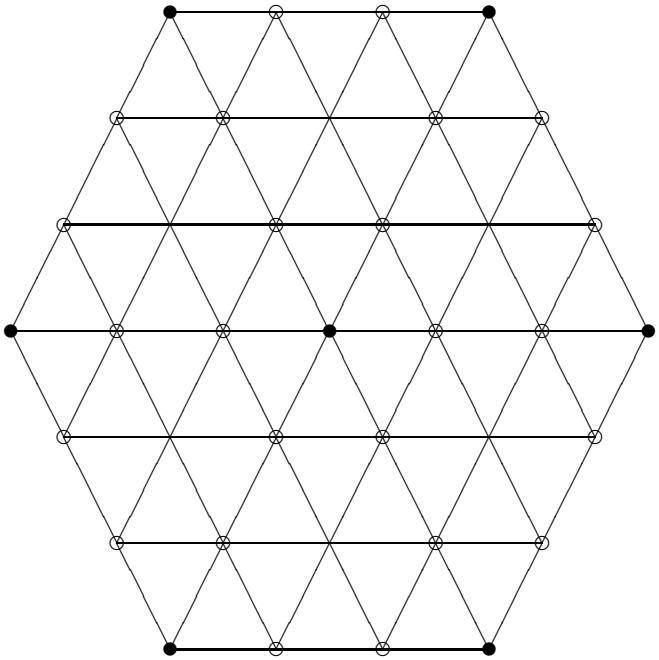


Fig.4. Example of a triangulation of the hexogonal domain for $k = 2$

In Table 1 the results of the experiments for the preconditioned conjugate gradient method for different choices of polynomial degrees and grids are given. In all tables of the present section ν is the degree of the matrix polynomials used and $\nu > 1$ at every $(\mu+1)$th level. For example, in a case $\nu = 2,\ \mu = 0$ we have a two-fold W-cycle.

From Table 1 one can see that for a resonable choice of the polynomial degrees or, equivalently, the values μ and ν, the W-cycle versions of the method has not improved the rate of convergence since due to the fact that the angles of triangles are close to right ones

the upper bound of the maximum eigenvalue of the matrix $\left[M^{(k)}\right]^{-1} A^{(k)}$ is large. Thus, the lower bound on polynomial degrees is not satisfied.

Another numerical experiment has been performed for the same problem with another domain Ω. Assume that Ω is a regular hexagonal(Fig.4), which can be the mapped domain using a certain isomorphic mapping from a triangular, a L-shape and a polygonal domain.

Let the initial triangulation of Ω, consisting of six equilateral triangles, be given. Introduce a refinement parameter k, which defines how many nodes are located on each edge of the original hexagonal. In Figure 4 one can see an example for $k = 2$. Step by step modification of the original hexagonal mesh in the case $k = 10$ is found in Appendix B.

Let l be the number of levels, corresponding to the given refinement parameter k. The results of solution the problem (7.1)–(7.2) for unpreconditioned and preconditioned CG methods in a case of $\mu = 0$, $\nu = 1$ (V-cycle) are shown in Tables 2 and 3, respectively.

Table 2. CG method

k	Number of unknowns	Number of iterations	Average reduction factor
5	91	22	0.5275
10	331	41	0.7073
15	721	59	0.7897
20	1261	77	0.8349
25	1951	95	0.8635

Table 3. PCG method, V-cycle

k	Number of levels	Number of unknowns	Number of iterations	Average reduction factor
5	4	91	12	0.2950
10	5	331	14	0.3647
15	5	721	16	0.3994
20	6	1261	18	0.4413
25	6	1951	19	0.4641

In Table 4 one can see the behaviour of the number of iterations of the method for various values of μ and ν, $\nu \neq 1$ (W-cycle).

On the basis of the experiments the following conclusions can be made.

1. Comparing the results of Table 2 to the ones of Tables 3 and 4 one can see that the method gives a substantially fewer number of iterations with respect to the original (unpreconditioned) conjugate gradient method.

2. In a case $\mu = 0,\ \nu = 2$ the number of iterations for any refinement factor k is optimal. Analyzing the condition (5.4) one can see that the square root of maximum eigenvalues in Table 5 is less than the corresponding value of ν. For example, if we choose the polynomials degree as $\mu = 0,\ \nu = 2$, then for any value of refinement parameter k we have

$$k = 5,\ (\max\{2.241, 2.804\})^{1/2} = (2.804)^{1/2} = 1.6745149 < 2,$$
$$k = 10,\ (\max\{2.405, 2.815, 2.943\})^{1/2} = (2.943)^{1/2} = 1.7155174 < 2,$$
$$k = 15,\ (\max\{2.720, 2.910, 2.978\})^{1/2} = (2.978)^{1/2} = 1.728294 < 2,$$
$$k = 20,\ (\max\{2.077, 2.855, 2.892, 2.966\})^{1/2} = (2.966)^{1/2} = 1.7222079 < 2,$$
$$k = 25,\ (\max\{2.670, 2.917, 2.957, 2.964\})^{1/2} = (2.964)^{1/2} = 1.7216271 < 2.$$

Table 4. PCG method, W-cycle

k	Number of levels	Number of unknowns	μ	ν	Number of iterations	Average reduction factor
5	4	91	0	2	13	0.3210
			0	3	14	0.3604
			1	2	15	0.3952
			1	3	18	0.4669
10	5	331	0	2	13	0.3356
			0	3	18	0.4532
			1	2	18	0.4703
			1	3	24	0.5560
15	5	721	0	2	14	0.3614
			0	3	19	0.4854
			1	2	20	0.5052
			1	3	27	0.5952
20	6	1261	0	2	14	0.3670
			0	3	22	0.5274
			1	2	26	0.5929
			1	3	37	0.6903
25	6	1951	0	2	14	0.3518
			0	3	23	0.5452
			1	2	27	0.6018
			1	3	38	0.7029

Table 5. Examples of lower and upper bounds of the eigenvalues computed of the matrices $M^{(k)^{-1}}A^{(k)}$

k	μ	ν	Level number	Minimum eigenvalue	Maximum eigenvalue	Condition number
5	0	2	2	1.0000	2.241	2.241
			3	0.8534	2.804	3.286
10	0	2	2	1.0000	2.405	2.405
			3	0.8298	2.815	3.392
			4	0.7034	2.943	4.184
15	0	2	2	1.0000	2.720	2.720
			3	0.7862	2.910	3.701
			4	0.6699	2.978	4.445
20	0	2	2	1.0000	2.077	2.077
			3	0.8774	2.855	3.254
			4	0.7193	2.892	4.020
			5	0.6381	2.966	4.648
25	0	2	2	1.0000	2.670	2.670
			3	0.7929	2.917	3.679
			4	0.6722	2.957	4.398
			5	0.6037	2.964	4.909

Moreover, it is readily seen from (4.6) and (4.7) that in this case ($\mu = 0$) the method has an optimal order of complexity also since $\rho = 3$ and $\theta < \rho^{-1/4}$.

3. As it can be seen from Table 3 the number of iterations for the method (for $\mu = 0$, $\nu = 1$) grows very slowly with respect of the number of unknowns.

4. Another interesting observation was also made, namely that the behaviour of the number of iterations in numerical experiments performed for various domains Ω, a triangular, a square, a L-shape and a polygonal domain, for instance, is similar.

Based on the results of experiments it can be recommended to use the V-cycle version of the suggested method as it gives still a reasonable number of iterations (nearly optimal) and the total computational cost is proportional to the number of unknowns on the fine

mesh (optimal), and, moreover, it's programming realization is simpler than for the W-cycle versions.

As a final conclusion, we have found that algebraic multilevel iteration method, applied to the finite-element matrices, leads to iterative methods of a nearly optimal order of the computational complexity and rate of convergence. However, the method is only applied for special finite-element meshes, i.e., the method does not work on the isosceles right triangular mesh and on triangulations very close to this. It seems that it is not a strong restriction on the suggested method. However, it turns out that in a case of a strong anisotropy in a one direction the rate of convergence of the method applied to the system (1.1), arising from finite-element approximation on non-regular finite-element mesh, has a similar behavior to the one in a case of the isosceles right triangular mesh. The future investigation will be directed finding a way of avoiding the condition on the mesh.

Acknowledgments

This research was done during a visit of the second author to the University of Nijmegen. He is grateful to Professor Owe Axelsson, who has made the visit possible, and to all of the staff at the Department of Mathematics for their hospitality. The authors would like to thank Dr. M.Neytcheva for assistance in the software implementation of the algorithms. Comment by some anonymous referees helped to improve the presentation of the paper.

References

[1] O. Axelsson, *The method of diagonal compensation of reduced matrix entries and multilevel iteration*, J. Comput. and Applied Mathematics, 38 (1991), pp. 31–43.

[2] O. Axelsson and P.S. Vassilevski, *Algebraic multilevel preconditioning methods I*, Numer. Math., 56 (1989), pp. 157–177.

[3] O. Axelsson and P.S. Vassilevski, *Algebraic multilevel preconditioning methods II*, SIAM J. Numer. Anal., 27 (1990), pp. 1569–1590.

[4] O. Axelsson and V. Eijkhout, *The nested recursive two-level factorization method for nine-point difference matrices*, SIAM J. Sci. Stat. Comp., 12 (1991), pp. 1373–1400.

[5] O. Axelsson and L. Kolotilina, *Diagonally compensated reduction and related preconditioning methods*, Report 9117, Catholic University, Nijmegen, Netherlands, 1991, cappeared later in: Numer. Linear Algebra with Applications. 1 (1994), pp 155–178.

[6] O. Axelsson and M. Neytcheva, *Algebraic multilevel iteration method for Stieltjes matrices*, Numer. Lin. Alg. Appl., 1 (1994), pp. 213-236.

[7] C.W. Brand, *An incomplete factorization preconditioning using repeated red-black ordering* , Numer. Math., 61 (1992), pp. 433–454.

[8] F.R. Gantmakher, *The Theory of Matrices*, Vol 1 & 2, New York, Chelsea, 1959.

[9] A. George, and J.W-H. Liu, *Computer Solution of Large Sparse Positive Definite Systems*, Prentice-Hall, Inc., 1981.

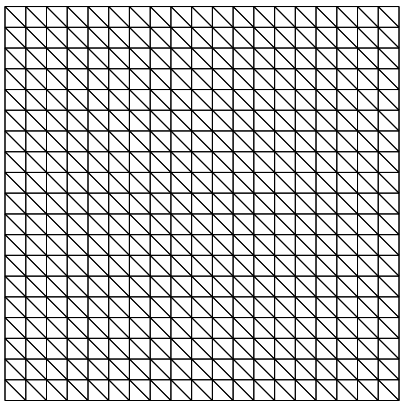
[10] Yu.R.Hakopian and Yu.A.Kuznetsov, *Algebraic multigrid substructuring preconditioners on triangular grids*, Sov. J. Numer. Anal. Math. Modelling, 6 (1989), pp. 453–483.

[11] Yu.A. Kuznetsov, *Algebraic Multigrid Domain Decomposition Methods*, Sov. J. Numer. Anal. Math. Modelling, 4 (1989), pp. 351–380.

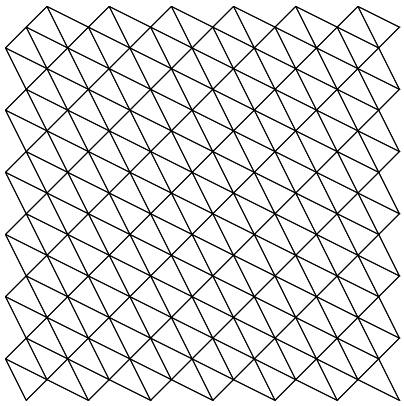
[12] P.S. Vassilevski, *Hybrid V-cycle algebraic multilevel preconditioners*, Math. Comp., 58 (1992), pp. 489–512.

Appendix A

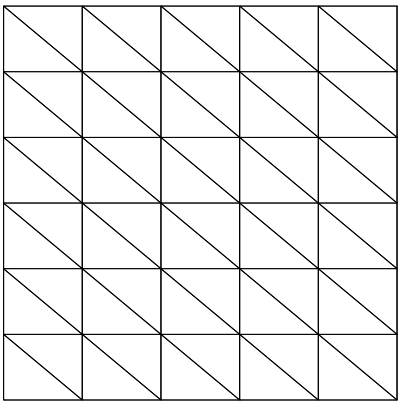
Step by step modification of the isosceles right triangular mesh in the case (20×20)-mesh.



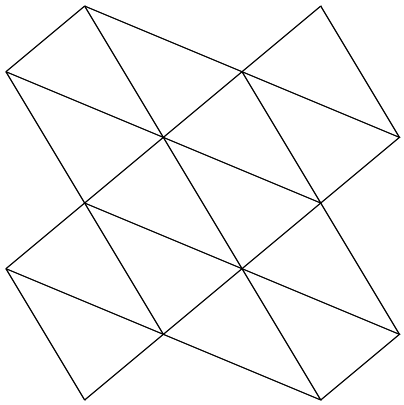
level=5



level=4



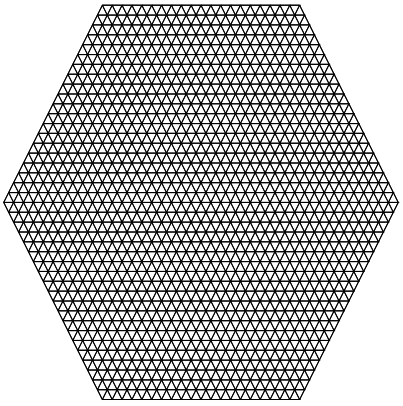
level=3



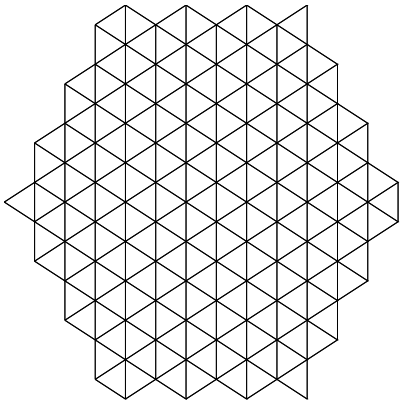
level=2

Appendix B

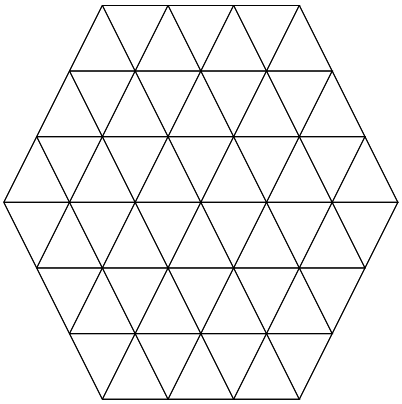
Step by step modification of the hexagonal mesh in the case $k = 10$.



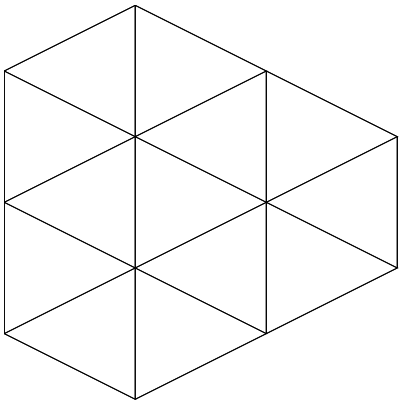
level=5



level=4



level=3



level=2