UCLA

COMPUTATIONAL AND APPLIED MATHEMATICS

Circulant Block-Factorization Preconditioning of Anisotropic Elliptic Problems

Ivan D. Lirkov
Svetozar D. Margenov
Lyudmil T. Zikatanov

August 1995

CAM Report 95-39

Department of Mathematics
University of California, Los Angeles
Los Angeles, CA. 90024-1555

Circulant block–factorization preconditioning of anisotropic elliptic problems

Ivan D. Lirkov Svetozar D. Margenov Lyudmil T. Zikatanov

Abstract

The recently introduced circulant block–factorization preconditioners are studied. The general approach is first formulated for the case of block tridiagonal sparse matrices. Then an estimate of the relative condition number for a model anisotropic Dirichlet boundary value problem is derived in the form $\kappa < \sqrt{2\varepsilon}(n+1)+2$, where $N = n^2$ is the size of the discrete problem, and ε stands for the ratio of the anisotropy. Various numerical tests demonstrating the behaviour of the circulant block–factorization preconditioners for anisotropic problems are presented.

AMS Subject Classification: 65F10, 65N20

Key words: preconditioning, circulant matrices, elliptic problems, anisotropy, FFT, block-tridiagonal matrices, conjugate gradients.

1 Introduction

This paper is concerned with the numerical solution of anisotropic second order elliptic boundary value problems. Using finite differences or finite elements, such problems generally are reduced to linear systems of the form $Au = b$, where A is a sparse matrix. We consider here symmetric and positive definite problems. We assume also, that A is a large scale matrix. It is well known, that in this case the iterative solvers based on the preconditioned conjugate gradient (PCG) method are the best way to solve the linear algebraic system. The key problem is how to construct the preconditioning matrix M. The general strategy of the efficient preconditioning can be formulated by the following goal: to minimize the relative condition number $\kappa(M^{-1}A)$ on a given class of preconditioning matrices, for which it is possible to solve efficiently the preconditioned system of equations $Mv = w$ for given vectors w.

The class of the preconditioning matrices studied in this paper contains block–tridiagonal matrices, which blocks are circulants. This approach in particular leads to the recently introduced (see [16]) circulant block–factorization (CBF) preconditioners. The CBF preconditioners incorporate some of the advantages of the block–incomplete LU factorization methods and the block–circulant methods. We study here the robustness of the CBF preconditioners for anisotropic second order elliptic boundary value problems.

The incomplete LU factorization of the matrix is one of the most popular classical preconditioning technique, see e.g. [2], [3], [9], [11]. The main idea of these methods is to approximate the exact factors in the Choleski (LU) factorization of the given sparse matrix such that the resulting approximate (lower and upper) triangular factors L and U are with a certain sparse structure. The basic results for the incomplete factorization (ILU) methods are proved for the case of M–matrices. It is known, e.g., that for some of the pointwise ILU preconditioners (see [11]) holds the estimate $\kappa(C^{-1}A) = O(\sqrt{N})$, where N is the size of the discrete problem. The basic general schemes of the block–ILU methods were proposed in [7], [1], [4],[19]. The block-ILU methods converge slower than the pointwise ones, which is an disadvantage. The advantage is that they provide highly parallel algorithms.

Another class of preconditioners based on a diagonal by diagonal averaging of the block entries of a given matrix A to form a *block–circulant* approximation C was proposed in [6] (see also [13], [14] and [20]). This leads to an improper in general approximation of the original Dirichlet boundary conditions with periodic ones. The usage of the block–circulant approximations is motivated by their fast inversion based on the FFT. For the model problem, it is shown that the block–circulant preconditioner can be constructed such that $\kappa(C^{-1}A) = O(\sqrt{N})$ which is asymptotically the same as for certain (modified) ILU type preconditioners. The block–circulant preconditioners are highly parallelizable, see e.g. [15] and [17], but they are substantially sensitive with respect to possible high variation of the coefficients of the given elliptic operator. In this respect they do not provide obvious advantages over the more classical incomplete block-factorization preconditioners.

The sensitivity of the block–circulant approximations with respect to possible high variation of the problem coefficients was relaxed in the recently proposed ([16]) circulant block–factorization preconditioner. The idea of this preconditioning technique is to average the coefficients of the given differential operator only along one of the coordinate directions (say, "y"). Thus we will give reasonable relative condition number if we have moderately varying coefficients in the y–direction. This preconditioning technique incorporates the circulant approximations into the framework of the LU block–factorization. The computational efficiency and parallelization of the resulting algorithm is as high as of the block circulant one ([6], [17]). It is proved in [16], that for the model problem ($\Delta u = f$) in a rectangle, the relative condition number $\kappa(M^{-1}A) = O(\sqrt{N})$, i.e, we have the same estimate as for the methods mentioned above.

The goal of the present paper is to study the convergence of the circulant block–factorization preconditioners for Dirichlet boundary value problems with (possibly strong) anisotropy. This is one of the important benchmark problems for the robustness of the iterative methods (see, e.g., in [5], [8], [12], [18]). Here we will consider the case of anisotropy with a fixed dominating direction (e.g., $\varepsilon < 1$).

The remainder of this paper is organized as follows. In section §2 we describe the general form of the circulant block–factorization method. A model analysis of the relative condition number based on exact spectral analysis is presented in §3. In §4 we show numerical tests illustrating various aspects of the behaviour of the circulant block–factorization preconditioners.

2 Circulant block-factorization preconditioner

We consider the following anisotropic 2D elliptic problem,

$$\begin{aligned} -\frac{\partial}{\partial x}\left(a(x,y)\frac{\partial u}{\partial x}\right)-\frac{\partial}{\partial y}\left(b(x,y)\frac{\partial u}{\partial y}\right) &= f, \quad \forall (x,y)\in\Omega, \\ 0<\sigma_{\min}\le a(x,y),\, b(x,y) &\le \sigma_{\max}, \\ u(x,y)=0, \qquad \forall (x,y)\in\Gamma &= \partial\Omega, \end{aligned} \tag{1}$$

where $\Omega = (0,1)\times(0,1)$ is covered by a uniform square mesh ω_h, with a size $h = 1/(n+1)$ for a given integer $n \geq 1$. Problem (1) is approximated by the standard 5-point finite difference stencil (the finite element method for linear triangular elements results to a similar result). This discretization leads to a system of linear algebraic equations

$$Au = f. \tag{2}$$

If the grid points are ordered along, e.g., the y-grid lines, the matrix A admits a block–tridiagonal structure (with blocks formed by the unknowns within a given grid–line). A can be written in the following form

$$A = tridiag(-A_{i,i-1}, A_{i,i}, -A_{i,i+1}) \qquad i=1,2,\ldots,n,$$

where

$$\begin{aligned} A_{i,i} = tridiag(-a_{j,j-1}, a_{j,j}, -a_{j,j+1}), &\quad j=(i-1)n+1,\ldots,in, \quad i=1,2,\ldots,n, \\ A_{i,i+1} = diag(a_{j,j+n}), &\quad j=(i-1)n+1,\ldots,in, \quad i=1,\ldots,n-1, \\ A_{i,i-1} = diag(a_{j,j-n}), &\quad j=(i-1)n+1,\ldots,in, \quad i=2,\ldots,n. \end{aligned}$$

The coefficients $a_{i,j}$ are positive and $a_{j,j} \geq a_{j,j-1} + a_{j,j+1} + a_{j,j+n} + a_{j,j-n}$, i.e., the matrix A satisfies the maximum principle.

Using the standard LU factorization procedure, we can first split $A = D - L - U$ into its block-diagonal and (negative) strictly block-triangular parts respectively. Then the *exact* block-factorization can be written in the form,

$$A = (X - L)(I - X^{-1}U),$$

where the blocks of $X = diag(X_1, X_2, \ldots, X_n)$ are to be determined. We have

$$A = X - L - U + LX^{-1}U.$$

Therefore

$$X = D - LX^{-1}U,$$

which gives the recursion

$$X_1 = A_{1,1}, \text{ and } X_i = A_{i,i} - A_{i,i-1}X_{i-1}^{-1}A_{i-1,i}, \qquad i = 2, \dots, n \tag{3}$$

It is well-known that the above factorization exists if A is, for example, positive definite. This factorization can be used to solve system (2). This requires solution of linear systems involving the blocks X_i. Note that $\{X_i\}$ are in general full matrices and the resulting (direct) Gaussian elimination algorithm can become too expensive. The common idea of the block-ILU factorization methods is to approximate X_i (or X_i^{-1}) by sparse (band) matrices. The idea explored in [16] instead, is to first modify the original matrix A in such a way that the resulting matrices from the exact factorization of the thus modified matrix (in place of X_i) are now circulant.

Let us recall that a circulant matrix C has the form $(c_{k,j}) = \left(c_{(j-k)\mathrm{mod} m}\right)$, where m is the size of C. Let us also denote for any given coefficients $(c_0, c_1, \dots, c_{m-1})$ by $C = (c_0, c_1, \dots, c_{m-1})$ the circulant matrix

$$\begin{bmatrix} c_0 & c_1 & c_2 & \dots & c_{m-1} \\ c_{m-1} & c_0 & c_1 & \dots & c_{m-2} \\ \vdots & \vdots & \vdots & & \vdots \\ c_1 & c_2 & \dots & c_{m-1} & c_0 \end{bmatrix}.$$

Any circulant matrix can be factorized as

$$C = F\Lambda F^*, \tag{4}$$

where Λ is a diagonal matrix containing the eigenvalues of C, and F is the Fourier matrix

$$F = \frac{1}{\sqrt{m}} \left\{ e^{2\pi \frac{jk}{m} i} \right\}_{0 \le j,k \le m-1}.$$

Here i stands for the imaginary unit.

The general form of the CBF preconditioning matrix C for the matrix A is defined by

$$C = tridiag(-C_{i,i-1}, C_{i,i}, -C_{i,i+1}) \qquad i = 1, 2, \dots, n,$$

where $C_{i,j} = Circulant(A_{i,j})$ is some given circulant approximation (to be specified later) of the corresponding block $A_{i,j}$. Realizing the algorithm we use the exact block LU factorization for the preconditioner C. Note that the recursion (3), performed for C, is closed in the class of circulant matrices. That is, the corresponding blocks X_i are circulant and therefore the solution of the preconditioned system involving the matrix C can be performed efficiently based on the FFT using the representation (4) for the blocks $\{X_i\}$.

Following the notations from [16], in the present paper we use the second CBF algorithm, originally denoted by CBF2. The approach of defining block circulant approximations in this case can be interpreted as simultaneous averaging of the matrix coefficients and changing of the Dirichlet boundary conditions to periodic ones. The following mean-values are introduced,

$$\begin{aligned}
\bar{a}_{i,i\pm 1} &= \frac{1}{n}\sum_{j=(i-1)n+1}^{in} a_{j,j\pm n} \\
\bar{a}_{i,i,-1} &= \frac{1}{n}\sum_{j=(i-1)n+2}^{in} a_{j,j-1} + \frac{d_i}{n} \\
\bar{a}_{i,i,1} &= \frac{1}{n}\sum_{j=(i-1)n+1}^{in-1} a_{j,j+1} + \frac{d_i}{n} \\
\bar{a}_{i,i,0} &= \frac{1}{n}\sum_{j=(i-1)n+1}^{in} a_{j,j},
\end{aligned}$$

where

$$d_i = \min(d_i^{(1)}, d_i^{(n)}),$$

and where

$$\begin{aligned}
d_1^{(1)} &= \frac{a_{1,1} - a_{1,2} - a_{1,n+1}}{2}, \\
d_i^{(1)} &= a_{(i-1)n+1,(i-1)n+1} - a_{(i-1)n+1,(i-2)n+1} - a_{(i-1)n+1,in+1} - a_{(i-1)n+1,(i-1)n+2}, \\
&\quad \forall i = 2, \ldots, n-1, \\
d_n^{(1)} &= \frac{a_{(n-1)n+1,(n-1)n+1} - a_{(n-1)n+1,(n-2)n+1} - a_{(n-1)n+1,(n-1)n+2}}{2},
\end{aligned}$$

and

$$\begin{aligned}
d_1^{(n)} &= \frac{a_{n,n} - a_{n,n-1} - a_{n,2n}}{2}, \\
d_i^{(n)} &= a_{in,in} - a_{in,(i-1)n} - a_{in,(i+1)n} - a_{in,in-1}, \qquad \forall i = 2, \ldots, n-1, \\
d_n^{(n)} &= \frac{a_{n^2,n^2} - a_{n^2,(n-1)n} - a_{n^2,n^2-1}}{2}.
\end{aligned}$$

The circulant blocks are defined explicitly by the formulas

$$\begin{aligned}
C_{i,i\pm 1} &= (\bar{a}_{i,i\pm 1}, 0, \ldots, 0) = \bar{a}_{i,i\pm 1} I, \\
C_{i,i} &= (\bar{a}_{i,i,0}, -\bar{a}_{i,i,1}, 0, \ldots, 0, -\bar{a}_{i,i,-1}).
\end{aligned}$$

Then the CBF preconditioning matrix C has the form

$$C = tridiag(-C_{i,i-1}, C_{i,i}, -C_{i,i+1}).$$

It is easy to see that the matrix C defined by the above CBF algorithm is symmetric and positive definite (see for more details in [16]).

3 Model analysis of the condition number

We consider in this section the model 2D elliptic problem

$$\begin{aligned} -u_{xx} - \varepsilon u_{yy} &= f(x,y), \qquad \forall (x,y) \in \Omega, \\ u(x,y) &= 0, \qquad \forall (x,y) \in \Gamma = \partial\Omega, \end{aligned} \tag{5}$$

where $\Omega = (0,1) \times (0,1)$ and ω_h is a uniform square mesh with a size $h = 1/(n+1)$ for a given integer $n \geq 1$. Problem (5) is approximated by the standard 5-point finite difference stencil. This discretization leads to a system of linear algebraic equations

$$Au = f. \tag{6}$$

Following the standard procedure we order the grid points along the y-grid lines. Then, the matrix A admits a block–tridiagonal structure and can be written in the form $A = tridiag(-I, B, -I)$, where $B = tridiag(-\varepsilon, 2+2\varepsilon, -\varepsilon)$ and for the corresponding CBF preconditioner we get $M = tridiag(-I, D, -I)$ with $D = circulant(2+2\varepsilon, -\varepsilon, 0, \ldots 0, -\varepsilon)$.

Consider also $T = tridiag(-1, 2, -1)$ and $C = (2, -1, 0, \ldots 0, -1)$. Then we have

$$\begin{aligned} A &= \varepsilon I \otimes T + T \otimes I, \\ M &= \varepsilon I \otimes C + T \otimes I. \end{aligned}$$

We estimate in the next lemma the condition number $\kappa(M^{-1}A)$ by the eigenvalues of eigenproblems of a reduced size n.

Lemma 1 *The condition number of preconditioned system satisfies the estimate*

$$\kappa(M^{-1}A) < \frac{\max_k \lambda((\varepsilon T + \alpha_k I)^{-1}(\varepsilon C + \alpha_k I))}{\min_k \lambda((\varepsilon T + \alpha_k I)^{-1}(\varepsilon C + \alpha_k I))},$$

where $\alpha_k = 4\sin^2 \frac{k\pi}{2(n+1)}$, $k = 1, 2, \ldots n$.

Proof: To estimate the relative condition number of the CBF preconditioner we shall analyze the eigenvalues of the generalized eigenvalue problem

$$(\varepsilon I \otimes C + T \otimes I)w = \lambda(\varepsilon I \otimes T + T \otimes I)w. \tag{7}$$

It is easy to compute the eigenvalues of the matrix T, that are expressed by $\alpha_k(T) = 4\sin^2 \frac{k\pi}{2(n+1)}$, $k = 1, 2, \ldots n$. Then the matrix T can be factored in the form $T = V^T \mathcal{A} V$, where $\mathcal{A}$ is the diagonal matrix of the eigenvalues of T, the matrix V has the corresponding eigenvectors of T and V is orthogonal matrix (i.e., $V^T V = I$). Following the introduced notations we rewrite (7) in the form

$$(\varepsilon(V^T V) \otimes C + (V^T \mathcal{A} V) \otimes I)w = \lambda(\varepsilon(V^T V) \otimes T + (V^T \mathcal{A} V) \otimes I)w. \tag{8}$$

$$(V^T \otimes I)(\varepsilon I \otimes C + \mathcal{A} \otimes I)(V \otimes I)w = \lambda(V^T \otimes I)(\varepsilon I \otimes T + \mathcal{A} \otimes I)(V \otimes I)w.$$

Denote by $u = (V \otimes I)w$ and obtain

$$(\varepsilon I \otimes C + \mathcal{A} \otimes I)u = \lambda(\varepsilon I \otimes T + \mathcal{A} \otimes I)u. \tag{9}$$

It follows from (9) that the eigenvalues of (7) are solutions of the split system of eigenvalue problems

$$(\varepsilon C + \alpha_k I)u_k = \lambda(\varepsilon T + \alpha_k I)u_k. \tag{10}$$

Obviously, the statement of the lemma follows directly from (10). ■

We will use in the rest part of this section the determinants Δ_i, defined for a fixed eigenvalue α_k.

Definition 1 *We denote by $\Delta_i = \Delta_i(\alpha_k) = det(tridiag(-1, 2+\rho, -1))$, where $\rho = \alpha_k/\varepsilon$, and i stands for the dimension of the determinant.*

Now, we derive directly from the definition the recurrence equation

$$\Delta_i = (2+\rho)\Delta_{i-1} - \Delta_{i-2}, \tag{11}$$

where $\Delta_0 = 1$ and $\Delta_1 = 2 + \rho$.

In the next lemma we will find explicitly the eigenvalues of a generalized eigenvalue problem of the form involved in Lemma 1.

Lemma 2 *The matrix $(T+\rho I)^{-1}(C+\rho I)$ has exactly two eigenvalues different from unity, and they are*

$$\lambda_{1,2} = 1 + \frac{1 \pm \Delta_{n-1}}{\Delta_n}.$$

Proof: We have to consider the eigenvalue problem (10) (recall that $\rho = \alpha_k/\varepsilon$), which can be rewritten in the form

$$[(\lambda - 1)\rho I + \lambda T - C]\, u_k = 0.$$

The last equation is equivalent to the following algebraic equation about the characteristic polynomial, i.e.

$$P(\lambda) = (\lambda-1)^n \begin{vmatrix} 2+\rho & -1 & 0 & 0 & \cdots & 0 & \frac{1}{\lambda-1} \\ -1 & 2+\rho & -1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 2+\rho & -1 & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{1}{\lambda-1} & 0 & 0 & 0 & \cdots & -1 & 2+\rho \end{vmatrix} = 0. \tag{12}$$

To solve the equation (12) we factorize the matrix of the above determinant.

$$\begin{pmatrix} 2+\rho & -1 & 0 & 0 & \cdots & 0 & \frac{1}{\lambda-1} \\ -1 & 2+\rho & -1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 2+\rho & -1 & \cdots & 0 & 0 \\ \hdotsfor{7} \\ \frac{1}{\lambda-1} & 0 & 0 & 0 & \cdots & -1 & 2+\rho \end{pmatrix} =$$

$$\begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ x_1 & 1 & 0 & \cdots & 0 \\ 0 & x_2 & 1 & \cdots & 0 \\ \hdotsfor{5} \\ \eta_1 & \eta_2 & \eta_3 & \cdots & 1 \end{pmatrix} \begin{pmatrix} y_1 & z_1 & 0 & \cdots & \xi_1 \\ 0 & y_2 & z_2 & \cdots & \xi_2 \\ 0 & 0 & y_3 & \cdots & \xi_3 \\ \hdotsfor{5} \\ 0 & 0 & 0 & \cdots & y_n \end{pmatrix}, \tag{13}$$

where

$$x_i = -\frac{1}{y_i} \quad i = 1, \dots n-2$$

$$z_i = -1 \quad i = 1, \dots n-2$$

$$y_1 = 2 + \rho$$

$$y_i = 2 + \rho - \frac{1}{y_{i-1}} \quad i = 2, \dots n-1$$

$$\xi_1 = \frac{1}{\lambda - 1}$$

$$\xi_i = \frac{\xi_{i-1}}{y_{i-1}} = \frac{1}{(\lambda-1)\prod_{j=1}^{i-1} y_j} \quad i = 2, \dots n-1$$

$$\xi_{n-1} = -1 + \frac{\xi_{i-1}}{y_{i-1}} = -1 + \frac{1}{(\lambda-1)\prod_{j=1}^{n-2} y_j}$$

$$\eta_i = \frac{\xi_i}{y_i} \quad i = 1, \dots n-1$$

$$\eta_i = \frac{1}{(\lambda-1)\prod_{j=1}^{i} y_j} \quad i = 1, \dots n-2$$

$$\eta_{n-1} = -\frac{1}{y_{n-1}} + \frac{1}{(\lambda-1)\prod_{j=1}^{n-1} y_j}$$

$$y_n = 2 + \rho - \sum_{i=1}^{n-1} \xi_i \eta_i$$

Using that $\Delta_i = \prod_{j=1}^{i} y_j$ for $i = 1, 2, \ldots n-1$ we get the expressions

$$y_n = 2 + \rho - \sum_{i=1}^{n-1} \frac{1}{(\lambda-1)^2 \Delta_i \Delta_{i-1}} - \frac{1}{y_{n-1}} + \frac{2}{(\lambda-1)\Delta_{n-1}}$$

and

$$\begin{aligned} \frac{1}{(\lambda-1)^n} P(\lambda) &= \prod_{j=1}^{n} y_j = \Delta_{n-1} y_n = \\ &= (2+\rho)\Delta_{n-1} - \Delta_{n-1} \sum_{i=1}^{n-1} \frac{1}{(\lambda-1)^2 \Delta_i \Delta_{i-1}} - \Delta_{n-2} + \frac{2}{\lambda-1} = \\ &= \Delta_n - \frac{\Delta_{n-1}}{(\lambda-1)^2} \sum_{i=1}^{n-1} \frac{1}{\Delta_i \Delta_{i-1}} + \frac{2}{\lambda-1}. \end{aligned} \tag{14}$$

Now we determine Δ_i from the recurrence equation (11), and find

$$\Delta_i = \frac{1 - \psi^{2i+2}}{\psi^i (1 - \psi^2)}, \tag{15}$$

where ψ is one of the roots (to be chosen later) of the square equation

$$\psi^2 - (2+\rho)\psi + 1 = 0. \tag{16}$$

Now, we simplify (14) as follows

$$\begin{aligned} \Delta_{n-1} \sum_{i=1}^{n-1} \frac{1}{\Delta_i \Delta_{i-1}} &= \frac{1 - \psi^{2n}}{\psi^{n-1}(1-\psi^2)} \sum_{i=1}^{n-1} \frac{\psi^{2i-1}(1-\psi^2)^2}{(1-\psi^{2i+2})(1-\psi^{2i})} = \\ &= \frac{1-\psi^{2n}}{\psi^n} \sum_{i=1}^{n-1} \left(\frac{1}{1-\psi^{2i}} - \frac{1}{1-\psi^{2i+2}} \right) = \\ &= \frac{1-\psi^{2n}}{\psi^n} \left(\frac{1}{1-\psi^2} - \frac{1}{1-\psi^{2n}} \right) = \\ &= \frac{1-\psi^{2n-2}}{\psi^{n-2}(1-\psi^2)} \end{aligned} \tag{17}$$

Substituting (15) and (17) into (14) we get the final presentation of the polynomial $P(\lambda)$, and of the related equation in the form

$$P(\lambda) = (\lambda-1)^{n-2} \left[(\lambda-1)^2 \frac{1-\psi^{2n+2}}{\psi^n(1-\psi^2)} + 2(\lambda-1) - \frac{1-\psi^{2n-2}}{\psi^{n-2}(1-\psi^2)} \right] = 0. \tag{18}$$

Obviously, the above equation has $(n-2)$ roots equal to unit. The last two roots are solutions of the square equation in the brakes, and have the form

$$\lambda_{1,2} = 1 + \frac{\psi^n(1-\psi^2) \pm \psi(1-\psi^{2n})}{1-\psi^{2n+2}}.$$

We rewrite now the last equality into the terms of Δ_i, and derive the representation

$$\lambda_{1,2} = 1 + \frac{1 \pm \Delta_{n-1}}{\Delta_n}, \tag{19}$$

which completes the proof of the lemma. ■

Let us remind, that the goal of this section is to estimate the relative condition number of the CBF preconditioner in the terms of n and ε. This result is the contents of the next theorem.

Theorem 1 *The relative condition number of the CBF preconditioner for the model problem (6) satisfies the inequality*

$$\kappa(M^{-1}A) < \sqrt{2\varepsilon}(n+1) + 2.$$

Proof: Combining the results from Lemma 1 and Lemma 2 we get the estimate

$$\kappa(M^{-1}A) < \frac{\max_k \lambda_1}{\min_k \lambda_2},$$

where $\lambda_{1,2}$ are given by (19), depending on k, as $\rho = \alpha_k/\varepsilon$. Now we chose ψ to be the larger root of (16), which implies $\psi > 1$. It follows from (15), that Δ_i can be expanded in the form

$$\Delta_i = \frac{1}{\psi^i} + \frac{1}{\psi^{i-2}} + \ldots + \psi^{i-2} + \psi^i.$$

Hence

$$\Delta_n = \psi\Delta_{n-1} + \frac{1}{\psi^n},$$

and therefore

$$\begin{aligned} \lambda_2 &= 1 + \frac{1-\Delta_{n-1}}{\Delta_n} = 1 + \frac{1-\Delta_{n-1}}{\psi\Delta_{n-1} + \frac{1}{\psi^n}} = \frac{(\psi-1)\Delta_{n-1} + 1 + \frac{1}{\psi^n}}{\psi\Delta_{n-1} + \frac{1}{\psi^n}} \\ &> \frac{(\psi-1)\Delta_{n-1}}{(\psi+1)\Delta_{n-1}} = \frac{\psi-1}{\psi+1} = \frac{1}{\sqrt{1+4/\rho}}, \end{aligned} \tag{20}$$

and

$$\lambda_1 < 2. \tag{21}$$

Combining the estimates (20) and (21) we get the estimate for the relative condition number

$$\kappa(M^{-1}A) < 2\sqrt{1+\frac{4}{\rho}} = 2\sqrt{1+\frac{4\varepsilon}{\alpha_k}}.$$

At the end, we use the inequality $\alpha_k = 4\sin^2\frac{k\pi}{2(n+1)} > \frac{8}{(n+1)^2}$, and obtain the final result of the theorem, namely

$$\kappa(M^{-1}A) < \sqrt{4 + 2\varepsilon(n+1)^2} < \sqrt{2\varepsilon}(n+1) + 2. \tag{22}$$

■

Remark 1 *It is clear that for $\varepsilon = 1$ the last theorem gives the estimate*

$$\kappa(M^{-1}A) < \sqrt{2}(n+2)$$

for the Poisson model problem. This estimate improves considerably the constant in the related result from [16], that was obtained using the estimation technique from [6].

4 Numerical tests

The numerical tests presented in this section illustrate the convergence rate of the CBF algorithm for anisotropic elliptic problems. The computations are done with double precision on a SUN SPARC Station 2. The iteration stopping criterion is $||r^{N_{it}}||/||r^0|| < 10^{-6}$, where r^j stands for the residual at the jth iteration step of the preconditioned conjugate gradient method.

Example 1. The first test problem is the model problem analyzed in the previous section, i.e.

$$\begin{aligned} -u_{xx} - \varepsilon u_{yy} &= f(x,y), \qquad \forall (x,y) \in \Omega = (0,1) \times (0,1), \\ u(x,y) &= 0, \qquad \forall (x,y) \in \Gamma = \partial\Omega. \end{aligned} \tag{23}$$

Table 1 shows the number of iterations as a measure of the convergence rate, where the mesh size n and the ratio of anisotropy ε are varied. As we proved in Theorem 1, the CBF algorithm is characterized by the estimate $\kappa(M^{-1}A) < \sqrt{2\varepsilon}(n+1)+2$. The presented data demonstrate a behaviour of the convergence, that confirms the high accuracy of the estimate of the relative condition number. In particular, the data from the first column ($\varepsilon = 10$) shows very clearly the importance of the ordering the unknowns along the direction of the weak anisotropy. Let us remind, that the proper ordering for the block-ILU algorithms is just the opposite, i.e. along the direction of the strong anisotropy (see, e.g., in [19]).

Table 1: Number of iterations for the model problem.

n	$\varepsilon = 10.$	$\varepsilon = 1.$	$\varepsilon = 0.1$	$\varepsilon = 0.01$	$\varepsilon = 0.001$	$\varepsilon = 0.0001$	$\varepsilon = 0.00001$
8	15	10	7	5	5	5	5
16	19	13	9	5	4	4	4
32	25	17	10	7	5	4	4
64	31	20	13	8	5	4	3
128	42	28	17	11	7	4	3
256	56	34	22	14	9	6	3
512	77	47	28	18	11	7	4

We consider further test problems with variable coefficients in the form

$$\begin{aligned} -\frac{\partial}{\partial x}\left(a(x,y)\frac{\partial u}{\partial x}\right)-\varepsilon\frac{\partial}{\partial y}\left(b(x,y)\frac{\partial u}{\partial y}\right) &= f, \quad \forall(x,y)\in(0,1)\times(0,1), \\ u(x,y) &= 0, \qquad \forall(x,y)\in\Gamma=\partial\Omega. \end{aligned} \tag{24}$$

Example 2. Two test problems are considered in this example, where the problem coefficients have the form

$$(A) \qquad a(x,y)=b(x,y)=\begin{cases} 1 & x<0.5 \\ 100 & x>0.5 \end{cases};$$

$$(B) \qquad a(x,y)=b(x,y)=\begin{cases} 1 & x<0.5 \\ 0.01 & x>0.5 \end{cases}.$$

In these cases the coefficients have jump along the line $x = 0.5$. Table 2 and Table 3. show that this kind of coefficient jumps has a weak influence on the convergence of the CBF preconditioner. Such a behaviour of the number of the iterations is natural. Note that in these cases the ratio of anisotropy remains equal to ε in the subdomains, independently of the coefficients jump.

Table 2: Number of the iterations; Example 2-A.

n	$\varepsilon = 10.$	$\varepsilon = 1.$	$\varepsilon = 0.1$	$\varepsilon = 0.01$	$\varepsilon = 0.001$	$\varepsilon = 0.0001$	$\varepsilon = 0.00001$
8	15	11	8	6	6	6	6
16	19	12	9	6	6	6	6
32	24	16	10	7	6	6	6
64	35	20	13	8	6	6	6
128	43	27	17	11	7	6	6
256	58	34	22	14	9	6	6
512	75	46	29	18	11	8	6

Table 3: Number of the iterations; Example 2-B.

n	$\varepsilon = 10.$	$\varepsilon = 1.$	$\varepsilon = 0.1$	$\varepsilon = 0.01$	$\varepsilon = 0.001$	$\varepsilon = 0.0001$	$\varepsilon = 0.00001$
8	14	11	8	6	6	6	6
16	18	13	9	6	6	6	6
32	25	17	11	7	6	6	6
64	33	20	13	8	6	6	6
128	42	26	17	11	8	6	6
256	56	34	22	14	9	7	6
512	79	47	29	18	12	8	6

Example 3. In this example we consider the test problem (24) with

$$a(x,y)=1+\frac{1}{2}\sin(2\pi x), \quad b(x,y)=e^{x+y}.$$

The coefficient $a(x,y)$ is oscillating function of x. As in the previous example this does not changes the general behaviour of the iterative process. The second coefficient $b(x,y)$ varies moderately, and as a result, the number of the iterations shown in Table 4 is weakly increased.

Table 4: Number of iterations for the problem (24), where $a(x,y) = 1 + \frac{1}{2}\sin(2\pi x)$ and $b(x,y) = e^{x+y}$.

n	$\varepsilon = 10.$	$\varepsilon = 1.$	$\varepsilon = 0.1$	$\varepsilon = 0.01$	$\varepsilon = 0.001$	$\varepsilon = 0.0001$	$\varepsilon = 0.00001$
8	15	13	9	6	6	6	6
16	23	16	11	8	5	5	5
32	31	21	14	10	7	4	4
64	41	27	18	12	9	6	4
128	54	37	26	16	11	8	5
256	72	56	37	20	14	10	7
512	109	93	61	28	18	12	9

Example 4. In the last test example we consider coefficients in the form

$$a(x,y) = 1 + \frac{1}{2}\sin(2\pi(x+y)), \quad b(x,y) = e^{x+y}.$$

They differ a bit from the previous ones. The coefficient $b(x,y)$ is exactly the same, while now $a(x,y)$ is oscillating function of x and y. The corresponding number of iterations are presented in Table 5. The number of iterations is larger than in the previous example, that reflects the varying the oscillating coefficient as function of y.

Table 5: Number of iterations for the problem (24), where $a(x,y) = 1+\frac{1}{2}\sin(2\pi(x+y))$ and $b(x,y) = e^{x+y}$.

n	$\varepsilon = 10.$	$\varepsilon = 1.$	$\varepsilon = 0.1$	$\varepsilon = 0.01$	$\varepsilon = 0.001$	$\varepsilon = 0.0001$	$\varepsilon = 0.00001$
8	16	13	9	10	10	11	11
16	23	16	13	11	11	12	12
32	33	21	16	14	12	12	12
64	46	27	21	17	13	12	12
128	63	39	29	20	16	13	12
256	78	57	41	26	19	14	12
512	114	92	62	35	22	17	13

Acknowledgements

This paper was partially supported by the National Science Foundation grant INT-95-06184 and also in part by the Bulgarian Ministry of Education, Science and Technology under grant MM 417/94.

References

[1] O. Axelsson, A survey of vectorizable preconditioning methods for large scale finite element matrices, *Colloquium Topics in Applied Numerical Analysis, (J.G.Verwer, ed.)*, Syllabus 4, Center of Mathematics and Informatics (CMI), Amsterdam (1983), 21–47.

[2] O. Axelsson and V.A. Barker, *Finite Element Solution of Boundary Value Problems: Theory and Computations*, Academic Press, Orlando, Fl. (1983).

[3] O. Axelsson, S. Brinkkemper, and V.P. Il'in, On some versions of incomplete block-matrix factorization methods, *Lin.Alg.Appl.*, **38** (1984), 3–15.

[4] O. Axelsson, B. Polman, On approximate factorization methods for block-matrices suitable for vector and parallel processors, *Lin. Alg. Appl.*, **77** (1986), 3–26.

[5] O. Axelsson and V.L. Eijkhout, Robust vectorizable preconditioners for three-dimensional elliptic difference equations with anisotropy, *Algorithms and Applications on Vector and Parallel Computers*, (H.J.J. te Riele , Th.J. Dekker, and H. van der Vorst, eds.), North Holland, Amsterdam, 1987.

[6] R.H.Chan and T.F.Chan, Circulant preconditioners for elliptic problems, *J. Numerical Lin.Alg.Appl.*, **1** (Mar. 1992), 77–101.

[7] P. Concus, G.H. Golub, and G. Meurant, Block preconditioning for the conjugate gradient method, *SIAM J.Sci.Stat.Comput.*, **6** (1985), 220–252.

[8] I.S. Duff, G.A. Meurant, The effect of ordering on preconditioned conjugate gradients, **BIT**, **29** (1989), 635–657

[9] G.H. Golub and C.F. Van Loan, *Matrix Computations*, 2nd edition, Johns Hopkins Univ.Press, Baltimore (1989).

[10] J.E. Gunn, The solution of difference equations by semi-explicit iterative techniques, *SIAM J.Num.Anal.*, **2** (1965), 24–45.

[11] I. Gustafsson, A class of first-order factorization methods, **BIT**, **18** (1978), 142–156.

[12] W. Hackbusch, The frequency decomposition multi-grid method. 1. Application to anisotropic equations, *Numer. Math.*, **56** (1989), 219–245.

[13] S. Holmgren and K. Otto, Iterative solution methods for block-tridiagonal systems of equations, *SIAM J.Matr.Anal.Appl.*, **13** (1992), 863–886.

[14] T. Huckle, Some aspects of circulant preconditioners, *SIAM J.Sci. Comput.*, **14** (1993), 531–541.

[15] C. Van Loan, *Computational frameworks for the fast Fourier transform*, SIAM, Philadelphia (1992).

[16] I. Lirkov, S. Margenov, P.S. Vassilevski, Circulant block-factorization preconditioners for elliptic problems, *Computing*, **53** 1 (1994), 59–74.

[17] S.D. Margenov and I.T. Lirkov, Preconditioned conjugate gradient iterative algorithms for transputer based systems, *in Parallel and distributed processing, (K.Boyanov ed.)*, Sofia (1993), 406–415.

[18] S.D. Margenov and P.S. Vassilevski, Algebraic multilevel preconditioning of anisotropic elliptic problems, *SIAM J. Sci. Comp.*, **15** 5 (1994), 1026–1037.

[19] G. Meurant, A review on the inverse of symmetric tridiagonal and block tridiagonal matrices, *SIAM J.Matrix Anal.Appl.*, **13** (1992), 707–728.

[20] G. Strang, A proposal for Toeplitz matrix calculations, *Stud. Appl.Math.*, **74** (1986), 171–176.