On a two-level parallel MIC(0) preconditioning of Crouzeix-Raviart non-conforming FEM systems *

R. D. Lazarov[1] and S. D. Margenov[2]

[1] Department of Mathematics, Texas A& M University, College Station, TX, USA
Raytcho.Lazarov@radon.math.tamu.edu
[2] Central Laboratory of Parallel Processing, Bulgarian Academy of Sciences, Bulgaria
margenov@parallel.bas.bg

Abstract. In this paper we analyze a two-level preconditioner for finite element systems arising in approximations of second order elliptic boundary value problems by Crouzeix-Raviart non-conforming triangular linear elements. This study is focused on the efficient implementation of the modified incomplete LU factorization MIC(0) as a preconditioner in the PCG iterative method for the linear algebraic system. A special attention is given to the implementation of the method as a scalable parallel algorithm.

Key words: non-conforming FEM, preconditioning, parallel algorithms
AMS subject classifications: 65F10, 65N30

1 Introduction

In this paper we consider the elliptic boundary value problem

$$\begin{aligned} Lu \equiv -\nabla \cdot (a(x)\nabla u(x)) &= f(x) \text{ in } \quad \Omega, \\ u &= 0 \quad \text{on} \quad \Gamma_D, \\ (a(x)\nabla u(x)) \cdot n &= 0 \quad \text{on} \quad \Gamma_N, \end{aligned} \tag{1}$$

where Ω is a convex polygonal domain in $\mathbb{R}^2$, $f(x)$ is a given function in $L^2(\Omega)$, $a(x) = [a_{ij}(x)]_{i,j=1}^2$ is a symmetric and uniformly positive definite matrix in Ω, n is the outward unit vector normal to the boundary $\Gamma = \partial\Omega$, and $\Gamma = \bar{\Gamma}_D \cup \bar{\Gamma}_N$. We assume that the entries $a_{ij}(x)$ are piece-wise smooth functions on $\bar{\Omega}$. In the paper we use the terminology of the flow in porous media and we refer to u as a pressure and $-a(x)\nabla u(x)$ as a velocity vector.

The problem (1) can be discretized by the finite volume method, the Galerkin finite element method (conforming or non-conforming) or the mixed finite element method. Each of these methods has its advantages and disadvantages when the problem (1) is used in a particular application. For example, for application related to highly heterogeneous porous media the finite volume and mixed finite element methods have proven to be accurate and locally mass conservative. While applying the mixed FEM to problem (1) the continuity of the velocity normal to the boundary between two adjacent finite element could be enforced by Lagrange multipliers. In [2] Arnold and Brezzi have demonstrated that after the elimination of the unknowns representing the pressure and the velocity from the algebraic system the resulting Schur system for the Lagrange multipliers is equivalent to a discretization of (1) by Galerkin method using linear non-conforming finite elements. Namely, in[2] is shown that the lowest-order Raviart-Thomas mixed finite element approximations are equivalent to the usual Crouzeix-Raviart non-conforming linear finite element approximations when the non-conforming space is augmented with cubic bubbles. Further, such a relationship between the mixed and non-conforming finite element methods has been studied and simplified for various finite element

* This work was supported in part by the National Science Foundation under grant DMS 9973328 and by the gift-grant by Saudi Aramco Oil Co. The second author has been also supported by the Bulgarian NSF Grants MM-801.

spaces (see, e.g. [1, 6]). The work in this direction resulted also in construction of efficient iterative methods for solving mixed FE systems (see, e.g., [7–9]).

Galerkin method based on non-conforming Crouzeix-Raviart linear triangular finite elements has been also used in the construction of so called *locking-free* approximations for parameter-dependent problems. Furthermore, the stiffness matrix has a regular sparsity structure such that in each row the number of non-zero entries is at most five.

Therefore, the development of efficient and parallelizable solution methods for non-conforming finite element (FEM) systems is an important problem with a range of applications in scientific computations and engineering. In this paper we construct and study a preconditioner for the algebraic system obtained from discretization of (1) by non-conforming finite element. Our preconditioner is based on MIC(0) factorization of the modified finite element stiffness matrix so that the condition number of the preconditioned system is independent of the possible jumps in the coefficients of the differential equation. Our analysis is done for problems in 2-D domains under the condition that the jumps are aligned with the finite element partition. The study uses the main ideas of the recently proposed highly parallelizable and efficient preconditioners based on MIC(0) for linear conforming and rotated bilinear non-conforming finite elements (see, e.g. [5, 10]).

The rest of the paper is organized as follows. In Sections 2 and 3 we introduce the finite element approximation and the two-level algorithm. In Section 4 we propose a locally modified sparse approximation of the Schur complement and prove that the preconditioned Schur system has a condition number that is bounded uniformly with respect to both the problem size and the possible jumps of the coefficients. The algorithm has been analyzed in the case of coefficient and mesh isotropy. Further, in Section 5 we derive estimates for the execution time on a multiprocessor computer system which shows a good parallel scalability for large scale problems. Finally, in Section 6 we present numerical results on a test problem that shows that the proposed parallel preconditioner preserves the robustness and the computational efficiency of the standard MIC(0) factorization algorithm.

2 Finite element discretization

The domain Ω is partitioned using triangular elements. The partition is denoted by $\mathcal{T}_h$ and is assumed to be quasi-uniform with a characteristic mesh-size h. Most of our analysis is valid for general tensors $a(x)$, but here we restrict our considerations to the $a(x)$ being a scalar function. The partition $\mathcal{T}_h$ is aligned with the discontinuities of the coefficient $a(x)$ so that over each element $e \in \mathcal{T}_h$ the function $a(x)$ is smooth. Further, we assume that $\mathcal{T}_h$ is generated by first partitioning Ω into quadrilaterals Q and then splitting each quadrilateral into two triangles by one of its diagonals, see Figure 1. To simplify our considerations we assume that the splitting into quadrilaterals is topologically equivalent to a square mesh.

The weak formulation of the above problem reads as follows: given $f \in L^2(\Omega)$ find $u \in H^1_D(\Omega) = \{v \in H^1(\Omega) : v = 0 \text{ on } \Gamma_D\}$, satisfying

$$\mathcal{A}(u, v) = (f, v) \quad \forall v \in H^1_D(\Omega), \quad \text{where} \quad \mathcal{A}(u, v) = \int_\Omega a(x)\nabla u(x) \cdot \nabla v(x) dx. \tag{2}$$

We shall discretize this problem by using Crouzeix-Raviart non-conforming linear triangular finite elements. The finite element space V_h consists of piece wise linear functions over $\mathcal{T}_h$ determined by their values in the midpoints of the edges of the triangles. The nodal basis functions of V_h have a support of no more than two neighboring triangles where the corresponding node is the midpoint of their common side. Then the finite element formulation is: find $u_h \in V_h$, satisfying

$$\mathcal{A}_h(u_h, v_h) = (f, v_h) \quad \forall v_h \in V_h, \text{ where} \quad \mathcal{A}_h(u_h, v_h) = \sum_{e \in \mathcal{T}_h} \int_e a(e)\nabla u_h \cdot \nabla v_h dx. \tag{3}$$

Here $a(e)$ is defined as the integral averaged value of $a(x)$ over each $e \in \mathcal{T}_h$. We note that we allow strong coefficient jumps across the boundaries between the adjacent finite elements. Now, the

standard computational procedure leads to the linear system of equations

$$A\mathbf{u} = \mathbf{f}, \tag{4}$$

where A is the corresponding global stiffness matrix and $\mathbf{u} \in \mathbb{R}^N$ is the vector of the unknown nodal values of u_h. The matrix A is sparse, symmetric and positive definite. For large scale problems, the preconditioned conjugate gradient (PCG) method is known to be the best solution method. The goal of this study is to present a **robust and parallelizable** preconditioner for the system (4).

3 The two-level algorithm

Since the triangulation $\mathcal{T}_h$ is obtained by diagonal-wise subdividing each cell Q into two triangles, see Figure 1 (a), we can partition the grid nodes into two groups. The first group contains the centers of the quadrilateral super-elements $Q \subset \Omega$ (the midpoints of the diagonals that split Q into two triangles) and the second group contains the rest of the nodes. With respect to this splitting, A admits the following two-by-two block partitioning that can be written also in a block-factored form

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} A_{11} & 0 \\ A_{21} & S \end{bmatrix} \begin{bmatrix} I & A_{11}^{-1}A_{12} \\ 0 & I \end{bmatrix}, \tag{5}$$

where S stands for the related Schur complement. Obviously, A_{11} is a diagonal matrix so that the Schur complement S can be assembled from the corresponding super-element Schur complements $S_Q = A_{22;Q} - A_{21;Q}A_{11;Q}^{-1}A_{12;Q}$, i.e.

$$S = \sum_{Q \in \mathcal{T}_h} L_Q^T S_Q L_Q,$$

where L_Q stands for the restriction mapping of the global vector of unknowns to the local one corresponding to a macroelement Q containing two triangles. Such a procedure is called *static condensation*. We now introduce S_Q, the local stiffness matrix for Q (in fact, this is the local Schur complement matrix), and its approximation B_Q:

$$S_Q = \begin{bmatrix} s_{11} & s_{12} & s_{13} & s_{14} \\ s_{21} & s_{22} & s_{23} & s_{24} \\ s_{31} & s_{32} & s_{33} & s_{34} \\ s_{41} & s_{42} & s_{43} & s_{44} \end{bmatrix}, \quad B_Q = \begin{bmatrix} b_{11} & s_{12} & 0 & s_{14} \\ s_{21} & b_{22} & s_{23} & 0 \\ 0 & s_{32} & b_{33} & s_{34} \\ s_{41} & 0 & s_{43} & b_{44} \end{bmatrix}. \tag{6}$$

Here $b_{11} = s_{11} + s_{13}$, $b_{22} = s_{22} + s_{24}$, $b_{33} = s_{33} + s_{31}$, $b_{44} = s_{44} + s_{42}$, which ensures that S_Q and B_Q have equal rowsums. The definition of B_Q corresponds to the node numbering as shown in Figure 1. Here the dash lines represent the connectivity pattern of (b) the local Schur complement S_Q and (c) its locally modified sparse approximation B_Q. Assembling the local matrices B_Q we get

$$B = \sum_{Q \in \mathcal{T}_h} L_Q^T B_Q L_Q. \tag{7}$$

The structure of B could be interpreted as a skewed five point stencil. In a very general setting S and B are spectrally equivalent.

After the *static condensation* step, the initial problem (4) is reduced to the solution of a system with the Schur complement matrix S. At this point we apply the PCG method with a preconditioner $\mathcal{C}$ defined as a MIC(0) factorization (see, e.g., [4]) of B, that is, $\mathcal{C} = \mathcal{C}_{MIC(0)}(B)$. This needs of course B to allow for a stable MIC(0) factorization, which will be shown in Section 4.

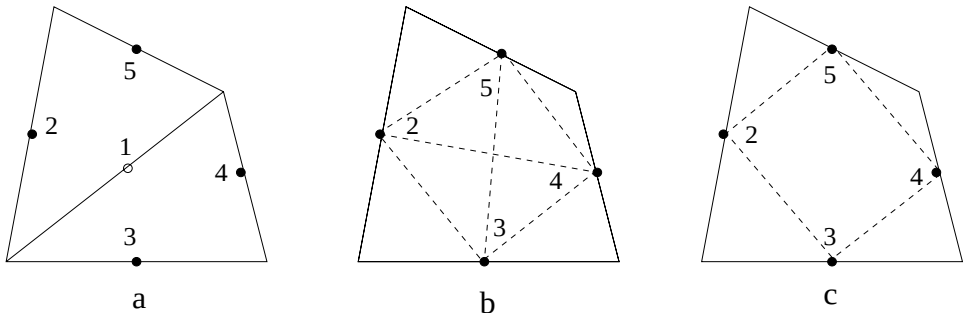

Fig. 1. (a) Node numbering of the super-element Q; (b) Connectivity pattern of S_Q; (c) Connectivity pattern of B_Q.

4 Condition number analysis for a square mesh

The model problem we analyze in this section is set on a uniform square mesh. Then the element stiffness matrix corresponding to the triangle element $e \in \mathcal{T}_h$ has the form

$$A_e = 2a_e \begin{bmatrix} 2 & -1 & -1 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}. \tag{8}$$

Let us assume now that the square super-element Q contains of the triangles e_1 and e_2 where the element-wise averaged diffusion coefficients are respectively a_1 and a_2. Then the matrices needed for our local analysis are as follows:

$$S_Q = \frac{1}{a_1 + a_2} \begin{bmatrix} a_1^2 + 2a_1a_2 & -a_1^2 & -a_1a_2 & -a_1a_2 \\ -a_1^2 & a_1^2 + 2a_1a_2 & -a_1a_2 & -a_1a_2 \\ -a_1a_2 & -a_1a_2 & a_2^2 + 2a_1a_2 & -a_2^2 \\ -a_1a_2 & -a_1a_2 & -a_2^2 & a_2^2 + 2a_1a_2 \end{bmatrix}, \tag{9}$$

$$B_Q = \frac{1}{a_1 + a_2} \begin{bmatrix} a_1^2 + a_1a_2 & -a_1^2 & 0 & -a_1a_2 \\ -a_1^2 & a_1^2 + a_1a_2 & -a_1a_2 & 0 \\ 0 & -a_1a_2 & a_2^2 + a_1a_2 & -a_2^2 \\ -a_1a_2 & 0 & -a_2^2 & a_2^2 + a_1a_2 \end{bmatrix}. \tag{10}$$

We consider now the local eigenvalue problem: find $\lambda \in R$ and $\mathbf{0} \neq \mathbf{w} \in \mathbb{R}^4$ such that

$$S_Q\mathbf{w} = \lambda B_Q\mathbf{w}. \tag{11}$$

Obviously $Ker(S_Q) = Ker(B_Q) = Span\{\mathbf{e}\}$ where $\mathbf{e} = (1, 1, 1, 1)^t$. Thus, (11) reduces to a 3×3 eigenvalue problem. Then using the substitution $\nu = a/b$, $\mu = 1-\lambda$ we get the following characteristic equation for μ

$$det \begin{bmatrix} \nu + (\nu^2 + \nu)\mu & -\nu^2\mu & -\nu \\ -\nu^2\mu & \nu + (\nu^2 + \nu)\mu & -\nu\mu \\ -\nu & -\nu\mu & \nu + (1 + \nu)\mu \end{bmatrix} = 0. \tag{12}$$

A further simple computation shows that $\mu_1 = 0$ and $\mu_2 = \mu_3 = -1$, and therefore $\lambda_1 = 1$, $\lambda_2 = \lambda_3 = 2$. The global condition number estimate follows directly from the presented local analysis. Namely, we have

$$\mathbf{v}^T S\mathbf{v} = \sum_{Q \in \mathcal{T}_h} \mathbf{v}^T L_Q^T S_Q L_Q \mathbf{v} \leq 2 \sum_{Q \in \mathcal{T}_h} \mathbf{v}^T L_Q^T B_Q L_Q \mathbf{v} = 2\mathbf{v}^T B\mathbf{v}$$

and, similarly, $\mathbf{v}^T S\mathbf{v} \leq \mathbf{v}^T B\mathbf{v}$. The result of our local analysis is summarized in the following theorem:

Theorem 1. *Consider the non-conforming FEM problem (3) on a square mesh. Then:*

(i) the sparse approximation B of the Schur complement S satisfies the conditions for a stable MIC(0) factorization;

(ii) the matrices B and S are spectrally equivalent, namely the following estimate for the relative condition number holds uniformly with respect to any possible jumps of the diffusion coefficients

$$\kappa\left(B^{-1}S\right) \leq 2. \tag{13}$$

5 Parallel preconditioning algorithm

In this section we study the possibility to parallelize the proposed method. The first step, i.e. the static condensation, is local and therefore can be performed fully in parallel. This is the reason to focus our attention on the PCG solution of the reduced system with the Schur complement S. Let us recall that the preconditioner was introduced as $\mathcal{C} = \mathcal{C}_{MIC(0)}(B)$. Each PCG iteration consists of one solution of a system with the matrix $\mathcal{C}$, one matrix vector multiplication with the original matrix S, two inner products, and three linked vector triads of the form $\mathbf{v} := \alpha\mathbf{v} + \mathbf{u}$. Therefore the computational complexity of one PCG iteration is given by $\mathcal{N}^{it}_{PCG} \approx \mathcal{N}(\mathcal{C}^{-1}\mathbf{v}) + \mathcal{N}(S\mathbf{v}) + 10N$. Then, for the algorithm introduced in Section 3 we find $\mathcal{N}(\mathcal{C}^{-1}\mathbf{v}) \approx 11N$, $\mathcal{N}(S\mathbf{v}) \approx 13N$, and finally

$$\mathcal{N}^{it}_{PCG} \approx 34N. \tag{14}$$

As we see, the algorithm under consideration is relatively cheap where the solution of the preconditioned system takes less then one third of the total cost.

It is well known that MIC(0) is an inherently sequential algorithm. In the general case, the solution of the arising triangular systems is typically recursive. Below we overcome this disadvantage by a special construction of the matrix B. For simplicity of the presentation we consider the model problem in a square $\Omega = (0,1)^2$ on a square mesh with a mesh size $h = 1/n$ (subsequently each square is split into two triangles to get $\mathcal{T}_h$). The structure of the matrices S and B is illustrated on Figure 2 where each of the diagonal blocks corresponds to one vertical line of the mesh if a column-wise numbering of the unknowns has been used. The big advantage of the introduced matrix B is that all of its diagonal blocks are diagonal. In this case, the implementation of the PCG solution step $\mathcal{C}^{-1}\mathbf{v}$ is fully parallel within each of these blocks.

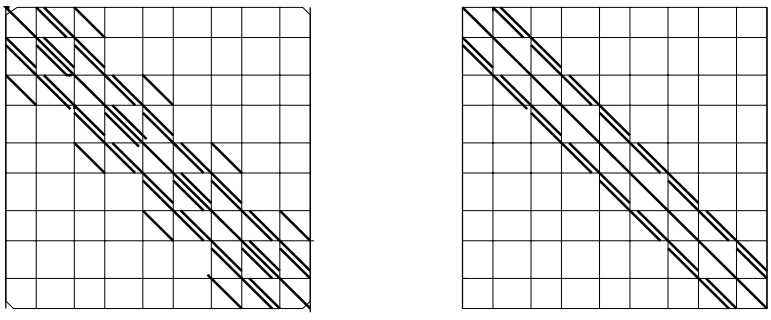

Fig. 2. Sparsity pattern of the matrices S and B, $\Omega = (0,1)^2$.

To establish the theoretical performance characteristics of the preconditioner, a simple general model for the arithmetic and the communication times is applied (see, e.g., [11]). We assume that the computations and communications do not overlap, and therefore, the parallel execution time is the sum of the computation and communication times. We also assume that the execution of

M arithmetic operations on one processor takes time $T_a = Mt_a$, where t_a is the average unit time to perform one arithmetic operation on one processor (no vectorization). We assume that the communication time to transfer M data elements from one processor to another can be approximated by $T_{com} = \ell(t_s + Mt_c)$, where t_s is the start-up time and t_c is the incremental time necessary for each of M elements to be sent, and ℓ is the graph distance between the processors.

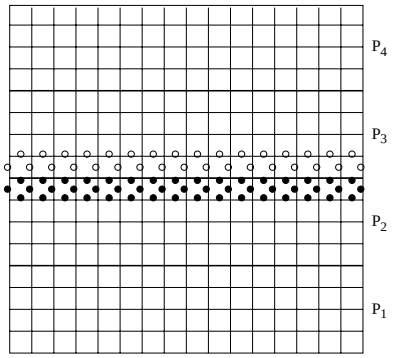

Fig. 3. Strip-wise data distribution between the processors.

Further, we consider a distributed memory parallel algorithm where the number of processors is p, and $n = mp$ for some natural number m. The computational domain is split in p equally sized strips where the processor P_k is responsible for computations related to the k-th strip. Then, we get the following expressions for the communication times related to $\mathcal{C}^{-1}\mathbf{v}$ and $S\mathbf{v}$

$$T_{com}(C^{-1}\mathbf{v}) = 6n(t_s + t_c), \qquad T_{com}(S\mathbf{v}) = 2t_s + (3n+1)t_c.$$

Note that the above communications are completely local and do not depend on the number of processors p assuming that P_k and P_{k+1} are neighbors. The linked triads are free of communications. The inner product can be performed using one broadcasting and one gathering global communication but they do not contribute to the leading terms of the total parallel time and will not be considered in our analysis. This setting leads to the following expression for the parallel time per one PCG iteration

$$T_p = T_p^{it} \approx \frac{2n(n+1)}{p} t_a + 6nt_s + 9nt_c. \tag{15}$$

¿From (15) we conclude that the parallel algorithm is asymptotically optimal and

$$\lim_{n\to\infty} S_p = p, \qquad \lim_{n\to\infty} E_p = 1,$$

where the parallel speed-up and the parallel efficiency are given in the usual form $S_p = T_1/T_p$, and $E_p = S_p/p$.

Remark 1. A more realistic analysis of the parallel performance needs some specific information about the behavior of the introduced average timing parameters t_a, t_s and t_c. The key point here is that a good parallel scalability could be expected only if $n >> pt_s/t_a$.

6 Numerical tests

The numerical tests presented below illustrate the PCG convergence rate of the studied MIC(0) preconditioning algorithms when the size of the discrete problem and the coefficient jumps are varied. A relative stopping criterion $(\mathcal{C}^{-1} r^{n_{it}}, r^{n_{it}})/(\mathcal{C}^{-1} r^0, r^0) < \varepsilon$ is used in the PCG algorithm, where r^i

stands for the residual at the i-th iteration step, $(\cdot,\cdot)$ is the standard Euclidean inner product, and $\varepsilon = 10^{-6}$. The computational domain is the unit square $\Omega = (0,1)^2$ where homogeneous Dirichlet boundary conditions are assumed at the bottom side. A uniform mesh is used, where $h = 1/n$, and the size of the discrete problem is $N = 2n(n+1)$. Let $\Omega = \Omega_1 \bigcup \Omega_2$,

$$\Omega_2 := \{\frac{n-1}{2}h \leq x_1 \leq \frac{n+1}{2}h, \quad x_2 > \frac{n+1}{4}h\},$$

and let a_i be the problem coefficient corresponding to Ω_i, $i = 1, 2$. In what follows $a_1 = 1$. This test problem allows us to examine the influence of the coefficient jumps on the number of iterations. Note that the coefficient jumps are highly localized since the width of the domain Ω_2 is just one mesh-size.

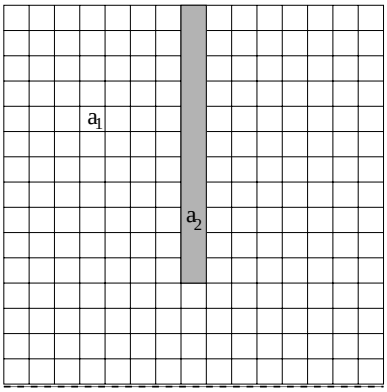

Fig. 4. Test problem: $n = 15$, $\Omega_2 := \{7/15 \leq x_1 \leq 8/15, \quad x_2 > 4/15\}$.

Table 1. *PCG* iterations: $MIC(0)$ factorization of S and B.

$h = 1/n$	problemsize	n_{it}^{SS}		n_{it}^{SB}	
n	N	$L \equiv -\Delta$	$a_2 = 10^3$	$L \equiv -\Delta$	$a_2 = 10^3$
7	112	10	16	11	17
15	480	16	29	17	30
31	1984	23	47	24	52
63	8064	34	73	35	81
127	32512	50	117	49	129

Two model tests are reported in Table 1 where: (a) $a_2 = 1$ or the differential operator L is the Laplacian $-\Delta$ and (b) $a_2 = 10^3$. We investigate also the influence of approximation of the Schur complement matrix S by the introduced sparse approximation B. We denote by n_{it}^{SS} and n_{it}^{SB} the number of iterations obtained when MIC(0) factorization of S and B are used as preconditioners of S.

The qualitative analysis of the results given in Table 1 shows that: (a) the number of iterations in all cases is $O(n^{1/2}) = O(N^{1/4})$, namely, it grows proportionally to $\sqrt{n}$ in agreement with the properties of the MIC(0) factorization of S; (b) the number of iterations n_{it}^{SS} and n_{it}^{SB} are practically the same for both the model problem and for the problem with large jumps in the coefficients. Note that the obtained results are considerably better than what we have as a prediction from the uniform estimate of Theorem 1.

Table 2. *PCG* iterations for $n = 65$, varying a_2, and $MIC(0)$ factorization of B

a_2	1	10	10^2	10^3	10^4
n_{it}^{SB}	35	45	62	81	93

The impact of the coefficient jump on the number of PCG iterations for a fixed 63×63-mesh is presented in Table 2. We see some increase of the iterations with a_2. Nevertheless, the obtained results can be viewed as very promising taking into account that the jump is not only very large, but it is also highly localized within a strip of width h.

7 Concluding remarks

In this paper we have proposed a new two-level preconditioner for Crouzeix-Raviart non-conforming finite element approximation of second order elliptic equations. Our study is motivated by two factors. First, the Crouzeix-Raviart non-conforming finite elements produce algebraic systems that are equivalent to the Schur complement system for the Lagrange multipliers arising from the mixed finite element method for Raviart-Thomas elements (see, e.g. [1, 2, 6]). Second, a class of highly parallelizable and efficient preconditioners based on MIC(0) have been proposed recently for linear conforming and rotated bilinear non-conforming finite elements (see, e.g. [5, 10]).

Our further plans include a generalization to 3-D problems on tetrahedral meshes and problems with orthotropy. These are much more complicated problems but we expect to extend our study to such problems and to be able to construct, test, and implement efficient preconditioners with similar theoretical and computational properties.

References

1. T. Arbogast and Z. Chen, On the implementation of mixed methods as nonconforming methods for second order elliptic problems, *Math. Comp.*, **64** (1995), 943–972.
2. D.N. Arnold and F. Brezzi, Mixed and nonconforming finite element methods: implementation, post-processing and error estimates, *RAIRO, Model. Math. Anal. Numer.*, **19** (1985), 7–32.
3. O. Axelsson, L. Laayyonni, S. Margenov, On multilevel preconditioners which are optimal with respect to both problem and discretization parameters, *SIAM J. Matr. Anal. Appl.* (submitted).
4. R. Blaheta, Displacement decomposition - incomplete factorization preconditioning techniques for linear elasticity problems, *Numer. Lin. Alg. Appl.*, **1** (1994), 107–126.
5. G. Bencheva, S. Margenov, Parallel incomplete factorization preconditioning of rotated linear FEM systems, *Computer & Mathematics with Applications* (2002) (submitted).
6. Z. Chen, Analysis of mixed methods using conforming and nonconforming finite element methods, *RAIRO, Math. Model. Numer. Anal.*, **27** (1993), 9–34.
7. Z. Chen, Equivalence between multigrid algorithms for mixed and nonconformning methods for second order elliptic problems, *East-West J. Numer. Math.*, **4** (1996), 1–33.
8. Z. Chen, R.E. Ewing, Yu.A. Kuznetsov, R.D. Lazarov, and S. Maliassov, Multilevel preconditioners for mixed methods for second order elliptic problems, *Numer. Lin. Alg. with Appl.*, **3(5)** (1996), 427-453.
9. Z. Chen, R.E. Ewing, and R.D. Lazarov, Domain decomposition algorithms for mixed methods for second order elliptic problems, *Math. Comp.*, **65(214)** (1996), 467-490.
10. I. Gustafsson, G. Lindskog, On parallel solution of linear elasticity problems: Part I: Theory, *Numer. Lin. Alg. Appl.*, **5** (1998), 123–139.
11. Y. Saad, M.H. Schultz, Data Communication in Parallel Architectures, *Parallel Comput.*, **11** (1989), 131–150.