Domain Decomposition Preconditioner for Anisotropic Diffusion

Yuri A. Kuznetsov

Department of Mathematics, University of Houston, 651 Philip G. Hoffman Hall, Houston, TX 77204–3008, USA. kuz@math.uh.edu

Summary. We propose and investigate two-level preconditioners for the diffusion equations with anisotropic coefficients in model polyhedral domains. Preconditioners are based on a partitioning of the mesh in (x, y)-plane into non-overlapping subdomains and on a special coarsening algorithm in each of the mesh layers. The condition number of the preconditioned matrix does not depend on the coefficients in the diffusion operator. Numerical experiments confirm the theoretical results.

1 Introduction

In this paper, we propose and investigate a new approach to the construction of two-level preconditioners for the diffusion equation with anisotropic diffusion tensor. We consider the case of special polyhedral domains and special prismatic meshes. The diffusion tensor is assumed to be a diagonal matrix, and the simplest version of the finite volume method is used for the discretization of the diffusion equation. The choice of the domain, the meshes, and the discretization method are motivated by applications in reservoir simulation. We can also use for the discretization the polyhedral "div–const" mixed finite element method, see [4].

The paper is organized as follows. In Section 2, we describe the model problem and the matrices which arise in the simplest version of the finite volume method.

In Section 3, we propose a special coarsening procedure based on the partitioning of the mesh domain in (x, y)-plane into non-overlapping subdomains. This procedure is a special modification of the algorithm earlier proposed in [6] and utilized in [2]. We prove that the condition number of the preconditioned matrix is independent of the values of the coefficients in the diffusion equation, i.e. it does not depend on an anisotropy in the diffusion tensor.

An implementation algorithm in the form of a two-step iterative method is considered in Section 4. It is based on the idea of the matrix iterative methods in subspaces, see [3, 5]. The algorithm naturally leads to a coarse mesh system.

In Section 5, we design another two-level preconditioner which is much cheaper with respect to the arithmetic implementation cost than the previous one. Another advantage of the second preconditioner is that it allows a multilevel extension. Numerical results in Section 6 demonstrate the efficiency of the second preconditioner. They confirm the theoretical results in Sections 3 and 5.

2 Formulation of Model Problem

We consider the Neumann boundary value problem for the diffusion equation

$$\begin{array}{rcl} -\nabla\cdot\big(a\nabla p\big)+c\,p=f & \text{in} & \Omega \\ \big(a\nabla p\big)\cdot\boldsymbol{n}=0 & \text{on} & \partial\Omega \end{array} \tag{1}$$

where $\Omega=\Omega_{xy}\times(0;1)$ is a prismatic domain in $\mathbb{R}^3$, $\boldsymbol{n}$ is the outward unit normal to $\partial\Omega$, and Ω_{xy} is a polygon in the (x,y)-coordinate plane. An example of Ω is given in Figure 1. The diffusion tensor a is a diagonal 3×3 matrix with coinciding diagonal entries in (x,y)-plane, i.e. $a=\text{diag}\,\{a_{xy},\ a_{xy},\ a_z\}$, and c is a non-negative function. The domain Ω is partitioned into subdomains $\Omega_l=\Omega_{xy,l}\times(Z_B^{(l)};Z_T^{(l)})$ where $\Omega_{xy,l}$ are convex polygons, $0\le Z_B^{(l)}<Z_T^{(l)}\le1$, $l=\overline{1,\ m}$, and m is a positive integer. We assume that a_{xy} and a_z are positive constants and c is a non-negative constant in each of the subdomains $\Omega_{xy,l}$, $l=\overline{1,\ m}$. We also assume that the coefficient c is positive in at least one subdomain Ω_l, $1\le l\le m$.

Let $\Omega_{xy,h}$ be a conforming polygonal mesh in Ω_{xy}, and Z_h be a partitioning of $[0;1]$ into segments $[z_{k-1};z_k]$, $k=\overline{1,\ n_z}$, where n_z is a positive integer. Then, $\Omega_h=\Omega_{xy,h}\times Z_h$ is a prismatic mesh in Ω. We assume that the mesh Ω_h is conforming with respect to the boundaries $\partial\Omega_l$ of subdomains Ω_l, $l=\overline{1,\ m}$. We also assume that the interfaces between neighboring cells in $\Omega_{xy,h}$ are always straight segments.

To discretize diffusion equation we utilize the simplest version of the finite volume method. In the case of uniform rectangular or hexagonal meshes $\Omega_{xy,h}$ this discretization is sufficiently accurate. We may also assume that $\Omega_{xy,h}$ is a Voronoi mesh. The simplest finite volume method results in the system of linear algebraic equations

$$K\,\bar{p}\ =\ \bar{f} \tag{2}$$

with a symmetric positive definite $\tilde{n}\times\tilde{n}$ matrix K where $\tilde{n}=\tilde{n}_{xy}\times n_z$ and $\tilde{n}_{xy}$ is the total number of polygonal cells in $\Omega_{xy,h}$. System (2) can be easily hybridized algebraically by introducing additional degrees-of-freedom (DOF) λ on the interfaces between all or selected neighboring mesh cells in $\Omega_{xy,h}$ as well as on the edges of cells in Ω_{xy} belonging to the boundary $\partial\Omega_{xy}$ of Ω_{xy}. In terms of old variables p and new variables λ the underlying system of linear algebraic equations can be written as follows:

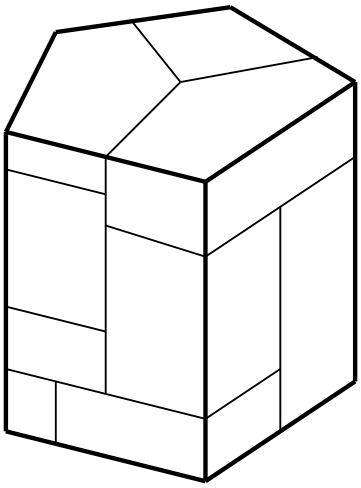

Fig. 1. An example of Ω partitioned into subdomains Ω_l, $l = \overline{1,\ m}$

$$A \begin{bmatrix} \bar{p} \\ \bar{\lambda} \end{bmatrix} \equiv \begin{pmatrix} A_p & A_{p\lambda} \\ A_{\lambda p} & A_\lambda \end{pmatrix} \begin{bmatrix} \bar{p} \\ \bar{\lambda} \end{bmatrix} = \overline{F}. \tag{3}$$

The matrix

$$K = A_p - A_{p\lambda} A_\lambda^{-1} A_{\lambda p} \tag{4}$$

in (2) is the Schur complement of A, and

$$\overline{F} = \begin{bmatrix} \bar{f} \\ 0 \end{bmatrix}. \tag{5}$$

The definition of λ is based on the observation that the three-point finite difference equation

$$\left(\frac{2a_1}{h_1} + \frac{2a_2}{h_2}\right) w + \left(p_2 - p_1\right) = 0 \tag{6}$$

for the flux equation $w + \partial p/\partial \xi = 0$ on the interface between two cells is equivalent to three equations

$$\begin{aligned} &\frac{2a_1}{h_1} w_1 - p_1 - \lambda = 0, \\ &\frac{2a_2}{h_2} w_2 + p_2 + \lambda = 0, \\ &w_1 - w_2 = 0 \end{aligned} \tag{7}$$

with $w = w_1 = w_2$. Then, we use the standard condensation procedure to derive system (3).

Let us present the matrix K as the sum of two matrices:

$$K \;=\; K_{xy} \;+\; K_z \tag{8}$$

where K_{xy} corresponds to the discretization of the operator

$$\mathcal{L}_{xy} \;=\; -\,\frac{\partial}{\partial x}\Big(a_{xy}\frac{\partial}{\partial x}\Big) \;-\; \frac{\partial}{\partial y}\Big(a_{xy}\frac{\partial}{\partial y}\Big) \tag{9}$$

and K_z corresponds to the discretization of the operator

$$\mathcal{L}_z \;=\; -\,\frac{\partial}{\partial z}\Big(a_z\frac{\partial}{\partial z}\Big) \;+\; c. \tag{10}$$

Then,

$$A \;=\; A_{xy} \;+\; A_z \tag{11}$$

where

$$A_{xy} \;=\; \begin{pmatrix} A_{xy,p} & A_{p\lambda} \\ A_{\lambda p} & A_{\lambda} \end{pmatrix}, \qquad A_z \;=\; \begin{pmatrix} K_z & 0 \\ 0 & 0 \end{pmatrix}, \tag{12}$$

and

$$K_{xy} \;=\; A_{xy,p} \;-\; A_{p\lambda}\,A_{\lambda}^{-1}\,A_{\lambda p} \tag{13}$$

is the Schur complement of the matrix A_{xy}.

We observe that with an appropriate permutation matrix P the matrix $PA_{xy}P^T$ is a block diagonal matrix, i.e.

$$P\,A_{xy}\,P^T \;=\; A_{xy}^{(1)} \;\oplus\; \cdots \;\oplus\; A_{xy}^{(n_z)} \tag{14}$$

where the submatrices $A_{xy}^{(k)}$ correspond to the above hybridized discretization of the operator $\mathcal{L}_{xy}$ in (9) in the mesh layers $\Omega_{xy,h} \times [z_{k-1}; z_k]$, $k = \overline{1,\; n_z}$.

In the next section, we shall derive a preconditioner H for the matrix A in (3). Let us assume that A and B are symmetric and positive definite matrices and the inequalities

$$\alpha\,\big(B\,\bar{v},\;\bar{v}\big) \;\le\; \big(A\,\bar{v},\;\bar{v}\big) \;\le\; \beta\,\big(B\,\bar{v},\;\bar{v}\big) \tag{15}$$

hold for all $\bar{v} \in \mathbb{R}^n$ with some positive coefficients α and β, where n is the size of A. We present the matrix $H = B^{-1}$ in the block form similar to (3):

$$H \;=\; \begin{pmatrix} H_p & H_{p\lambda} \\ H_{\lambda p} & H_{\lambda} \end{pmatrix}. \tag{16}$$

Then, the inequalities

$$\frac{1}{\beta}\,\big(H_p\,\bar{q},\;\bar{q}\big) \;\le\; \big(K^{-1}\,\bar{q},\;\bar{q}\big) \;\le\; \frac{1}{\alpha}\,\big(H_p\,\bar{q},\;\bar{q}\big) \tag{17}$$

hold for all $\bar{q} \in \mathbb{R}^{\tilde{n}}$. Thus, with respect to estimates (15) the matrix H_p is not a worse preconditioner for the matrix K than the preconditioner $H = B^{-1}$ for the matrix A.

3 Two-Level Preconditioner

Let $\Omega_{xy,h}$ be partitioned into non-overlapping mesh subdomains $G_{h,s}$, $s = \overline{1,\ t}$, where t is a positive integer. We assume that this partitioning is conforming with respect to the boundaries of subdomains $\Omega_{xy,l}$, i.e. the boundaries of $\Omega_{xy,l}$ are subsets of the union of the boundaries of subdomains $G_{h,s}$, $s = \overline{1,\ t}$, $l = \overline{1,\ m}$. It follows from the above assumptions that the coefficients a_{xy} and a_z are positive constants $a_{xy}^{k,s}$ and $a_z^{k,s}$, respectively, and the coefficient c is a non-negative constant $c_{k,s}$ in each of the mesh subdomains $G_{h,s} \times [z_{k-1}; z_k]$, $s = \overline{1,\ t}$, $k = \overline{1,\ n_z}$.

We assume that the additional DOF λ are imposed only on the interface boundaries between mesh subdomains $G_{h,s}$ and $G_{h,s'}$, $s' \neq s$, $s, s' = \overline{1,\ t}$, and on the boundary $\partial\Omega_{xy}$. Then, assembling matrices N_s, $s = \overline{1,\ t}$, exist such that

$$A_{xy}^{(k)} \ = \ h_{z,k} \sum_{s=1}^{t} a_{xy}^{k,s}\, N_s\, A_{xy,s}\, N_s^T \tag{18}$$

where $h_{z,k} = z_k - z_{k-1}$, $k = \overline{1,\ n_z}$, and $A_{xy,s}$ represents the hybridized discretization of the operator $\mathcal{L}_{xy}$ in mesh subdomain $G_{h,s}$, $s = \overline{1,\ t}$. The matrices $A_{xy,s}$ are symmetric and positive semi-definite, and $\ker A_{xy,s}$ (nullspace of $A_{xy,s}$) is the span of $\bar{e}_s \in \mathbb{R}^{n_s}$ where $\bar{e}_s = \big(\ 1,\ \ldots\ ,\ 1\big)^T$ and n_s is the size of $A_{xy,s}$, $s = \overline{1,\ t}$.

Let D_s be a diagonal $n_s \times n_s$ matrix with positive entries on the diagonal, $1 \leq s \leq t$. Consider the eigenvalue problem

$$A_{xy,s}\, \bar{w} \ = \ \mu\, D_s\, \bar{w}, \quad \bar{w} \in \mathbb{R}^{n_s}. \tag{19}$$

Then the spectral decomposition of $A_{xy,s}$ is defined as follows:

$$A_{xy,s} \ = \ D_s\, W_s\, \Lambda_s\, W_s^T\, D_s \tag{20}$$

where

$$\Lambda_s \ = \ \mathrm{diag}\Big\{\mu_1^{(s)},\ \mu_2^{(s)},\ \cdots\ ,\ \mu_{n_s}^{(s)}\Big\} \tag{21}$$

is a diagonal matrix, and

$$W_s \ = \ \Big(\bar{w}_{s,1},\ \bar{w}_{s,2},\ \ldots\ ,\ \bar{w}_{s,n_s}\Big). \tag{22}$$

Here, $0 = \mu_1^{(s)} < \mu_2^{(s)} \leq \cdots \leq \mu_{n_s}^{(s)}$ are the eigenvalues, and $\bar{w}_{s,1},\ \bar{w}_{s,2},\ \ldots,$ $\bar{w}_{s,n_s}$ are the corresponding D_s-orthonormal eigenvectors. It is obvious that $\bar{w}_{s,1} = \sigma_s^{-1}\bar{e}_s$ with $\sigma_s = \big(D_s\bar{e}_s, \bar{e}_s\big)^{1/2}$.

Let us define the matrices

$$B_{xy,s} \ = \ \hat{\mu}_s\Big[\, D_s \ - \ D_s\, \bar{w}_{s,1}\, \bar{w}_{s,1}^T\, D_s\,\Big] \tag{23}$$

where $\hat{\mu}_s$ are arbitrary positive numbers, $s = \overline{1,\ t}$. It can be easily shown that the inequalities

$$\mu_2^{(s)} \left(B_{xy,s}\, \bar{v},\ \bar{v}\right) \ \leq\ \hat{\mu}_s \left(A_{xy,s}\, \bar{v},\ \bar{v}\right) \ \leq\ \mu_{n_s}^{(s)} \left(B_{xy,s}\, \bar{v},\ \bar{v}\right) \tag{24}$$

hold for all $\bar{v} \in \mathbb{R}^{n_s}$, $s = \overline{1,\ t}$.

We define the matrices

$$B_{xy}^{(k)} \ =\ h_{z,k} \sum_{s=1}^{t} a_{xy}^{k,s}\, N_s\, B_{xy,s}\, N_s^T, \tag{25}$$

$k = \overline{1,\ n_z}$, the matrix

$$B_{xy} \ =\ P^T \left(B_{xy}^{(1)} \ \oplus\ \cdots\ \oplus\ B_{xy}^{(n_z)}\right) P, \tag{26}$$

and, finally, the matrix

$$B \ =\ B_{xy} \ +\ A_z. \tag{27}$$

The matrix B in (27) may be considered as the first candidate to precondition the matrix A in (3).

It can be easily proved that for the matrix B in (27) inequalities (15) hold with

$$\alpha \ =\ \min\left\{1;\ \min_{1\leq s\leq t} \frac{\mu_2^{(s)}}{\hat{\mu}_s}\right\}, \quad \beta \ =\ \max\left\{1;\ \max_{1\leq s\leq t} \frac{\mu_{n_s}^{(s)}}{\hat{\mu}_s}\right\} \tag{28}$$

where $\mu_2^{(s)}$ and $\mu_{n_s}^{(s)}$ are the minimal non-zero and the maximal eigenvalues in (19), respectively.

Let $\mathrm{cond}_A(B^{-1}A)$ be the condition number of the matrix $B^{-1}A$ with respect to the norm generated by the matrix A. Then, the estimate

$$\mathrm{cond}_A\left(B^{-1}\, A\right) \ \leq\ \nu \tag{29}$$

holds with $\nu = \beta/\alpha$ where the values of α and β are given in (28). We observe that the value of ν does not depend on the values of the coefficients a_{xy}, a_z, and c in diffusion equation (1) as well as on the mesh Z_h.

To define a proper diagonal matrix D_s in (19) we have to analyze the matrix $A_{xy,s}$ and the restriction of the mesh $\Omega_{xy,h}$ onto the subdomain $G_{h,s}$, $1 \leq s \leq t$. The matrices $A_{xy,s}$ and D_s can be presented in the 2×2 block form by

$$A_{xy,s} \ =\ \begin{pmatrix} A_p & A_{p\lambda} \\ A_{\lambda p} & A_\lambda \end{pmatrix}, \qquad D_s \ =\ \begin{pmatrix} D_p & 0 \\ 0 & D_\lambda \end{pmatrix} \tag{30}$$

where the index "s" in the blocks is omitted. Here, diagonal blocks A_p and D_p are associated with the cell-centered DOFs, and diagonal blocks A_λ and D_λ are associated with the interface DOFs. Let E be a polygonal cell in $G_{h,s}$.

Then, the diagonal entry of the matrix D_p in (30), associated with E, is equal to the area of E. The boundary of $G_{h,s}$ is the union of edges of polygonal cells E in $G_{h,s}$. We assigned with each of such edges one DOF in $\bar{\lambda}$, and with each DOF in $\bar{\lambda}$ we associate the length of the underlying edge in $G_{h,s}$. Moreover, the boundary of $G_{h,s}$ consists of the interfaces $\Gamma_{s,j}$, $j = \overline{1,\ l_s}$, between $G_{h,s}$ and neighboring subdomains $G_{h,s'}$, $s' \neq s$, as well as of the interfaces between $G_{h,s}$ and $\partial\Omega_{xy}$ where l_s is a positive integer. We assume that each of the interfaces is a simply connected subset of the boundary of $G_{h,s}$. We assign for each of the interfaces $\Gamma_{s,j}$ a positive number $d_{s,j}$, $j = \overline{1,\ l_s}$. We assume that for the interfaces $\Gamma_{s,j} = \Gamma_{s',j'}$ between neighboring subdomains $G_{h,s}$ and $G_{h,s'}$, $s' \neq s$, the values $d_{s,j}$ and $d_{s',j'}$ are equal to each other. Now, we define the diagonal entries of the matrix D_λ in (30). Let λ be a DOF in $\bar{\lambda}$ assigned for a segment γ belonging to interface $\Gamma_{s,j}$, $1 \leq j \leq l_s$. Then, the associated with λ the diagonal entry of D_λ is the product of the length of γ and $d_{s,j}$.

To derive estimates for α and β in (28), we assume that the mesh $\Omega_{xy,h}$ and the partitioning of $\Omega_{xy,h}$ into subdomains $G_{h,s}$, $s = \overline{1,\ t}$, are quasi-uniform and regular shaped. On the basis of the latter assumptions we introduce two parameters:

$$h_f \ = \ \tilde{n}_{xy}^{-1/2} \quad \text{and} \quad h_c \ = \ t^{-1/2}. \tag{31}$$

It is clear that h_f and h_c can be called as the fine mesh step size and the coarse mesh step size, respectively. We assume that $d_{s,j} = h_f$ in the definition of the diagonal entries of the submatrices D_λ in (30), $j = \overline{1,\ l_s}$, $s = \overline{1,\ t}$.

It can be proved that under the above assumptions the estimates

$$\begin{aligned} \min_{1\leq s\leq t} \mu_2^{(s)} &\geq c_1\, h_c^{-2} \\ \max_{1\leq s\leq t} \mu_{n_s}^{(s)} &\leq c_2\, h_f^{-2} \end{aligned} \tag{32}$$

hold, where c_1 and c_2 are positive constants independent of the mesh $\Omega_{xy,h}$ and subdomains $G_{h,s}$, $s = \overline{1,\ t}$.

Let us choose $\hat{\mu}_s = h_c^{-2}$, $s = \overline{1,\ t}$. Then, combining (28), (29), and (32) we get the following result.

Proposition 1. *Under assumptions made the estimate*

$$\mathrm{cond}_A\big(B^{-1}\,A\big) \ \leq \ c_3\Big(\frac{h_c}{h_f}\Big)^2 \tag{33}$$

holds where c_3 is a positive constant independent of the coefficients a_{xy}, a_z, and c in (1), mesh Ω_h, and the subdomains $G_{h,s}$, $s = \overline{1,\ t}$.

Thus, the proposed preconditioner is robust with respect to the diffusion tensor but it is not optimal with respect to the mesh in the case $h_c \gg h_f$.

4 Implementation Algorithm

In this Section, we derive a solution algorithm for an algebraic system

$$B\,\bar{v} \;=\; \bar{g} \tag{34}$$

with the matrix B defined in (27) and a right hand side vector $\bar{g} \in \mathbb{R}^n$.

The solution algorithm is based on the splitting

$$B \;=\; B_0 \;-\; C_0 \tag{35}$$

of the matrix B into the matrices

$$B_0 \;=\; A_z \;+\; D \tag{36}$$

and

$$C_0 \;=\; P^T \left[C_0^{(1)} \;\oplus\; \ldots \;\oplus C_0^{(n_z)} \right] P. \tag{37}$$

Here,

$$D \;=\; P^T \left[\widetilde{D}_1 \;\oplus\; \ldots \;\oplus \widetilde{D}_{n_z} \right] P, \tag{38}$$

is a diagonal matrix with diagonal submatrices

$$\widetilde{D}_k \;=\; h_{z,k} \sum_{s=1}^{t} \hat{\mu}_s \, a_{xy}^{k,s} \, N_s \, D_s \, N_s^T, \tag{39}$$

$k = \overline{1,\; n_z}$, and

$$C_0 \;=\; D \;-\; B_{xy}, \tag{40}$$

where B_{xy} is defined in (25), (26).

The implementation algorithm consists of two steps. At the first step, we compute the solution vector of the system

$$B_0\,\bar{v}_1 \;=\; \bar{g}. \tag{41}$$

With a proper permutation matrix P_z the matrix $P_z B_0 P_z^T$ is a block diagonal matrix. Each diagonal block of this matrix is either a tridiagonal matrix (for p-variables) or a diagonal matrix (for λ-variables). The total number of blocks is equal to n_{xy}.

At the second step, we are looking for the vector

$$\bar{v}_2 \;=\; \bar{v}_1 \;+\; \bar{\eta} \tag{42}$$

where $\bar{\eta}$ is the solution vector of the system

$$B\,\bar{\eta} \;=\; -\left(B\,\bar{v}_1 \;-\; \bar{g}\right), \tag{43}$$

or of the equivalent system

$$B\,\bar{\eta} \;=\; \bar{\xi} \tag{44}$$

with the right hand side vector

$$\bar{\xi} \;=\; C_0\,B_0^{-1}\,\bar{g}. \tag{45}$$

It is obvious that the vector $\bar{v}_2$ in (42) is the solution of system (34).

The vector $\bar{\xi}$ in (45) belongs to the image of the matrix C_0. We observe that the rank of C_0 is equal to $t \times n_z$. It is much smaller than the size of system (44).

The crucial observation for the implementation algorithm is that the components of the solution vector $\bar{\eta}$ in (44) have a special structure. Namely, in mesh layer $\Omega_{xy,h} \times [z_{k-1}; z_k]$ all the components of the solution vector $\bar{\eta}$ corresponding to the interior of $G_{h,s}$, $1 \le s \le t$, are equal, and all the components of $\bar{\eta}$, corresponding to the interfaces $\Gamma_{s,j}$, $1 \le j \le l_s$, between neighboring subdomains $G_{h,s}$ and $G_{h,s'}$, $s' \ne s$, or between $G_{h,s}$ and the boundary of Ω_{xy}, are equal.

For instance, if $G_{h,s}$ is a polygon with six interfaces $\Gamma_{s,j}$, $j = \overline{1,\ 6}$, then the components of the subvector of $\bar{\eta}$ assigned for this subdomain may take only seven different values.

In the matrix form, the above property of the vector $\bar{\eta}$ in (44) can be presented by the formula

$$\bar{\eta} \;=\; R\,\bar{\psi} \tag{46}$$

where $\bar{\psi} \in \mathbb{R}^{n_c}$ and R is an $n \times n_c$ matrix. Here, $n_c = n_{xy,c} \times n_z$ where $n_{xy,c}$ is equal to the total number of subdomains $G_{h,s}$ and different interfaces $\Gamma s, j$, $j = \overline{1,\ l_s}$, $s = \overline{1,\ t}$. It is clear that the matrix R has only one non-zero entry in each row, and this entry is equal to one. Thus, system (44) can be replaced by an equivalent system

$$B_c\,\bar{\psi} \;=\; \bar{\phi} \tag{47}$$

where

$$B_c \;=\; R^T\,B\,R \quad \text{and} \quad \bar{\phi} \;=\; R^T\,\bar{\xi}. \tag{48}$$

Here, B_c is said to be a coarse mesh matrix.

The above implementation algorithm can be presented in the form of the two-step iterative procedure: $\bar{v}_0 = 0$,

$$\begin{aligned} \bar{v}_1 &= \bar{v}_0 - B_0^{-1}\left(B\,\bar{v}_0 \;-\; \bar{g}\right), \\ \bar{v}_2 &= \bar{v}_1 - R\,B_c^{-1}\,R^T\left(B\,\bar{v}_1 \;-\; \bar{g}\right) \end{aligned} \tag{49}$$

where $v_2 = B^{-1}\bar{g}$ is the solution vector of system (34).

Let us introduce the matrix

$$T \;=\; \big(I \;-\; R\,B_c^{-1}\,R^T\,B\big)\,\big(I \;-\; B_0^{-1}\,B\big). \tag{50}$$

Then we get

$$\bar{v}_2 \;=\; \big(I \;-\; T\big)\,B^{-1}\,\bar{g} \tag{51}$$

where I is the identity $n \times n$ matrix. Because $\bar{v} = \bar{v}_2$ we get the formula

$$H \;\equiv\; B^{-1} \;=\; \big(I \,-\, T\big)\,B^{-1}. \tag{52}$$

It follows immediately that T is the null matrix.

5 A Better Two-Level Preconditioner

In this Section, we derive another preconditioner for the matrix A in (3) which is spectrally equivalent to preconditioner H in (52) but its implementation is much cheaper.

Let us complement iterative procedure (49) with one additional iteration step:

$$\bar{v}_3 \;=\; \bar{v}_2 \;-\; B_0^{-1}\,\big(B\,\bar{v}_2 \;-\; \bar{g}\big). \tag{53}$$

It is obvious that $\bar{v}_3 = \bar{v}_2$. Thus, we derived an alternative representation

$$B^{-1} \;=\; \Big[I \;-\; \big(I \,-\, B_0^{-1}\,B\big)\,T\Big]\,B^{-1} \tag{54}$$

for the matrix $H = B^{-1}$.

Let a matrix $\widehat{B}_c$ be spectrally equivalent to the matrix B_c in (48), i.e. the inequalities

$$q_1\,\big(\widehat{B}_c\bar{u},\;\bar{u}\big) \;\le\; \big(B_c\bar{u},\;\bar{u}\big) \;\le\; q_2\,\big(\widehat{B}_c\bar{u},\;\bar{u}\big) \tag{55}$$

hold for all $\bar{u} \in \mathbb{R}^{n_c}$ with positive constants q_1 and q_2 independent of the coefficients of the diffusion operator in (1) and the mesh Ω_h.

Let us introduce the matrix

$$\widehat{T} \;=\; \big(I \;-\; B_0^{-1}\,B\big)\,\big(I \;-\; q_3\,R\,\widehat{B}_c^{-1}\,R^T\,B\big)\,\big(I \;-\; B_0^{-1}\,B\big) \tag{56}$$

where q_3 is a positive constant independent of the coefficients in (1) and the mesh Ω_h, and satisfying the inequality $q_3 < 2/q_2$. Then, the matrix

$$\widehat{H} \;=\; I \;-\; \widehat{T} \tag{57}$$

is spectrally equivalent to the matrix H in (52).

To describe the derivation procedure for a matrix $\widehat{B}_c$ we consider a polygonal subdomain $G_{h,s}$ with interface boundaries $\Gamma_{s,j}$, $j = \overline{1,\;l_s}$, $1 \le s \le t$. In this case, we have

$$\sigma_s^2 \;=\; |G_{h,s}| \;+\; h_f \sum_{j=1}^{l_s} |\Gamma_{s,j}|. \tag{58}$$

Here, $|G_{h,s}|$ is the area of $G_{h,s}$ and $|\Gamma_{s,j}|$ is the length of $\Gamma_{s,j}$, $j = \overline{1,\ l_s}$, $1 \le s \le t$.

To define the matrix $\widehat{B}_c$, we replace each submatrix

$$R_s^T \left(D_s \;-\; D_s\, \bar{w}_{1,s}\, \bar{w}_{1,s}^T\, D_s \right) R_s \tag{59}$$

in the matrix B_c, where R_s is the underlying block in R, by the matrix

$$\frac{h_f\, |G_{h,s}|}{\sigma_s^2} \left(\widehat{D}_s \;-\; \check{D}_s\, Q_s\, \check{D}_s \right) \tag{60}$$

where $\widehat{D}_s = \mathrm{diag}\big\{ \sum\limits_{j=1}^{l_s} |\Gamma_{s,j}|, |\Gamma_{s,1}|, \ldots, |\Gamma_{s,l_s}| \big\}$, $\check{D}_s = \mathrm{diag}\big\{1, |\Gamma_{s,1}|, \ldots, |\Gamma_{s,l_s}|\big\}$, and

$$Q_s \;=\; \begin{pmatrix} 0 & 1 & \ldots & 1 \\ 1 & 0 & \ldots & 0 \\ \vdots & \vdots & & \vdots \\ 1 & 0 & \ldots & 0 \end{pmatrix} \in \mathbb{R}^{(l_s+1)\times(l_s+1)}. \tag{61}$$

It can be proved that the matrix $\widehat{B}_c$ is spectrally equivalent to the matrix B_c with

$$q_1 \;=\; 1 \qquad \text{and} \qquad q_2 \;=\; 1 \;+\; \max_{1\le s\le t} \frac{h_f \sum\limits_{j=1}^{l_s} |\Gamma_{s,j}|}{|G_{h,s}|}. \tag{62}$$

Due to the regularity and quasiuniformity assumptions about the mesh $\Omega_{xy,h}$ and the partitioning of $\Omega_{xy,h}$ into subdomains $G_{h,s}$, $s = \overline{1,\ t}$, the value of q_2 in (62) is bounded from above by a positive constant c_4 which is independent of the coefficients a_{xy}, a_z, and c in (1) as well as of the mesh Ω_h. Thus, the matrix $\widehat{H}$ in (56), (57) with $q_3 < 2/c_4$ is spectrally equivalent to the matrix $H = B^{-1}$.

Numerical results in Section 6 are given for the PCG method with the preconditioner (56), (57) defined in this Section.

Let us denote the matrix $\widehat{B}_c$ by A_c, i.e. $A_c = \widehat{B}_c$. The matrix A_c can be presented as the 2×2 block matrix similar to presentation (3) for the matrix A:

$$A_c \;=\; \begin{pmatrix} A_{c,p} & A_{c,p\lambda} \\ A_{c,\lambda p} & A_{c,\lambda} \end{pmatrix} \tag{63}$$

where $A_{c,\lambda}$ is a diagonal matrix. It can be easily shown that the Schur complement

$$K_c \;=\; A_{c,p} \;-\; A_{c,p\lambda}\, A_{c,\lambda}^{-1}\, A_{c,\lambda p} \tag{64}$$

of the matrix A_c has the same structure as the original matrix K in (2).

Remark 1. The size of the matrix K_c in (64) is at least 2.5 times smaller than the size of the matrix B_c in (48). To this end, the Cholesky factorization of the matrix K_c is at least fifteen times cheaper than the same factorization of the matrix B_c. Thus, it can be shown that in the case $h_c \sim \sqrt{h_f}$ the PCG-method with the preconditioner $\widehat{H}$ proposed in this Section is more efficient than with the preconditioner H proposed in Sections 3 and 4.

Remark 2. Due to the structure of the matrix K_c in (64), we can design a two-level preconditioner $H_{c,p}$ for this matrix using the same coarsening technique. Replacing the matrix K_c^{-1} in the definition of $\widehat{H}$ by the matrix $H_{c,p}$ we get a three-level preconditioner.

Remark 3. The number of iterations of the PCG method with the proposed preconditioner is $O(h_f^{-1}h_c|\ln h_f|)$. The factorization of the matrices B_c and $\widehat{B}_c$ defined in (48) and (58)-(62), respectively, requires $O(h_c^{-6}h_f^{-1})$ arithmetic operations. Then, the solution of algebraic systems with factorized matrices B_c and $\widehat{B}_c^{-1}$ requires $O(h_c^{-4}h_f^{-1})$ arithmetic operations. The PCG method is faster for smaller values of h_c but implementation algorithms are more expensive. A reasonable choice is $h_c = \sqrt{h_f}$. In this case, the factorization of the matrices B_c and $\widehat{B}_c$ requires $O(h_f^{-4})$ arithmetic operations, and the implementation of the PCG with the factorized matrices B_c and $\widehat{B}_c$ requires $O(h_f^{-7/2}|\ln\ h_f|)$ arithmetic operations.

6 Numerical Results

To demonstrate the performance of the proposed two-level preconditioner we consider two examples. For both examples we compare the number of iterations and the total CPU time of the PCG-method with two-level preconditioner (TLP) and with the block Jacobi preconditioner (BJP). Both preconditioners are applied to system (2). The block Jacobi preconditioner is defined by

$$B_J \;=\; D_{xy} \;+\; K_z \tag{65}$$

where D_{xy} is the diagonal of the matrix K_{xy} in (8).

In the first example, the cubic domain Ω is partitioned into eight equal subcubes Ω_l, $l = \overline{1,\ 8}$. The coefficients a_z and c are equal to one. The coefficient a_{xy} is equal to one in four subdomains. In the other four subdomains the value of the coefficient a_{xy} is shown in Table 1. The distribution of two different values of the coefficient a_{xy} is based on the 3D-chess ordering of the subdomains. The mesh Ω_h is cubic with the mesh step size $h = 10^{-2}$. The square domain Ω_{xy} is partitioned into 100 square subdomains $G_{h,s}$, $s = \overline{1,\ 100}$. The coarse mesh matrix K_c is a block tridiagonal matrix (100 blocks, each block is a 10×10 matrix). The stopping criterion is to reduce the K-norm of the original error vector in 10^6 times.

Table 1. Variable $a_{xy}^{(2)}$, cubic mesh

	TLP $h_c = \sqrt{h_f}$		z-line BJP		Speed up TLP vs. BJP
$a_{xy}^{(2)}$	#it	CPU	#it	CPU	
10	64	23.2	984	212.	9.3
100	62	22.7	2336	491.	22.0
1000	61	22.2	6793	1450.	66.5

In the second example, $\Omega_{xy,h}$ is a uniform hexagonal mesh, and the shape of Ω_{xy} depends on the mesh. The domain Ω is again partitioned into eight subdomains as shown in Figure 2. The coefficients a_z and c are equal to one. The coefficient a_{xy} is equal to one in four subdomains and is equal to 100 in four others. The distribution of two values for a_{xy} is done in the 3D-chess order similar to Example 1. The mesh $\Omega_{xy,h}$ is partitioned into t identical subdomains $G_{h,s}$, $s = \overline{1,\ t}$, where t is equal to 36, 64, 100, and 144. In Table 2, the number of iterations and CPU time for the proposed two-level preconditioner on a sequence of meshes are given versus to the block Jacobi preconditioner.

Table 2. Sequence of hexagonal meshes $\Omega_{xy,h}$

	TLP $h_c = \sqrt{h_f}$		z-line BJP		Speed up TLP vs. BJP
$1/h_f$	#it	CPU	#it	CPU	
36	44	0.62	839	8.42	13.5
64	56	7.33	1463	90.3	12.3
100	68	25.5	2346	545.	21.4
144	81	116.	3391	2466.	21.3

The numerical results confirm the theoretical statements in Section 3: the number of iterations does not depend on the values of the coefficients, and the condition number of the matrix H_pK is proportional to $\left(h_c/h_f\right)^2$.

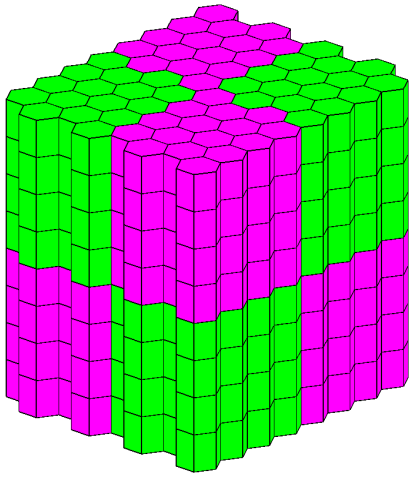

Fig. 2. Domain Ω with hexagonal mesh $\Omega_{xy,h}$

More detailed description of implementation algorithms as well as other results of numerical experiments can be found in [1].

Acknowledgement. This work was supported by ExxonMobil Upstream Research Company. The author is grateful to O. Boiarkine, I. Kapyrin, and N. Yavich for providing the numerical results for this paper. The author is also grateful to the referee for useful remarks.

References

[1] O. Boiarkine, I. Kapyrin, Yu. Kuznetsov, and N. Yavich. Numerical analysis of two-level preconditioners for diffusion equations with anisotropic coefficients. *Russian J. Numer. Anal. Math. Modelling*, 22(3), 2007. To appear.

[2] Yu. Kuznetsov and K. Lipnikov. An efficient iterative solver for a simplified poroelasticity problem. *East-West J. Numer. Math.*, 8(3):207–221, 2000.

[3] Yu. Kuznetsov and G. Marchuk. Iterative methods and quadratic functionals. In J.-L. Lions and G. Marchuk, editors, *Méthodes de l'Informatique–4*, pages 3–132, Paris, 1974. Dunod. In French.

[4] Yu. Kuznetsov and S. Repin. New mixed finite element method on polygonal and polyhedral meshes. *Russian J. Numer. Anal. Math. Modelling*, 18(3):261–278, 2003.

[5] Yu. A. Kuznetsov. Matrix iterative methods in subspaces. In *Proceedings of the International Congress of Mathematicians, Vol. 1, 2 (Warsaw, 1983)*, pages 1509–1521, Warsaw, 1984. PWN.

[6] Yu. A. Kuznetsov. Two-level preconditioners with projectors for unstructured grids. *Russian J. Numer. Anal. Math. Modelling*, 15(3-4):247–255, 2000.