MATHEMATICS OF COMPUTATION
Volume 79, Number 272, October 2010, Pages 2059–2078
S 0025-5718(10)02361-6
Article electronically published on April 26, 2010

NONOVERLAPPING DOMAIN DECOMPOSITION METHODS WITH A SIMPLE COARSE SPACE FOR ELLIPTIC PROBLEMS

QIYA HU, SHI SHU, AND JUNXIAN WANG

Abstract. We propose a substructuring preconditioner for solving three-dimensional elliptic equations with strongly discontinuous coefficients. The new preconditioner can be viewed as a variant of the classical substructuring preconditioner proposed by Bramble, Pasiack and Schatz (1989), but with much simpler coarse solvers. Though the condition number of the preconditioned system may not have a good bound, we are able to show that the convergence rate of the PCG method with such substructuring preconditioner is nearly optimal, and also robust with respect to the (possibly large) jumps of the coefficient in the elliptic equation.

.

1. Introduction

Nonoverlapping domain decomposition methods (DDMs), which are often used as preconditioners, are efficient techniques for solving large-scale discretized partial differential equations (especially those with strongly discontinuous coefficients). This type of preconditioners have been extensively investigated for various models in literature (cf., [1, 3, 4, 5, 6, 8, 9, 10, 11, 12, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 30]). A key component in such preconditioners is the construction of suitable coarse spaces, which can vary greatly for different models (compare [15], [16], [25] with [5], [20]). In particular, the design of coarse spaces for problems in three-dimensions is in general much more complicated than that for two-dimensional problems (compare [5], [16], [25] with [1], [4], [27]).

Sophisticated coarse spaces are needed for three-dimensional problems so that condition numbers of the resulting preconditioned systems are nearly optimal with respect to the mesh sizes and are independent of jumps of the coefficients in the underlying equations. It is clear that the most natural and the simplest coarse space is the space that consists of finite element functions associated with the coarse triangulation generated by the domain decomposition (cf. [8] and [30]). But, the condition number of the resulting preconditioned system is not quasi-optimal yet

Received by the editor October 9, 2008 and, in revised form, June 7, 2009.
2010 *Mathematics Subject Classification.* Primary 65F10, 65N30.
Key words and phrases. Domain decomposition, coarse subspace, substructuring preconditioner, inexact solver, condition number.
The first author was supported by Natural Science Foundation of China G10771178, The Key Project of Natural Science Foundation of China G10531080, and National Basic Research Program of China G2005CB321702.
The second author was supported by Natural Science Foundation of China (10676031, 10771178), the Basic Research Program of China under grant 2005CB321702.

for the case with large jumps (see, for example, [8] and [30]). According to this, the simplest coarse space has been regarded to be impractical in nonoverlapping DDMs for a long time. On the other hand, it is known that a convergence *rate* of the CG method is hardly affected by a few small eigenvalues of the iteration matrix. In particular, a more exact estimate of convergence of CG iteration was built by [2]. The estimate in [2] means that convergence *rate* of CG iteration depends only on the so-called *reduced condition number*, which can be roughly viewed as the condition number restricted in a large subspace of the underlying solution space. Based on this result, an excited convergence of the PCG method with a multigrid preconditioner for solving three-dimensional elliptic problems with jump coefficients was obtained recently by [29]. The results in [29] indicate that the PCG method with a multigrid preconditioner is robust with the jumps of the coefficients in the underlying equations, even if the condition number of the preconditioned system itself is not satisfactory.

In the present paper, mainly motivated by the work of Xu and Zhu [29], we still consider the simplest coarse space mentioned above. We introduce two substructuring preconditioners for solving the discrete system of three-dimensional elliptic equations with strongly discontinuous coefficients, one is an additive preconditioner, and the other is a multiplicative preconditioner. In the two preconditioners, we choose the simplest coarse space mentioned above instead of the usual complicated coarse spaces. More importantly, since this coarse space does not involve any action of the harmonic extension, we can use completely inexact subdomain solvers in our preconditioners. We will show that the convergence *rate* of the PCG method with this substructuring preconditioner is dependent only of the logarithm of the dimension of the local problem associated with an individual substructure, and is independent of possible large jumps of the coefficients in the elliptic equations, although the condition number of the resulting preconditioned system itself is not quasi-optimal. Our numerical experiments show that the multiplicative preconditioner, which is specially designed in this paper, possesses much faster convergence than the additive preconditioner.

The outline of the paper is as follows. In Section 2, we introduce a triangulation based on domain decomposition and give the corresponding discretization system. In Section 3, we recall a more exact convergence result of the PCG iterative method. The new substructuring preconditioners and their convergence are described in Section 4. In Section 5, we prove the main convergence results. Some numerical results are given in Section 6.

2. Nonoverlapping domain decomposition

Consider the model problem

$$\begin{cases} -\operatorname{div}(\omega\nabla u) = f, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega, \end{cases} \tag{2.1}$$

where Ω is a bounded and connected Lipschitz domain in $\mathcal{R}^3$, and the coefficient function $\omega(x)$ is a positive function in $L^\infty(\Omega)$.

Let $H_0^1(\Omega)$ denote the standard Sobolev space, and define the bilinear form

$$\mathcal{A}(v,w) = \int_\Omega \omega\nabla v\cdot\nabla w dx, \quad v,w\in H_0^1(\Omega).$$

Let $(\cdot,\cdot)$ denote the $L^2(\Omega)$-inner product. The weak formulation of (2.1) is: *Find* $u \in H_0^1(\Omega)$ *such that*

$$\mathcal{A}(u,v) = (f,v), \quad \forall v \in H_0^1(\Omega). \tag{2.2}$$

In the following, we define a discrete problem of (2.2) based on triangulations associated with nonoverlapping domain decompositions.

Assume that Ω can be written as the union of polyhedral subdomains $D_1, \cdots, D_{N_0}$: $\bar{\Omega} = \bigcup_{l=1}^{N_0} \bar{D}_l$, such that $\omega(x) = \omega_l$ (positive constant) for $x \in D_l$. Note that N_0 is a *fixed* constant in applications. For a number $d \in (0,\ 1)$, let each polyhedron D_l be decomposed into the union of nonoverlapping tetrahedra (or hexahedra) $\{\Omega_k\}$ with the size d. Then, we get a nonoverlapping domain decomposition for Ω: $\bar{\Omega} = \bigcup_{k=1}^{N} \bar{\Omega}_k$. Assume that $\Omega_i \cap \Omega_j = \emptyset$ when $i \neq j$; if $i \neq j$ and $\partial\Omega_i \cap \partial\Omega_j \neq \emptyset$, then $\partial\Omega_i \cap \partial\Omega_j$ is a common face of Ω_i and Ω_j, or a common edge of Ω_i and Ω_j, or a common vertex of Ω_i and Ω_j. It is clear that the subdomains $\Omega_1, \cdots, \Omega_N$ constitute a *coarse* triangulation $\mathcal{T}_d$ of Ω. If $\partial\Omega_i \cap \partial\Omega_j$ is just a common face of Ω_i and Ω_j, then set $\Gamma_{ij} = \partial\Omega_i \cap \partial\Omega_j$. Define $\Gamma = \bigcup \Gamma_{ij}$.

With each subdomain Ω_k we associate a regular triangulation made of tetrahedral elements (or hexahedral elements). We require that the triangulations in the subdomains match on the interfaces between subdomains, and so they constitute a triangulation $\mathcal{T}_h$ on the domain Ω. We denote by h the mesh size of $\mathcal{T}_h$, i.e., h denotes the maximum diameter of tetrahedra in the mesh $\mathcal{T}_h$. By $\mathcal{N}_h$ we denote the set of nodes in $\mathcal{N}_h$. Define $V_h(\Omega) \subset H_0^1(\Omega)$ as the space consisting of continuous piecewise linear functions associated with $\mathcal{T}_h$.

The discrete problem of (2.2) is: *Find* $u_h \in V_h(\Omega)$ *such that*

$$\mathcal{A}(u_h,v) = (f,v), \quad \forall v \in V_h(\Omega). \tag{2.3}$$

3. Preconditioned conjugate gradient (PCG) method

Let $A : V_h(\Omega) \to V_h(\Omega)$ be the discrete operator defined by

$$(Av_h, w_h) = \mathcal{A}(v_h, w_h), \quad v_h \in V_h(\Omega),\ \forall w_h \in V_h(\Omega).$$

Then, the discrete variational problem (2.3) can be written as the operator form

$$Au_h = f_h, \quad u_h \in V_h(\Omega). \tag{3.1}$$

In general, the space $V_h(\Omega)$ has very high dimensions, so the system needs to be solved by some iterative method, for example, the CG method . It is well known that the condition number $\kappa(A)$, which can be estimated by

$$\kappa(A) \lesssim \frac{\max\limits_{x\in\Omega} \omega(x)}{h^2 \min\limits_{x\in\Omega} \omega(x)},$$

is very great for small h or large jump coefficient $\omega(x)$. Thus, we need to construct an efficient preconditioner B for A, and solve (3.1) by the PCG method, i.e., solve the equivalent system by the CG method

$$BAu_h = Bf_h.$$

Let $\|\cdot\|_A$ be the norm induced by the positive definite operator A, and let $\kappa(BA)$ denote the condition number of BA associated with the inner product $(\cdot,\cdot)_A$. As

usual, the convergence of the PCG method is described as

$$\|u_n - u_h\|_A \le 2\left(\frac{\sqrt{\kappa(BA)}-1}{\sqrt{\kappa(BA)}+1}\right)^n \|u_0 - u_h\|_A. \tag{3.2}$$

This convergence tells us that the PCG method converges fast provided that $\kappa(BA)$ is small. But, in some applications, it is difficult to construct a preconditioner B satisfying the requirement that not only the action of B is cheap, but also $\kappa(BA)$ is small. A natural question is whether the estimate (3.2) is sharp. In fact, many existing numerical experiments indicate that the PCG method still possesses fast convergence when BA has a few small eigenvalues only, although $\kappa(BA)$ is great. This means that convergence of the PCG method may be described by a more exact estimate than (3.2). In this section, we give such a simplified result, and more detailed discussion can be found in [29] and its references.

Let $\lambda_1 \le \cdots \le \lambda_M$ be all the eigenvalues of BA associated with the inner product $(\cdot,\cdot)_A$. Assume that there exists a small positive integer m, such that

$$0 < \lambda_1 \le \cdots \le \lambda_m \ll \lambda_{m+1} \le \cdots \le \lambda_M.$$

Namely, BA has m small eigenvalues only. As in [29], define the *reduced condition number* $\kappa_{m+1}(BA)$ by

$$\kappa_{m+1}(BA) = \frac{\lambda_M}{\lambda_{m+1}}.$$

The following two results can be found in [29].

Proposition 3.1. *The convergence of PCG iteration can be estimated by*

$$\|u_n - u_h\|_A \le 2(\kappa(BA)-1)^m \left(\frac{\sqrt{\kappa_{m+1}(BA)}-1}{\sqrt{\kappa_{m+1}(BA)}+1}\right)^{n-m} \|u_0 - u_h\|_A. \quad \square \tag{3.3}$$

Proposition 3.2. *Let $\mathcal{V}$ be a subspace of $V_h(\Omega)$, with $dim(\mathcal{V}) = M - m$. Then,*

$$\lambda_{m+1} \ge \min_{0\ne v\in\mathcal{V}} \frac{(BAv, Av)}{(v, Av)}. \quad \square \tag{3.4}$$

From Proposition 3.2, one can see that λ_{m+1} can be viewed as the minimal eigenvalue of the restriction of BA on the subspace $\mathcal{V}$. In particular, if one of the eigenfunctions associated with λ_M belongs to $\mathcal{V}$, then the reduced condition number $\kappa_{m+1}(BA)$ can be viewed as the condition number of the restriction of BA on the subspace $\mathcal{V}$.

4. Substructuring preconditioners

This section is devoted to introduction of two substructuring preconditioners with the simplest coarse space and inexact subdomain solvers. For such substructuring preconditioners, the resulting preconditioned systems may possess nearly optimal *reduced condition numbers*, although they have "bad" minimal eigenvalues. By the results described in the last section, PCG iteration with these preconditioners still possesses a fast convergence rate.

4.1. **A space decomposition.** Let $V_d(\Omega)$ denote the space consisting of continuous piecewise linear functions associated with the *coarse* triangulation $\mathcal{T}_d$. It is clear that a function in $V_d(\Omega)$ is uniquely determined by the values of the function at the *cross-points* (i.e., the internal nodes of this coarse triangulation). Moreover, we have $V_d(\Omega) \subset V_h(\Omega)$.

Let $\mathcal{W}_k$ denote the *basket-set* of Ω_k, i.e., the union of the edges and the vertices of the tetrahedron (or hexahedron) Ω_k. Set $\mathcal{W} = \bigcup_{k=1}^N \mathcal{W}_k$, and define the *wire-basket* subspace:

$$V_h^{\mathcal{W}}(\Omega) = \left\{ v_h \in V_h(\Omega) : \ v_h \text{ vanishes at all the nodes away from } \mathcal{W} \right\}.$$

For two neighboring subdomains Ω_i and Ω_j, set $\Omega_{ij} = \Omega_i \cup \Omega_j \cup \Gamma_{ij}$, and define

$$V_h^0(\Omega_{ij}) = \{ v_h \in V_h(\Omega) : \ \text{supp } \ v_h \subset \Omega_{ij} \}.$$

Then, we have a space decomposition

$$V_h(\Omega) = V_d(\Omega) + V_h^{\mathcal{W}}(\Omega) + \sum_{\Gamma_{ij}} V_h^0(\Omega_{ij}). \tag{4.1}$$

Remark 4.1. Note that, besides the coarse space $V_d(\Omega)$, an extra subspace $V_h^{\mathcal{W}}(\Omega)$ is involved in the decomposition (4.1). However, the subspace $V_h^{\mathcal{W}}(\Omega)$ possesses a very particular structure, which makes the design of a cheap solver on it to be easy (see the next subsection).

4.2. **An additive preconditioner.** In this subsection, we first introduce an additive preconditioner for the operator A.

For convenience, let φ_p denote the nodal basis function corresponding to the node p. It is easy to see that

$$(Av_h, \ v_h) \eqsim \sum_{p \in \mathcal{N}_h \cap \mathcal{W}} v_h^2(p) \mathcal{A}(\varphi_p, \varphi_p), \quad v_h \in V_h^{\mathcal{W}}(\Omega).$$

This means that an inexact solver $B_{\mathcal{W}} : V_h^{\mathcal{W}}(\Omega) \to V_h^{\mathcal{W}}(\Omega)$ can be defined by

$$(B_{\mathcal{W}} v_h, \ w_h) = \sum_{p \in \mathcal{W}_h \cap \mathcal{W}} v_h(p) w_h(p) \mathcal{A}(\varphi_p, \varphi_p), \quad v_h \in V_h^{\mathcal{W}}(\Omega), \ \forall w_h \in V_h^{\mathcal{W}}(\Omega).$$

The action of $B_{\mathcal{W}}^{-1}$ can be expressed explicitly as

$$B_{\mathcal{W}}^{-1} g = \sum_{p \in \mathcal{N}_h \cap \mathcal{W}} \frac{(g, \varphi_p)}{\mathcal{A}(\varphi_p, \varphi_p)} \varphi_p, \quad g \in V_h^{\mathcal{W}}(\Omega).$$

It is known that $B_{\mathcal{W}}^{-1}$ is just the well-known Jacobi smoother.

Let $B_d : V_d(\Omega) \to V_d(\Omega)$ and $B_{ij} : V_h^0(\Omega_{ij}) \to V_h^0(\Omega_{ij})$ be two symmetric and positive definite operators which are spectrally equivalent to the restrictions of A on $V_d(\Omega)$ and $V_h^0(\Omega_{ij})$, respectively. Namely,

$$(B_d v_h, \ v_h) \eqsim \int_\Omega \omega |\nabla v_h|^2 dx, \quad \forall v_h \in V_d(\Omega) \tag{4.2}$$

and

$$(B_{ij} v_h, \ v_h) \eqsim \omega_i \int_{\Omega_i} |\nabla v_h|^2 dx + \omega_j \int_{\Omega_j} |\nabla v_h|^2 dx, \quad \forall v_h \in V_h^0(\Omega_{ij}). \tag{4.3}$$

In applications, the solver B_{ij} can be chosen as a symmetric multigrid solver for the restriction of A on $V_h^0(\Omega_{ij})$. The coarse solver B_d can be simply chosen as the

restriction of the operator A on $V_d(\Omega)$, since the dimension of $V_d(\Omega)$ is in general very low.

Then, a preconditioner for A is defined as

$$B = B_d^{-1}Q_d + B_{\mathcal{W}}^{-1}Q_{\mathcal{W}} + \sum_{\Gamma_{ij}} B_{ij}^{-1}Q_{ij}, \tag{4.4}$$

where Q_d, $Q_{\mathcal{W}}$ and Q_{ij} denote the L^2 projector into the corresponding subspace, respectively.

Remark 4.2. The preconditioner B is different from most existing substructuring preconditioners (cf. [22], [23], [26] and [30]): it does not involve "exact" subdomain solvers associated with the subdomains Ω_k $(k = 1, \cdots, N)$. A similar preconditioner for two-dimensional problems was considered in [10]. We emphasize that B_{ij}^{-1} in the preconditioner B is an "inexact" solver, so its action is inexpensive.

In the rest of the paper, we study the spectrum of BA on a special subspace of $V_h(\Omega)$. To this end, let $J = \{k : \ \partial D_k \cap \partial\Omega = \emptyset\}$ denote the index set of the subdomains $\{D_k\}_{k=1}^{N_0}$ which do not touch the boundary of Ω, and set

$$\tilde{V}_h(\Omega) = \{v_h \in V_h(\Omega) : \ \int_{D_k} v_h dx = 0, \ k \in J\}.$$

Let m_0 denote the number of the indices in J, and let $\kappa_{m_0+1}(BA)$ denote the *reduced condition number* of BA associated with the subspace $\tilde{V}_h(\Omega)$. Namely,

$$\kappa_{m_0+1}(BA) = \frac{\lambda_{\max}(BA)}{\lambda_{m_0+1}(BA)},$$

where $\lambda_{m_0+1}(BA)$ is the minimal eigenvalue of the restriction of BA on the subspace $\tilde{V}_h(\Omega)$.

For ease of notation, following [28], the symbols $\lesssim$, $\gtrsim$ and $\eqsim$ will be used in the rest of this paper. That $x_1 \lesssim y_1$, $x_2 \gtrsim y_2$ and $x_3 \eqsim y_3$, mean that $x_1 \leq C_1 y_1$, $x_2 \geq c_2 y_2$ and $c_3 x_3 \leq y_3 \leq C_3 x_3$ for some constants C_1, c_2, c_3 and C_3 which are independent of h and d.

Theorem 4.1. *For the preconditioner B defined by* (4.4)*, we have*

$$\lambda_{m_0+1}(BA) \gtrsim \frac{1}{\log(1/d)\log^2(d/h)} \quad \textit{and} \quad \kappa_{m_0+1}(BA) \lesssim \log(1/d)\log^2(d/h). \tag{4.5}$$

When the coefficient $\omega(x)$ has no large jump across the interface Γ, or there is no cross-point in the distribution of the jumps of the coefficient, the factor $\log(1/d)$ *in the above inequalities can be removed.*

Remark 4.3. Theorem 4.1 indicates that the *reduced condition number* $\kappa_{m_0+1}(BA)$ is nearly optimal with respect to the number of subdomains and the dimension of the local problem, and is independent of possible large jumps of the coefficient $\omega(x)$. Thus, from Proposition 3.1, we know that PCG iteration for solving (3.1) with the preconditioner B possesses fast convergence rate.

The following result can be proved in the standard manner by using the estimates of the weighted L^2 projector given in [7]:

Theorem 4.2. *For the preconditioner B defined by* (4.4)*, we have*

$$\lambda_{\min}(BA) \gtrsim \frac{h}{d\log^2(d/h)} \quad \textit{and} \quad \mathrm{cond}(BA) \lesssim \frac{d}{h}\log^2(d/h). \tag{4.6}$$

If the coefficient $\omega(x)$ has no large jump across the interface Γ, or there is no cross-point in the distribution of the jumps of the coefficient, then

$$\lambda_{\min}(BA) \gtrsim \frac{1}{\log^2(d/h)} \quad \textit{and} \quad \mathrm{cond}(BA) \lesssim \log^2(d/h).$$

Remark 4.4. Theorem 4.2 indicates that the minimal eigenvalue of BA in general strongly depends on the dimension d/h of the local spaces $V_h(\Omega_k)$. But, one can see from Theorem 4.1 that the number of such "bad" small eigenvalues is not greater than N_0, which is a small (fixed) positive integer in applications.

The action of the preconditioner B can be described by the following algorithm.

Algorithm 4.1. *For $g \in V_h(\Omega)$, the solution $u_g = Bg \in V_h(\Omega)$ can be obtained as follows:*

Step 1. Computing $u_d \in V_d(\Omega)$ by

$$(B_d u_d, v_h) = (g, v_h), \ \forall v_h \in V_d(\Omega).$$

Step 2. Computing $u_{\mathcal{W}} \in V_h^{\mathcal{W}}(\Omega)$ by

$$(B_{\mathcal{W}} u_{\mathcal{W}}, v_h) = (g, v_h), \ \forall v_h \in V_h^{\mathcal{W}}(\Omega).$$

Step 3. Computing every $u_{ij} \in V_h^0(\Omega_{ij})$ in parallel by

$$(B_{ij} u_{ij}, v_h) = (g, v_h), \ \forall v_h \in V_h^0(\Omega_{ij}).$$

Step 4. Set

$$u_g = u_d + u_{\mathcal{W}} + \sum_{\Gamma_{ij}} u_{ij}.$$

Remark 4.5. From the above algorithm, one can see more clearly that the action of B is cheap and is easy to implement.

4.3. **A multiplicative preconditioner.** In this subsection, we design a *simplified* multiplicative preconditioner, inspired by [13].

For ease of notation, set

$$V_\Gamma^0(\Omega) = \sum_{\Gamma_{ij}} V_h^0(\Omega_{ij}).$$

Then, the decomposition (4.1) can be written as

$$V_h(\Omega) = V_d(\Omega) + V_h^{\mathcal{W}}(\Omega) + V_\Gamma^0(\Omega). \tag{4.7}$$

The desired multiplicative preconditioner is associated with the above space decomposition. Let $A_d : V_d(\Omega) \to V_d(\Omega)$ denote the restriction of the operator A on $V_d(\Omega)$. For convenience, set

$$B_\Gamma^{-1} = \sum_{\Gamma_{ij}} B_{ij}^{-1} Q_{ij}.$$

Let $\tilde{B}_{\mathcal{W}}^{-1}$ be a suitable damped Jacobi smoother (i.e., a scaled variant of $B_{\mathcal{W}}^{-1}$) or the symmetrized Gauss-Seidel smoother for the restriction of A on $V_h^{\mathcal{W}}(\Omega)$. Define the operators

$$P_d = A_d^{-1} Q_d A,\ T_{\mathcal{W}} = \tilde{B}_{\mathcal{W}}^{-1} Q_{\mathcal{W}} A \text{ and } T_\Gamma = B_\Gamma^{-1} A.$$

It is clear that P_d is the energy projector from $V_h(\Omega)$ into the subspace $V_d(\Omega)$.

Let I denote the identity operator on $V_h(\Omega)$, and set

$$T = I - (I - T_{\mathcal{W}})(I - T_\Gamma)(I - T_{\mathcal{W}}).$$

It is easy to see that the operator $T : V_h(\Omega) \to V_h(\Omega)$ is symmetric with respect to the inner-product $(A\cdot, \cdot)$. Define

$$M = [I - (I - P_d)(I - T)]A^{-1}. \tag{4.8}$$

Let $V_d^\perp(\Omega) \subset V_h(\Omega)$ denote the orthogonal complement of $V_d(\Omega)$ with respect to the inner-product $(A\cdot, \cdot)$. It can be verified that $(MA)|_{V_d^\perp(\Omega)}$ is a symmetric and positive definite operator with respect to the inner-product $(A\cdot, \cdot)$ (cf. [13]). Then, the operator M can be chosen as a multiplicative preconditioner in PCG iteration for solving (3.1), where the initial guess u_0 in PCG iteration would be computed in a suitable manner so that $e_0 = u_h - u_0 \in V_d^\perp(\Omega)$. We want to emphasize that the preconditioner M is different from the standard symmetrized-multiplicative preconditioner (cf. [22]): (1) the operators P_d and T_Γ appear in M only one time, instead of two times; (2) the maximal eigenvalue of T_Γ may not be less than 2.

In the following we investigate convergence rate of the PCG method for solving (3.1) with the multiplicative preconditioner M. For this purpose, set

$$\tilde{V}_d^\perp(\Omega) = \{v_h \in V_d^\perp(\Omega) : \int_{D_k} v_h dx = 0,\ k \in J\}.$$

Let $\lambda_{m_0+1}^\perp(MA)$ denote the minimal eigenvalue of the restriction of MA on the subspace $\tilde{V}_d^\perp(\Omega)$, and let $\lambda_{\min}^\perp(MA)$ and $\lambda_{\max}^\perp(MA)$ denote the minimal eigenvalue and maximal eigenvalue of the restriction of MA on the subspace $V_d^\perp(\Omega)$, respectively. Define

$$\kappa^\perp(MA) = \frac{\lambda_{\max}^\perp(MA)}{\lambda_{\min}^\perp(MA)} \text{ and } \kappa_{m_0+1}^\perp(MA) = \frac{\lambda_{\max}^\perp(MA)}{\lambda_{m_0+1}^\perp(MA)}.$$

Combining Proposition 3.1 in Section 3 with Theorem 4.1 in [13], we get

Proposition 4.1. *The convergence of PCG iteration for solving* (3.1) *with the multiplicative preconditioner M can be described as*

$$\|u_n - u_h\|_A \le 2(\kappa^\perp(MA) - 1)^{m_0} \left(\frac{\sqrt{\kappa_{m_0+1}^\perp(MA)} - 1}{\sqrt{\kappa_{m_0+1}^\perp(MA)} + 1} \right)^{n-m_0} \|u_0 - u_h\|_A. \tag{4.9}$$

The condition numbers of MA can be estimated by the following result.

Theorem 4.3. *For the preconditioner M defined by* (4.8)*, we have*

$$\kappa^\perp(MA) \lesssim \frac{d}{h} \log^2(d/h) \quad \textit{and} \quad \kappa_{m_0+1}^\perp(MA) \lesssim \log(1/d) \log^2(d/h). \tag{4.10}$$

Let the initial guess u_0 in the PCG method for solving (3.1) with the multiplicative preconditioner M be chosen in a suitable manner so that the error $e_0 = u_h - u_0 \in V_d^{\perp}(\Omega)$ (for example, choose $u_0 = P_d f_h$). The action of the preconditioner M can be described by the following algorithm.

Algorithm 4.2. *For $g \in V_h(\Omega)$, the solution $u_g = Bg \in V_h(\Omega)$ can be obtained as follows:*

Step 1. Computing $u_{\mathcal{W}}^{(1)} \in V_h^{\mathcal{W}}(\Omega)$ by

$$(B_{\mathcal{W}} u_{\mathcal{W}}^{(1)}, v_h) = (g, v_h), \ \forall v_h \in V_h^{\mathcal{W}}(\Omega).$$

Step 2. Computing every $u_{ij} \in V_h^0(\Omega_{ij})$ in parallel by

$$(B_{ij} u_{ij}, v_h) = (g, v_h) - \mathcal{A}(u_{\mathcal{W}}^{(1)}, v_h), \ \forall v_h \in V_h^0(\Omega_{ij}),$$

and set

$$u' = u_{\mathcal{W}}^{(1)} + \sum_{\Gamma_{ij}} u_{ij}.$$

Step 3. Computing $u_{\mathcal{W}}^{(2)} \in V_h^{\mathcal{W}}(\Omega)$ by

$$(B_{\mathcal{W}} u_{\mathcal{W}}^{(2)}, v_h) = (g, v_h) - \mathcal{A}(u', v_h), \ \forall v_h \in V_h^{\mathcal{W}}(\Omega),$$

and set

$$u'' = u' + u_{\mathcal{W}}^{(2)}.$$

Step 4. Computing $u_d \in V_d(\Omega)$ by

$$(A_d u_d, v_h) = (g, v_h) - \mathcal{A}(u'', v_h), \ \forall v_h \in V_d(\Omega).$$

Step 5. Set

$$u_g = u'' + u_d.$$

Remark 4.6. Although the action of $B_{\mathcal{W}}^{-1}$ needs to be implemented two times in Algorithm 4.2, the cost of Algorithm 4.2 is almost the same as that of Algorithm 4.1, since the action of $B_{\mathcal{W}}^{-1}$ is defined explicitly. As we will see in our numerical results, Algorithm 4.2 possesses faster convergence than Algorithm 4.1 (although one cannot see this from Theorem 4.3).

4.4. **Can the space $V_h^{\mathcal{W}}(\Omega)$ be reduced?** It is clear that the first sum in the decomposition (4.1) is not a direct sum. We would like to investigate whether the space $V_h^{\mathcal{W}}(\Omega)$ in (4.1) can be replaced by another smaller space. Note that the coarse space $V_d(\Omega)$ is defined by the cross-points, it is natural to consider the space

$$\hat{V}_h^{\mathcal{W}}(\Omega) = \{v_h \in V_h^{\mathcal{W}}(\Omega) : \ v_h \text{ vanishes at the cross-points}\}.$$

Then we have the following space decomposition:

$$V_h(\Omega) = V_d(\Omega) \oplus \hat{V}_h^{\mathcal{W}}(\Omega) + \sum_{\Gamma_{ij}} V_h^0(\Omega_{ij}). \tag{4.11}$$

As before, we can define an additive preconditioner $\hat{B}$ and an multiplicative preconditioner $\hat{M}$ associated with the decomposition (4.11). However, the estimate (4.5) (resp. (4.10)) will not hold yet when replacing B by $\hat{B}$ (resp. replacing M by $\hat{M}$). In fact, as in the proof of Theorem 4.1 and Theorem 4.3, we can show:

Theorem 4.4. *For the preconditioner $\hat{B}$, we only have*

$$\kappa_{m_0+1}(\hat{B}A) \lesssim \frac{d}{h}\log^2(d/h) \text{ and } \kappa^{\perp}_{m_0+1}(\hat{M}A) \lesssim \frac{d}{h}\log^2(d/h). \tag{4.12}$$

In an analogous way to Section 5.2.2 of [30], we can explain that the two inequalities in (4.12) cannot be improved essentially. This means that the *wire-basket* subspace $V_h^{\mathcal{W}}(\Omega)$ cannot be replaced by the smaller subspace $\hat{V}_h^{\mathcal{W}}(\Omega)$. This conclusion will be illustrated further by numerical experiments in Section 6.

4.5. **Comparison with some existing preconditioners.** In this subsection, we give some comparison between the new preconditioners and some existing preconditioners.

• Comparison with BPS-type preconditioners.

The additive preconditioner B introduced in Subsection 4.2 can be viewed as a variant of the well-known BPS preconditioner (see [4]). For BPS preconditioner, a complicated coarse solver based on a large coarse subspace was designed, and every local problem associated with subdomains $\{\Omega_k\}$ should be solved exactly. An interesting substructuring preconditioner with inexact subdomain solvers was constructed in [6] by replacing the harmonic extension on each subdomain with a simple *average* extension. However, nearly optimal convergence cannot be obtained for the preconditioner in [6]. In this paper, we use two simple and cheap solvers, A_d and $B_{\mathcal{W}}$, where A_d is defined on the natural subspace $V_d(\Omega)$ associated with the initial triangulation, and the action of $B_{\mathcal{W}}^{-1}$ can be expressed explicitly. Notice that the actions of the two "coarse" solvers, i.e., solutions of u_d and $u_{\mathcal{W}}$ in Algorithm 4.1, do not involve any action of the discrete harmonic extension yet, and so no local problem in Ω_k needs to be solved exactly. Similar coarse solvers with A_d and $B_{\mathcal{W}}$ were considered in Algorithm 6.2 of [8], but exact subdomain solvers were still used there. More importantly, the estimate of convergence of that algorithm contains a factor which may depend on jumps of the coefficient. The multiplicative preconditioner M, which has the same merits as the additive preconditioner B, possesses faster convergence than B.

• Comparison with Neumann-type preconditioners.

The main merit of Neumann-type preconditioners (see [9] and [20]) is that they possess a small coarse subspace, the dimension of which equals the number of the *floating* subdomains. In short, each basis function of the coarse subspace in Neumann-type preconditioners is generated by a constant function defined on a subdomain Ω_k. But, since the zero extension of a constant does not belong to $V_h(\Omega) \subset H^1(\Omega)$, such a basis function has to be defined as a complicated extension of a constant on some subdomain Ω_k. Because of such complicated extensions, Neumann-type preconditioners are difficult to implement for three-dimensional problems. Our coarse subspace $V_d(\Omega)$ not only has low dimension, which is almost the same as the dimension of the coarse subspace in Neumann-type preconditioners, but also has natural basis functions.

• Comparison with FETI-type methods.

FETI-type methods (see [11] and [17]) has some connection with Neumann-type methods. Since Lagrange multipliers are introduced in FETI-type methods, the complicated extension in Neumann-type methods can be avoided in the construction of the coarse subspace for FETI-type methods. But, extra techniques are needed in FETI-type methods to deal with the *floating* subdomains (refer to [11] and [17]).

The method introduced in [14] has similar merits as Algorithm 4.1, but a saddle-point system needs to be solved for that method.

5. Analysis

In this section, we prove the results given in the last section. To this end, we need to recall several simple auxiliary results. For convenience, we first define some notation.

5.1. Some notation.

• Several local spaces:

For subdomain Ω_k, set

$$V_h^0(\Omega_k) = \{v \in V_h(\Omega) : \text{ supp } v \subset \Omega_k\}$$

and

$$W_h(\partial\Omega_k) = \{v|_{\partial\Omega_k} : v \in V_h(\Omega)\}.$$

For a face F of Ω_k, define

$$W_h^0(F) = \{\varphi \in W_h(\partial\Omega_k) : \text{ supp } \varphi \subset F\}.$$

Let $V_h^H(\Omega_{ij})$ denote the "discrete harmonic" subspace defined by

$$V_h^H(\Omega_{ij}) = \{v \in V_h^0(\Omega_{ij}) : \mathcal{A}(v, w) = 0, \quad \forall w \in V_h^0(\Omega_k) \text{ for } k = i, j\}.$$

• Interpolant-type operators ([30]):

For a node $p \in \mathcal{N}_h$, let φ_p denote the nodal basis function on p. We use K to denote an open subset of Ω. Define the interpolant-type operator I_K^0 by

$$I_K^0 v(x) = \sum_{p \in K \cap \mathcal{N}_h} v(p)\varphi_p(x), \quad v \in V_h(\Omega).$$

For example, given a face F of Ω_k, the *face* interpolant $I_F^0 v \in V_h(\Omega)$ satisfies

$$I_F^0 v(p) \begin{cases} v(p), & p \in F \cap \mathcal{N}_h, \\ 0, & p \in (\Omega \backslash F) \cap \mathcal{N}_h. \end{cases}$$

It is clear that we have $I_F^0 v|_{\partial\Omega_k} \in W_h^0(F)$. In the rest of this section, we will use the operators $I_{\mathcal{W}}^0$ and I_F^0 with $F = \Gamma_{ij}$.

• $H^{\frac{1}{2}}$ norms defined in the boundary of a subdomain:

For a subdomain Ω_k, define the scaled norm

$$\|\varphi\|_{\frac{1}{2},\ \partial\Omega_k} = (|\varphi|^2_{\frac{1}{2},\ \partial\Omega_k} + d^{-1}\|\varphi\|^2_{0,\ \partial\Omega_k})^{\frac{1}{2}}, \quad \forall \varphi \in H^{\frac{1}{2}}(\partial\Omega_k).$$

For a face F of $\partial\Omega_k$, define

$$\|\varphi_h\|^2_{H^{\frac{1}{2}}_{00}(F)} = |\varphi_h|^2_{\frac{1}{2},\ F} + \int_F \frac{|\varphi_h(x)|^2}{\text{dist}(x,\ \partial F)} ds(x),\ \varphi_h \in W_h^0(F),$$

where $\text{dist}(x,\ \partial F)$ denotes the shortest distance from a point $x \in F$ to the boundary ∂F. It is known that

$$\|\varphi_h\|^2_{H^{\frac{1}{2}}_{00}(F)} \overset{=}{\sim} |\tilde{\varphi}_h|^2_{\frac{1}{2},\ \partial\Omega_k},$$

where $\tilde{\varphi}_h \in W_h(\partial\Omega_k)$ denotes the zero extension of φ_h.

• Weighted norms:

In the rest of this paper, we will use repeatedly two weighted inner products associated with the positive numbers ω_l.

Define the weighted L^2-inner product

$$(v,w)_{L^2_\omega(\Omega)} = \sum_{i=1}^{N_0} \omega_l \int_{D_l} vw dx, \quad v,\ w \in L^2(\Omega)$$

and the weighted H^1-inner product

$$(v,w)_{H^1_\omega(\Omega)} = \sum_{i=1}^{N_0} \omega_l \int_{D_l} \nabla v \cdot \nabla w dx, \quad v,\ w \in H^1_0(\Omega).$$

Let $\|\cdot\|_{L^2_\omega(\Omega)}$ and $|\cdot|_{H^1_\omega(\Omega)}$ denote, respectively, the norm and the semi-norm induced by the inner product $(\cdot,\cdot)_{L^2_\omega(\Omega)}$ and $(\cdot,\cdot)_{H^1_\omega(\Omega)}$. For convenience, define

$$\|v\|_{H^1_\omega(\Omega)} = (|v|^2_{H^1_\omega(\Omega)} + d^{-2}\|v\|^2_{L^2_\omega(\Omega)})^{\frac{1}{2}}.$$

5.2. **Lemmas.** The following two results can be found in [5] and [30].

Lemma 5.1. *The following inequality holds for each face F of Ω_k:*

$$\|I^0_F v\|_{H^{\frac{1}{2}}_{00}(F)} \lesssim \log(d/h)\|v\|_{\frac{1}{2},\partial\Omega_k}, \quad \forall v \in V_h(\Omega). \quad \square \tag{5.1}$$

Lemma 5.2. *The following inequality holds for every Ω_k:*

$$\|v\|^2_{0,\mathcal{W}_k} \lesssim \log(d/h)\|v\|^2_{\frac{1}{2},\partial\Omega_k}, \quad \forall v \in V_h(\partial\Omega_k). \quad \square \tag{5.2}$$

For convenience, define the norm $\|v_h\|_{\mathcal{W}}$ by

$$\|v_h\|^2_{\mathcal{W}} = (B_{\mathcal{W}} v_h, v_h), \quad v_h \in V_h^{\mathcal{W}}(\Omega).$$

Using the definition of $B_{\mathcal{W}}$ and the discrete L^2 norms, we have

$$\|v_h\|^2_{\mathcal{W}} \eqsim \sum_{k=1}^{N} \omega_k \|v_h\|^2_{0,\mathcal{W}_k}, \quad v_h \in V_h^{\mathcal{W}}(\Omega).$$

This, together with (5.2), leads to

Corollary 5.1. *The following inequality holds:*

$$\|I^0_{\mathcal{W}} v\|^2_{\mathcal{W}} \lesssim \log(d/h)\|v\|^2_{H^1_\omega(\Omega)}, \quad \forall v \in V_h(\Omega). \tag{5.3}$$

Let $Q^\omega_d : L^2(\Omega) \to V_d(\Omega)$ be the weighted L^2 projections defined by

$$(Q^\omega_d v, w)_{L^2_\omega(\Omega)} = (v,w)_{L^2_\omega(\Omega)}, \quad \forall v \in L^2(\Omega), w \in V_d(\Omega). \tag{5.4}$$

The following results follow directly by the estimates in [7].

Lemma 5.3 (See [29])**.** *The weighted L^2 projection Q^ω_d satisfies*

$$\|(Q^\omega_d - I)v\|^2_{L^2_\omega(\Omega)} \lesssim d^2 \log(1/d)|v|^2_{H^1_\omega(\Omega)}, \quad \forall v \in \tilde{V}_h(\Omega) \tag{5.5}$$

and

$$|Q^\omega_d v|^2_{H^1_\omega(\Omega)} \lesssim \log(1/d)|v|^2_{H^1_\omega(\Omega)}, \quad \forall v \in \tilde{V}_h(\Omega). \quad \square \tag{5.6}$$

Remark 5.1. When the coefficient $\omega(x)$ has no jump across the interface Γ, or there is no cross-point in the distribution of the jumps of the coefficient, the factor $\log(1/d)$ in the inequalities (5.5) and (5.6) can be removed.

5.3. **Proofs.** Throughout this subsection, we use $(\cdot,\cdot)_A$ to denote the inner-product defined by the operator A, namely, the inner-product $(\cdot,\cdot)_{H^1_\omega(\Omega)}$, and use $\|\cdot\|_A$ to denote the norm induced by the inner-product $(\cdot,\cdot)_A$.

Proof of Theorem 4.1. In the standard manner, it can be verified that

$$\lambda_{\max}(BA) \lesssim 1.$$

In the following we prove that

$$\lambda_{m_0+1}(BA) \gtrsim \frac{1}{\log(1/d)\log^2(d/h)}.$$

It suffices to build a stable decomposition for $v_h \in V_h(\Omega)$.

For $v_h \in V_h(\Omega)$, define $v_d \in V_d(\Omega)$ as $v_d = Q_d^\omega v_h$. Let $\tilde{v}_h = v_h - v_d$, and define $\tilde{v}_{ij}^H \in V_h^H(\Omega_{ij})$ by $\tilde{v}_{ij}^H = I^0_{\Gamma_{ij}}\tilde{v}_h$ for each Γ_{ij}. For convenience, set

$$\tilde{v}_\Gamma^H = I^0_{\mathcal{W}}\tilde{v}_h + \sum_{\Gamma_{ij}} \tilde{v}_{ij}^H, \tag{5.7}$$

and define

$$\tilde{v}_k^0 = \big(\tilde{v}_h - \tilde{v}_\Gamma^H\big)|_{\Omega_k}.$$

For each k, let m_k be the number of faces that belong to $\partial\Omega_k$. Define

$$\tilde{v}_{ij} = \tilde{v}_{ij}^H + \tilde{v}_i^0/m_i + \tilde{v}_j^0/m_j. \tag{5.8}$$

Then we have the decomposition

$$v_h = v_d + I^0_{\mathcal{W}}\tilde{v}_h + \sum_{\Gamma_{ij}} \tilde{v}_{ij}. \tag{5.9}$$

In fact, we deduce, by (5.7) and the definitions of the interpolation-type operators, that

$$\tilde{v}_\Gamma^H = \tilde{v}_h \quad \text{on } \Gamma.$$

Then we get $v_k^0 \in V_h^0(\Omega_k)$ and

$$\big(\sum_{k=1}^N v_k^0\big)|_{\Omega_k} = v_k^0 = \big(\tilde{v}_h - \tilde{v}_\Gamma^H\big)|_{\Omega_k}.$$

Namely,

$$\sum_{k=1}^N v_k^0 = \tilde{v}_h - \tilde{v}_\Gamma^H.$$

Moreover, we have by (5.8),

$$\sum_{\Gamma_{ij}} \tilde{v}_{ij} = \sum_{\Gamma_{ij}} \tilde{v}_{ij}^H + \sum_{k=1}^N v_k^0.$$

Combining the above two equalities yields

$$\sum_{\Gamma_{ij}} \tilde{v}_{ij} = \sum_{\Gamma_{ij}} \tilde{v}_{ij}^H + \tilde{v}_h - \tilde{v}_\Gamma^H.$$

This, together with (5.7), leads to

$$\tilde{v}_h = (\tilde{v}_\Gamma^H - \sum_{\Gamma_{ij}} \tilde{v}_{ij}^H) + \sum_{\Gamma_{ij}} \tilde{v}_{ij} = I^0_{\mathcal{W}}\tilde{v}_h + \sum_{\Gamma_{ij}} \tilde{v}_{ij},$$

which implies (5.9). It is easy to see that

$$I^0_{\mathcal{W}}\tilde{v}_h \in V_h^{\mathcal{W}}(\Omega) \text{ and } \tilde{v}_{ij} \in V_h^0(\Omega_{ij}).$$

It suffices to verify that

$$\begin{aligned}&(B_d v_d, v_d) + \|I^0_{\mathcal{W}}\tilde{v}_h\|^2_{\mathcal{W}} + \sum_{\Gamma_{ij}}(B_{ij}\tilde{v}_{ij}, \tilde{v}_{ij})\\ &\lesssim \log(1/d)\log^2(d/h)\|v_h\|^2_A,\ \forall v_h \in \tilde{V}_h(\Omega).\end{aligned} \tag{5.10}$$

It follows by (5.6) that

$$\|v_d\|^2_A \lesssim \log(1/d)\|v_h\|^2_A. \tag{5.11}$$

From (5.3), we have

$$\|I^0_{\mathcal{W}}\tilde{v}_h\|^2_{\mathcal{W}} \lesssim \log(d/h)\|\tilde{v}_h\|^2_{H^1_\omega(\Omega)}. \tag{5.12}$$

Using the definition of $\tilde{v}_h$, together with (5.5), yields

$$\|\tilde{v}_h\|^2_{H^1_\omega(\Omega)} \lesssim \log(1/d)\|v_h\|^2_A.$$

Plugging this into (5.12) leads to

$$\|I^0_{\mathcal{W}}\tilde{v}_h\|^2_{\mathcal{W}} \lesssim \log(d/h)\log(1/d)\|v_h\|^2_A. \tag{5.13}$$

Similarly, we deduce, by the property of the harmonic extension, that

$$\begin{aligned}\|\tilde{v}^H_{ij}\|^2_A &= \omega_i|\tilde{v}^H_{ij}|^2_{1,\Omega_i} + \omega_j|\tilde{v}^H_{ij}|^2_{1,\Omega_j}\\ &\lesssim \omega_i|\tilde{v}^H_{ij}|^2_{\frac{1}{2},\partial\Omega_i} + \omega_j|\tilde{v}^H_{ij}|^2_{\frac{1}{2},\partial\Omega_j}\\ &\lesssim (\omega_i+\omega_j)|I^0_{\Gamma_{ij}}\tilde{v}_h|^2_{H^{\frac{1}{2}}_{00}(\Gamma_{ij})}.\end{aligned}$$

This, together with (5.1), gives

$$\begin{aligned}\|\tilde{v}^H_{ij}\|^2_A &\lesssim \log^2(d/h)[\omega_i\|\tilde{v}_h\|^2_{\frac{1}{2},\ \partial\Omega_i} + \omega_j\|\tilde{v}_h\|^2_{\frac{1}{2},\ \partial\Omega_j}]\\ &\lesssim \log^2(d/h)[\omega_i\|\tilde{v}_h\|^2_{1,\ \Omega_i} + \omega_j\|\tilde{v}_h\|^2_{\frac{1}{2},\ \Omega_j}].\end{aligned}$$

Thus, we get

$$\sum_{\Gamma_{ij}}\|\tilde{v}^H_{ij}\|^2_A \lesssim \log^2(d/h)\|\tilde{v}_h\|^2_{H^1_\omega(\Omega)} \lesssim \log(1/d)\log^2(d/h)\|v_h\|^2_A. \tag{5.14}$$

Here we have used (5.5) again.

In the following, we verify that

$$\|\tilde{v}_{ij}\|^2_A \lesssim \log(1/d)\log^2(d/h)\|v_h\|^2_A. \tag{5.15}$$

Since $\tilde{v}_h - \tilde{v}^H_\Gamma \in V^0_h(\Omega_k)$, we have

$$\sum_{k=1}^N \omega_k|\tilde{v}^0_k|^2_{1,\Omega_k} = \sum_{k=1}^N \omega_k|\tilde{v}_h - \tilde{v}^H_\Gamma|^2_{1,\Omega_k} = \|\tilde{v}_h - \tilde{v}^H_\Gamma\|^2_A.$$

Then we get by (5.7),

$$\begin{aligned}\sum_{k=1}^N \omega_k|\tilde{v}^0_k|^2_{1,\Omega_k} &\lesssim \|\tilde{v}_h\|^2_A + \|I^0_{\mathcal{W}}\tilde{v}_h\|^2_A + \|\sum_{\Gamma_{ij}}\tilde{v}^H_{ij}\|^2_A\\ &\lesssim \|v_h\|^2_A + \|Q^\omega_d v_h\|^2_A + \|I^0_{\mathcal{W}}\tilde{v}_h\|^2_{\mathcal{W}} + \sum_{\Gamma_{ij}}\|\tilde{v}^H_{ij}\|^2_A.\end{aligned}$$

Substituting (5.6), (5.13) and (5.14) into the above inequality yields

$$\sum_{k=1}^{N} \omega_k |\tilde{v}_k^0|_{1,\Omega_k}^2 \lesssim \log(1/d)\log^2(d/h)\|v_h\|_A^2.$$

Combining this with (5.8) and (5.14) leads to (5.15).

Now the inequality (5.10) follows by (5.11), (5.13) and (5.15) (note the assumptions (4.2) and (4.3)). □

Proof of Theorem 4.3. Let $v \in V_d^{\perp}(\Omega)$. Since $P_d(I-T)v \in V_d(\Omega)$, we have

$$(MAv, Av) = ([T + P_d(I-T)]v, Av) = \mathcal{A}(Tv, v). \tag{5.16}$$

It is easy to see, by (4.3), that

$$\mathcal{A}(Tv, v) \lesssim \mathcal{A}(v, v), \ v \in V_d^{\perp}(\Omega).$$

This, together with (5.16), leads to

$$\lambda_{\max}^{\perp}(MA) \lesssim 1. \tag{5.17}$$

In the following, we prove that

$$\lambda_{m_0+1}^{\perp}(MA) \gtrsim \frac{1}{\log(1/d)\log^2(d/h)}. \tag{5.18}$$

By (5.16), one needs only to estimate the minimal eigenvalue of the operator $T|_{\tilde{V}_d^{\perp}(\Omega)}$, namely, to estimate the maximal eigenvalue of the operator $E|_{\tilde{V}_d^{\perp}(\Omega)}$ with

$$E = (I - T_{\mathcal{W}})(I - T_{\Gamma})(I - T_{\mathcal{W}}).$$

By the definition of $T_{\mathcal{W}}$, we know that

$$(T_{\mathcal{W}}v, v)_A \le \theta_0 \|v\|_A^2, \ \forall v \in V_h(\Omega)$$

for a constant $\theta_0 \in (0,\ 2)$. Then we deduce, by the definition of E and the direct calculation, that (compare [28])

$$\begin{aligned}
\|v\|_A^2 - (Ev, v)_A &= \|v\|_A^2 - ((I - T_{\mathcal{W}})v, (I - T_{\mathcal{W}})v)_A + (T_{\Gamma}(I - T_{\mathcal{W}})v, (I - T_{\mathcal{W}})v)_A \\
&\ge (2 - \theta_0)(T_{\mathcal{W}}v,\ v)_A + (T_{\Gamma}(I - T_{\mathcal{W}})v, (I - T_{\mathcal{W}})v)_A \\
&\ge \min\{1,\ 2 - \theta_0\}[(T_{\mathcal{W}}v,\ v)_A + (T_{\Gamma}(I - T_{\mathcal{W}})v, (I - T_{\mathcal{W}})v)_A].
\end{aligned}$$

This, together with (4.9) of [28], yields (note that $K_1 = 1$)

$$\|v\|_A^2 - (Ev, v)_A \ge \frac{\min\{1,\ 2 - \theta_0\}}{4}((T_{\Gamma} + T_{\mathcal{W}})v, v)_A, \ v \in \tilde{V}_d^{\perp}(\Omega). \tag{5.19}$$

On the other hand, by the proof of Theorem 4.1, there is a decomposition for any $v \in \tilde{V}_d^{\perp}(\Omega)$,

$$v = v_d + v_{\mathcal{W}} + \sum_{\Gamma_{ij}} v_{ij}, \text{ with } v_d \in V_d(\Omega),\ v_{\mathcal{W}} \in V_h^{\mathcal{W}}(\Omega) \text{ and } v_{ij} \in V_h^0(\Omega_{ij}),$$

such that

$$\|v_{\mathcal{W}}\|_{\mathcal{W}}^2 + \sum_{\Gamma_{ij}} (B_{ij}v_{ij}, v_{ij}) \lesssim \log(1/d)\log^2(d/h)|v|_A^2. \tag{5.20}$$

Since $v \in \tilde{V}_d^{\perp}(\Omega)$, we have $(v, v_d)_A = 0$. Then

$$\|v\|_A^2 = (v, v_d)_A + (v, v_{\mathcal{W}})_A + \sum_{\Gamma_{ij}} (v, v_{ij})_A = (v, v_{\mathcal{W}})_A + \sum_{\Gamma_{ij}} (v, v_{ij})_A.$$

As in Theorem 4.1 of [28], we further prove by (5.20) that

$$\|v\|_A^2 \lesssim \log(1/d)\log^2(d/h)((T_\Gamma + T_{\mathcal{W}})v, v)_A,\ v \in \tilde{V}_d^\perp(\Omega).$$

Plugging this into (5.19) leads to

$$\|v\|_A^2 - (Ev, v)_A \gtrsim \frac{1}{\log(1/d)\log^2(d/h)}\|v\|_A^2,\ v \in \tilde{V}_d^\perp(\Omega).$$

Hence

$$(Ev, v)_A \leq \left(1 - \frac{1}{C_0 \log(1/d)\log^2(d/h)}\right)\|v\|_A^2,\ v \in \tilde{V}_d^\perp(\Omega),$$

where C_0 is a constant independent of h, d and possible jumps of the coefficient ω. Thus, we have by the definition of T,

$$(Tv, v)_A = \|v\|_A^2 - (Ev, v)_A \geq \frac{1}{C_0 \log(1/d)\log^2(d/h)}\|v\|_A^2,\ v \in \tilde{V}_d^\perp(\Omega).$$

Substituting the above inequality into (5.16) gives (5.18). Similarly, we can show

$$\lambda_{\min}^\perp(MA) \geq \frac{h}{d\log(1/d)\log^2(d/h)}. \qquad \square$$

6. Numerical experiments

In this section, we present some numerical results to demonstrate the theoretical results in Section 4. We consider the model problem (2.1) with Ω being the unit cube. Let the function $f(x)$ be defined by the first equation (2.1) with $\omega(x) = 1$ and $u(x, y, z) = \sin \pi x \cdot \sin \pi y \cdot \sin \pi z$. The coefficient $\omega(x)$ will be given below.

Let Ω be decomposed into $n \times n \times n$ hexahedra with the size $d = 1/n$. All the hexahedra constitute the desired domain decomposition. To get the final triangulation of Ω, we decompose each hexahedron mentioned above into $m \times m \times m$ smaller hexahedra with the size $h = d/m = 1/nm$. All the smaller hexahedra constitute the desired triangulation. It is clear that each hexahedral subdomain is just the union of some smaller hexahedral elements.

The standard Q_1 finite element space is used for the discretization of (2.2). The resulting system (3.1) is solved by the PCG method with the preconditioners B and M defined in Section 4. We will report iteration counts, condition numbers and reduced condition numbers. Here, the iteration terminates when the relative remainder is not greater than $1.0D - 6$.

Let $\kappa_r(BA)$ (resp. $\kappa_r^\perp(MA)$) denote the reduced condition number of BA (resp. MA) when removing $r-1$ small eigenvalues of BA (resp. $(MA)|_{V_d^\perp(\Omega)}$). In particular, $\kappa_2(BA)$ (resp. $\kappa_2^\perp(MA)$) denotes the reduced condition number of BA (resp. $(MA)|_{V_d^\perp(\Omega)}$) when removing the minimal eigenvalue of BA (resp. $(MA)|_{V_d^\perp(\Omega)}$). For convenience, we use "it." to denote the iteration counts.

Case (i): the coefficient $\omega(x) = 1$, which has no jump.

The numerical results are listed in Tales 6.1 and 6.2.

These results indicate that the condition numbers of BA and MA are nearly optimal when the coefficient has no jump.

Case (ii): the coefficient $\omega(x)$ has large jumps (refer to [29]):

$$I_F^0 v(p) \begin{cases} 10^5, & \text{in } D, \\ 1, & \text{in } \Omega\backslash D, \end{cases}$$

where D is a cube or the union of a few cubes.

TABLE 6.1. Iteration counts, condition numbers and reduced condition numbers for B.

	$d = 1/4$			$d = 1/5$			$d = 1/6$		
d/h	it.	$\kappa(BA)$	$\kappa_2(BA)$	it.	$\kappa(BA)$	$\kappa_2(BA)$	it.	$\kappa(BA)$	$\kappa_2(BA)$
8	30	26.79	22.45	29	26.94	22.08	29	27.55	22.70
16	33	36.77	34.93	35	37.46	35.41	35	38.47	35.79

TABLE 6.2. Iteration counts, condition numbers and reduced condition numbers for M.

	$d = 1/4$			$d = 1/5$			$d = 1/6$		
d/h	it.	$\kappa^{\perp}$(MA)	$\kappa_2^{\perp}$(MA)	it.	$\kappa^{\perp}$(MA)	$\kappa_2^{\perp}$(MA)	it.	$\kappa^{\perp}$(MA)	$\kappa_2^{\perp}$(MA)
8	22	15.28	12.80	23	15.26	13.05	23	15.95	13.99
16	26	21.88	19.16	27	21.81	18.97	27	22.42	18.97

We first consider an example without cross-point, with D defined by

$$D = [\frac{1}{4},\ \frac{1}{2}]^3.$$

The numerical results are given in Tables 6.3 and 6.4.

TABLE 6.3. Iteration counts, condition numbers and reduced condition numbers for B.

	$d = 1/4$			$d = 1/8$		
d/h	it.	$\kappa(BA)$	$\kappa_2(BA)$	it.	$\kappa(BA)$	$\kappa_2(BA)$
8	38	47.05	35.23	36	38.26	34.35
16	44	64.02	49.54	41	52.80	48.29

TABLE 6.4. Iteration counts, condition numbers and reduced condition numbers for M.

	$d = 1/4$			$d = 1/8$		
d/h	it.	$\kappa^{\perp}(MA)$	$\kappa_2^{\perp}(MA)$	it.	$\kappa^{\perp}(MA)$	$\kappa_2^{\perp}(MA)$
8	30	25.24	19.55	27	20.65	19.14
16	35	34.75	28.16	32	29.66	27.54

Tables 6.3 and 6.4 tell us that the condition numbers of BA and MA are also nearly optimal when there is no cross-point in the distribution of the jump of the coefficient.

Then we consider an example with cross-points:

$$D = [0,\ \frac{1}{4}]^3 \bigcup [\frac{1}{4},\ \frac{1}{2}]^3 \bigcup [\frac{1}{2},\ \frac{3}{4}]^3 \bigcup [\frac{3}{4}, 1].$$

In order to illustrate our theoretical results more clearly, we would like to calculate several reduced condition numbers. The numerical results are listed by Tables 6.5 and 6.6.

The results given in these two tables tell us that the PCG methods with the preconditioners B and M possess nearly optimal convergence rates, although the

TABLE 6.5. Iteration counts, condition numbers and reduced condition numbers for B.

d	d/h	it.	$\kappa(BA)$	$\kappa_2(BA)$	$\kappa_3(BA)$	$\kappa_4(BA)$
1/4	8	43	349.06	37.01	32.42	26.92
1/4	16	51	940.82	51.70	46.45	37.96
1/8	8	46	342.88	35.39	31.25	26.98
1/8	16	56	921.03	49.49	43.63	39

TABLE 6.6. Iteration counts, condition numbers and reduced condition numbers for M.

d	d/h	it.	$\kappa^{\perp}(MA)$	$\kappa_2^{\perp}(MA)$	$\kappa_3^{\perp}(MA)$	$\kappa_4^{\perp}(MA)$
1/4	8	33	156.84	20.59	17.77	14.87
1/4	16	41	473.08	29.53	26.07	21.41
1/8	8	35	155.56	19.77	17.78	15.30
1/8	16	44	467.23	28.34	25.63	22.47

condition numbers of the preconditioned systems themselves are not nearly optimal yet. The main reason is that the preconditioned system has only a few "bad" small eigenvalues, and so a reduced condition number is still nearly optimal.

All the above numerical results confirm our theoretical results. In particular, Algorithm 4.2 has faster convergence than Algorithm 4.1 in every case. Now, we see what will happen if we replace the *wire-basket* subspace $V_h^{\mathcal{W}}(\Omega)$ by the smaller subspace $\hat{V}_h^{\mathcal{W}}(\Omega)$. We consider the case without jump of the coefficient: the coefficient $\omega(x) = 1$. The numerical results are given in Table 6.7.

TABLE 6.7. Iteration counts, condition numbers and reduced condition numbers with $\hat{V}_h^{\mathcal{W}}(\Omega)$ instead of $V_h^{\mathcal{W}}(\Omega)$.

	$d = 1/4$			$d = 1/5$			$d = 1/6$		
d/h	it.	$\kappa(\hat{B}A)$	$\kappa_2(\hat{B}A)$	it.	$\kappa(\hat{B}A)$	$\kappa_2(\hat{B}A)$	it.	$\kappa(\hat{B}A)$	$\kappa_2(\hat{B}A)$
8	47	374.47	314.20	51	391.94	342.34	53	402.22	355.36
16	60	1032.13	857.61	62	1081.40	940.12	65	1110.81	982.58

These results illustrate that the subspace $V_h^{\mathcal{W}}(\Omega)$ cannot be replaced by the smaller one $\hat{V}_h^{\mathcal{W}}(\Omega)$ (see Subsection 4.4).

7. CONCLUSION

In this paper, we have constructed two substructuring preconditioners with the simplest coarse solver and inexact subdomain solvers for solving the second order three-dimensional elliptic equations with large jump coefficients. We have shown that the PCG method with such preconditioners has a nearly optimal convergence rate, although the condition numbers of the preconditioned systems are not quasi-optimal yet. The method in this paper can be extended to some other equations (for example, Maxwell's equations).

Acknowledgment

The authors wish to thank Professor Jinchao Xu for suggesting the topic of the work and providing many insightful comments on the article.

References

1. Y. Achdou, Y. Maday, and O. Widlund, *Iterative substructuring preconditioners for mortar element methods in two dimensions*. SIAM J. Numer. Anal., **36**(1999), 551-580. MR1675257 (99m:65233)
2. Owe Axelsson, *Iteration number for the conjugate gradient method*, Math. Comput. Simulation, **61**(2003), No. 3-6, 421-435. MR1984142 (2004g:65031)
3. C. Borgers, *The Neumann-Dirichlet domain decomposition method with inexact solvers on the subdomains*, Numer. Math., **55**(1989), 123-136. MR987381 (90f:65191)
4. J. Bramble, J. Pasciak and A. Schatz, *The construction of preconditioners for elliptic problems by substructuring*, I. Math. Comp., **47**(1986), 103-134. MR842125 (87m:65174)
5. J. Bramble, J. Pasciak and A. Schatz, *The construction of preconditioners for elliptic problems by substructuring*, IV. Math. Comp., **53**(1989), 1-24. MR970699 (89m:65098)
6. J. Bramble, J. Pasciak and A. Vassilev, *Analysis of nonoverlapping domain decomposition algorithms with inexact solvers*, Math. Comput., **67**(1998), 1-19. MR1432125 (98k:65065)
7. J. Bramble and J. Xu, *Some estimates for a weighted L^2 projection*, Math. Comp., **56**(1991), 463-476 MR1066830 (91k:65140)
8. M. Dryja, F. Smith and O. Widlund, *Schwarz analysis of iterative substructuring algorithms for elliptic problems in three dimensions*, SIAM J. Numer. Anal., **31**(1994), no. 6, 1662-1694. MR1302680 (95m:65211)
9. M. Dryja and O. Widlund, *Schwarz methods of Neumann-Neumann type for three-dimensional elliptic finite element problems*, Comm. Pure Appl. Math., **48**(1995), 121-155. MR1319698 (96d:65199)
10. M. Dryja and O. Widlund, *Towards a unified theory of domain decomposition algorithms for elliptic problems*. Third International Symposium on Domain Decomposition Methods for Partial Differential Equations (Houston, TX, 1989), 3–21, SIAM, Philadelphia, PA, 1990. MR1064335 (91m:65294)
11. C. Farhat, J. Mandel and F. Roux, *Optimal convergence properties of the FETI Domain Decomposition Method*. Comput. Methods. Appl. Mech. Engrg., **115** (1994), 367-388. MR1285024 (95d:65091)
12. G. Haase, U. Langer and A. Meyer, *The approximate Dirichlet domain decomposition method. Part II: applications to second order B.V.P.s*, Computing, **47**(1991), 153-167. MR1139434 (93e:65146b)
13. Qiya Hu and Guoping Liang, *Acceleration of the non-symmetrized, two-level iteration*, Appl. Numer. Math., **41**(2002), no. 2, 305–323. MR1897012 (2003e:65042)
14. Q. Hu, Z. Shi and D. Yu, *Efficient solvers for saddle-point problems arising from domain decompositions with Lagrange multipliers*, SIAM J. Numer. Anal., **42**(2004), no. 3, 905-933. MR2112786 (2005h:65045)
15. Q. Hu and J. Zou, *A nonoverlapping domain decomposition method for Maxwell's equations in three dimensions*, SIAM J. Numer. Anal., **41**(2003), 1682-1708. MR2035002 (2005a:65131)
16. Q. Hu and J. Zou, *Substructuring preconditioners for saddle-point problems arising from Maxwell's equations in three dimensions*. Math. Comput., **73**(2004), no. 245, 35-61. MR2034110 (2004m:65197)
17. A. Klawonn, O. Widlund and M. Dryja, *Dual-Primal FETI methods for three-dimensional elliptic problems with Heterogeneous coefficients*. SIAM J. Numer. Anal., **40**(2002), 159-179. MR1921914 (2003g:65044)
18. Y. Kuznetsov, *Efficient iterative solvers for elliptic finite element problems on non-matching grids*. Russian J. Numer. Anal. and Math. Modeling, **10**(1995), 3, 187-211. MR1343473 (96j:65125)
19. U. Langer and O. Steinbach, *Boundary element tearing and interconnecting methods*. Computing, **71**(2003), 205-228. MR2023942 (2004k:65239)
20. J. Mandel and M. Brezina, *Balancing domain decomposition for problems with large jumps in coefficients*. Math. Comp., **65**(1996), 1387-1401. MR1351204 (97a:65109)

21. J. Mandel and R. Tezaur, *On the convergence of a dual-primal substructuring method*. Numer. Math., **88**(2001), 543-558. MR1835470 (2002b:65161)
22. B. Smith, P. Bjorstad and W. Gropp, *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge University Press, 1996. MR1410757 (98g:65003)
23. P. Tallec, *Domain decomposition methods in computational mechanics*, Comput. Mech. Adv., 2:1321-220, 1994. MR1263805 (95b:65147)
24. C. Tong, T. Chan and C. Kuo, *A domain decomposition preconditioner based on a change to a multilevel nodal basis*. SIAM J. Sci. Stat. Comput., **12**(1991), 1486-1495. MR1129659 (92i:65070)
25. A. Toselli, *Dual-primal FETI algorithms for edge element approximations: three-dimensional h finite elements on shape-regular meshes*, to appear in IMA J. Numer. Anal.
26. A. Toselli and O. Widlund, *Domain Decomposition Methods—Algorithms and Theory*, Springer, 2004. MR2104179 (2005g:65006)
27. A. Toselli, O. Widlund and B. Wohlmuth, *An iterative substructuring method for Maxwell's equations in two dimensions*. Math. Comp., 70(2001), 935-949. MR1710632 (2001j:65140)
28. J. Xu, *Iterative methods by space decomposition and subspace correction*, SIAM Review, **34**(1992), 581-613. MR1193013 (93k:65029)
29. J. Xu and Y. Zhu, *Uniform convergent multigrid methods for elliptic problems with strongly discontinuous coefficients*, M^3AS, **18**(2008), 77-105. MR2378084 (2008k:65271)
30. J. Xu and J. Zou, *Some non-overlapping domain decomposition methods*, SIAM Review, **24**(1998)

LSEC, Institute of Computational Mathematics and Scientific Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100080, People's Republic of China

E-mail address: hqy@lsec.cc.ac.cn

Department of Mathematics, Xiangtan University, Hunan 411105, People's Republic of China

E-mail address: shushi@xtu.edu.cn

Department of Mathematics, Xiangtan University, Hunan 411105, People's Republic of China

E-mail address: xianxian.student@sina.com