ROBUST BDDC ALGORITHMS FOR FINITE VOLUME ELEMENT METHODS
YANRU SU†, XUEMIN TU†, AND YINGXIANG XU‡

Abstract. The balancing domain decomposition by constraints (BDDC) method is applied to the linear system arising from the finite volume element method (FVEM) discretization of a scalar elliptic equation. The FVEMs share nice features of both finite element and finite volume methods and are flexible for complicated geometries with good conservation properties. However, the resulting linear system usually is asymmetric. The generalized minimal residual (GMRES) method is used to accelerate convergence. The proposed BDDC methods allow for jumps of the coefficient across subdomain interfaces. When jumps of the coefficient appear inside subdomains, the BDDC algorithms adaptively choose the primal variables deriving from the eigenvectors of some local generalized eigenvalue problems. The adaptive BDDC algorithms with advanced deluxe scaling can ensure good performance with highly discontinuous coefficients. A convergence analysis of the BDDC method with a preconditioned GMRES iteration is provided, and several numerical experiments confirm the theoretical estimate.

Key words. finite volume element methods, domain decomposition, BDDC, deluxe scaling

AMS subject classifications. 65F10, 65N30, 65N55

1. Introduction. Finite volume methods are widely used in different areas of science and engineering where local conservation is an important property to be ensured for the discretizations. One special class of finite volume methods is called finite volume element methods (FVEMs for short); see for example [3, 9, 10, 32]. The FVEMs use two types of meshes. The approximation space of the exact solution is constructed based on a primal mesh. A dual mesh is used to construct the test function space. The FVEMs share nice features of both finite element and finite volume methods and are flexible for complicated geometries with good conservation properties. Recently, there appeared many works on stability, superconvergence, and high-order methods for FVEMs; see [11, 12, 13, 49, 50, 51, 56] and the references therein.

However, there are only a few works on fast solvers for the linear systems resulting from the FVEMs. This might be partially due to the fact that the resulting linear systems are usually asymmetric. In [48] the convergence rate of the generalized minimal residual (GMRES) method for solving linear systems from FVEMs was analyzed, where simple diagonal scaling is used to improve the convergence rate. Some wavelet and multilevel preconditioners are studied and analyzed in [31]. One family of widely used preconditioner techniques are the domain decomposition methods [39], which have provided efficient preconditioners for large linear systems arising from finite element discretizations (FEMs) for many partial differential equations (PDEs). Both overlapping and nonoverlapping domain decomposition methods have been applied to solve asymmetric linear systems from finite element and discontinuous Garlekin discretizations [1, 2, 6, 7, 8, 15, 38, 43, 47]. In these works, the asymmetry is due to the original PDEs, and usually they require the subdomain size to be small enough to ensure the convergence rate of the preconditioned GMRES method being independent of the number of subdomains. There are also some domain decomposition methods proposed for solving the linear systems of FVEMs. Overlapping domain decompositions are studied in [14, 55], and the convergence rate of the preconditioned GMRES method is proved to be independent of the number of subdomains under the assumption that the mesh size is small enough. Using the same assumption, iterative substructuring (nonoverlapping) domain decomposition methods
are studied in [34] using an additive Schwarz framework, wherein the convergence is proved to be independent of the number of subdomains, depends poly-logarithmically on the subdomain problem sizes, and is robust with respect to jumps in the coefficient across the subdomain interface as well.

One of the most popular nonoverlapping domain decomposition methods is the balancing domain decomposition by constraints (BDDC) method, which was introduced in [16] for symmetric positive definite problems and has been used for solving linear systems from different applications [28, 29, 40, 41, 42, 43, 44, 45, 45, 46]. See [53] for a recent review. Several methods, for example in [28, 43, 47], are designed and analyzed for asymmetric or indefinite problems, where the convergence is independent of the number of subdomains under the assumption that the subdomain size is as small as that for the overlapping domain decomposition methods. In this paper, we propose and analyze BDDC algorithms for the linear system from FVEMs. Different from previous works on BDDC methods for asymmetric linear systems, the elliptic PDE considered here is self-adjoint. The asymmetry of the linear system is due to the FVEM discretizations. Therefore, in our analysis we only require that the mesh size is small enough to ensure convergence independent of the number of subdomains. Different from the additive Schwarz approach used in [34], we will combine the estimate of an average operator [43, 47] and the connection between the linear systems from the FEMs and FVEMs for our analysis. In this paper, in addition to jumps of the coefficient across subdomain interfaces as in [34], we will also consider jumps of the coefficient inside subdomains. For the latter, we will use the deluxe scaling [20], and the primal variables in the BDDC algorithms are derived from the eigenvectors of some carefully chosen local generalized eigenvalue problems [24, 35, 36, 54]. To the best of our knowledge, this is the first adaptive BDDC algorithm applied to asymmetric problems.

The rest of the paper is organized as follows. We first describe the finite volume element discretization in Section 2. In Section 3, our domain decomposition and the BDDC preconditioner are introduced. In Section 4, different choices of the scaling and primal constraints are discussed. We provide an analysis of the convergence rate of our BDDC algorithms in Section 5. Finally, some computational results are presented in Section 6 to illustrate our theoretical results.

2. Problem setting and a finite volume element discretization. We consider the following second-order scalar elliptic problem in a bounded polyhedral domain \( \Omega \subset \mathbb{R}^2 \),

\[
\begin{aligned}
- \nabla \cdot (G \nabla u) &= f, & \text{in } \Omega, \\
u &= 0, & \text{on } \partial \Omega,
\end{aligned}
\]

where \( G = (g_{ij}) \in (L^\infty(\Omega))^{2 \times 2} \) is the diffusion coefficient matrix and \( f \in L^2(\Omega) \). We assume that \( G \) is a real symmetric positive definite matrix satisfying

\[
\exists \alpha_u, \alpha_l > 0 \quad \text{such that} \quad \alpha_u \xi^T \xi \geq \xi^T G(x) \xi \geq \alpha_l \xi^T \xi, \quad \forall x \in \Omega \text{ and } \forall \xi \in \mathbb{R}^2.
\]

By the assumptions above, there exists a solution \( u \in H^1_0(\Omega) \) of (2.1) such that

\[
(a_E(u, v) := \int_\Omega (G \nabla u) \cdot \nabla v dx = \int_\Omega f v dx, \quad \forall v \in H^1_0(\Omega).
\]

Let \( \hat{W} \subset H^1_0(\Omega) \) be the standard continuous, piecewise linear finite element function space on a shape-regular triangulation \( T_h \) of \( \Omega \). We denote the set of nodes in \( T_h \) by \( \mathcal{N} \) and an element of the triangulation by \( K \). \( h_K \) is the diameter of \( K \). We set \( h = \max_K h_K \). Similar to [14], we assume that no interior angle of any triangle in \( T_h \) is larger than \( \frac{\pi}{3} \) to avoid
unnecessary complexity. In order to describe the finite volume element methods, we call $T_h$ a primal mesh and introduce a dual partition of $T_h$. For each vertex $P \in \mathcal{N}$, we construct a (barycenter/Donald) dual element $K_P^*$ as follows: for each $K \in T_h$ sharing $P$, we choose a barycenter of $K$ and connect it with the midpoints of the edges of $K$. In Figure 2.1, $P_1$, $P_2$, and $P_3$ are three vertices of the element $K$. The barycenter of $K$ is denoted as $c_K$. $m_{12}$, $m_{23}$, and $m_{31}$ are the midpoints of the edges $P_1P_2$, $P_2P_3$, and $P_3P_1$, respectively. The dual element $K_P^*$ is the polygon with the dash-dotted line. The dual elements form a dual mesh $T_h^*$ of $\Omega$. Let $\hat{W}^*$ be the space of piecewise constant functions (a constant on each dual element $K_P^*$) over the dual mesh $T_h^*$. For any $P \in \partial \Omega$, the dual basis function is zero.

The finite element discretization problem is to find $u_h \in \hat{W}$ such that

\begin{equation}
(2.3) \quad a_E(u_h, v_h) = \int_{\Omega} f v_h \, dx, \quad \forall v_h \in \hat{W},
\end{equation}

where the bilinear form $a_E$ is defined in (2.2). Similarly, the finite volume element discretization problem is to find $u_h \in \hat{W}$ such that

\begin{equation}
(2.4) \quad - \sum_{K_P^* \in T_h^*} \int_{\partial K_P^*} (G \nabla u_h) \cdot \mathbf{n} \, v_h \, ds = \sum_{K_P^* \in T_h^*} \int_{K_P^*} f v_h \, dx, \quad \forall v_h \in \hat{W}^*.
\end{equation}

The convergence of this finite volume element method is of first order in the $H^1$-norm and of second order in the $L^2$-norm; see [3, 9, 10]. A review of the finite volume element method is given in [32].

We introduce a mapping $\Pi_h$ from $\hat{W} \to \hat{W}^*$ as in [14, 48]. Given $v \in \hat{W}$. Let $\Pi_h v \in \hat{W}^*$ be defined as follows: for each vertex $P \in \mathcal{N}$,

$$
\Pi_h v(x) = v(P), \quad \forall x \in K_P^*.
$$

Using this mapping $\Pi_h$, we can reformulate (2.4) as

\begin{equation}
(2.5) \quad a_h(u_h, v_h) = \sum_{K_P^* \in T_h^*} \int_{K_P^*} f \Pi_h v_h \, dx, \quad \forall v_h \in \hat{W},
\end{equation}

Fig. 2.1. The primal and dual mesh of a linear FVEM on element $K$ and Vertex $P_3$. 
where

\[
a_h(u_h, v_h) = - \sum_{K^*_p \in T_h} \int_{\partial K^*_p} \left( \nabla u_h \right) \cdot n v_h(P) \, ds = - \sum_{K^*_p \in T_h} \int_{\partial K^*_p} \left( G \nabla u_h \right) \cdot n \Pi_h v_h \, ds.
\]

The system of linear equations corresponding to the finite element and finite volume element problems (2.3) and (2.5) are denoted by

\[
(2.6) \quad A_E u_E = f_E
\]

and

\[
(2.7) \quad A u = f,
\]

respectively. Here the coefficient matrix \( A_E \) is symmetric positive definite, but \( A \) is usually asymmetric even though the original problem (2.1) is symmetric. In the next section, we will introduce a BDDC algorithm to solve the system (2.7). For the BDDC algorithms with simple scaling, we do not need to form the system (2.6), which is only used for our analysis of the BDDC algorithm. However, for the deluxe scaling and the adaptive BDDC algorithms, we do need to form the system (2.6). The details are provided in Section 4.

3. Domain decomposition and a BDDC preconditioner. We decompose the original computational domain \( \Omega \) into \( N \) nonoverlapping polyhedral subdomains \( \Omega_i \) based on the primal mesh. We assume that each subdomain is a union of shape-regular elements. Let \( H \) be the typical diameter of the subdomains and \( \Gamma = (\cup \partial \Omega_i) \setminus \partial \Omega \) the subdomain interface shared by neighboring subdomains. We define \( \Gamma_i = \partial \Omega_i \cap \Gamma \), the interface of the subdomain \( \Omega_i \). We note that our algorithm is also defined for the subdomain partition obtained from mesh partitioners, where less regular subdomains will be obtained. For the analysis with irregular subdomains in domain decomposition methods; see [17, 18, 19, 25, 52].

We first reduce the global system (2.7) to a subdomain interface problem on \( \Gamma \). In order to do that, we decompose the space \( \widehat{W} \) as follows:

\[
\widehat{W} = W_I \oplus \widehat{W}_\Gamma = \left( \Pi_{i=1}^N W_I^{(i)} \right) \oplus \widehat{W}_\Gamma,
\]

where \( W_I^{(i)} \) are the spaces of the subdomain interior variables, while \( \widehat{W}_\Gamma \) is the subspace corresponding to the variables on the interface. We can rewrite the original problem (2.7) in the following way: find \( u_I \in W_I \) and \( u_\Gamma \in \widehat{W}_\Gamma \), such that

\[
(3.1) \quad \begin{bmatrix} A_{II} & A_{I\Gamma} \\ A_{\Gamma I} & A_{\Gamma\Gamma} \end{bmatrix} \begin{bmatrix} u_I \\ u_\Gamma \end{bmatrix} = \begin{bmatrix} f_I \\ f_\Gamma \end{bmatrix}.
\]

Here \( A_{II} \) is block diagonal with one block corresponding to one subdomain. \( A_{\Gamma\Gamma} \) is assembled from subdomain matrices across the subdomain interfaces. Due to the structure of \( A_{II} \), we can eliminate the subdomain interior variables \( u_I \) in each subdomain independently from (3.1) and reduce (2.7) to a subdomain interface problem

\[
(3.2) \quad S_\Gamma u_\Gamma = g_\Gamma,
\]

where \( S_\Gamma = A_{\Gamma\Gamma} - A_{\Gamma I} A_{II}^{-1} A_{I\Gamma} \) and \( g_\Gamma = f_\Gamma - A_{\Gamma I} A_{II}^{-1} f_I \). After solving (3.2), we can solve for \( u_I \) in each subdomain independently given the value \( u_\Gamma \).
We will discuss our choices of the primal variables $u$. We define $\tilde{u}$ where $\tilde{u} = s u$. We can obtain $\tilde{u}$ from $u$ by a vector, we need to use the GMRES method to solve (3.5). In each iteration, to multiply $A$ from the FVEMs in (2.7) is asymmetric, $S_\Gamma$ is asymmetric as well. We can obtain $S_\Gamma$ by partially assembling the subdomain local Schur complements $S^{(i)}_\Gamma$ defined in (3.3) with respect to the primal interface variables. We define the injection operator $\Pi$ defined as follows: given $u^{(i)}_\Gamma \in W^{(i)}_\Gamma$, $S^{(i)}_\Gamma u^{(i)}_\Gamma$ is defined as

\begin{equation}
\begin{bmatrix}
A^{(i)}_{III} & A^{(i)}_{I} \\
A^{(i)}_{II} & A^{(i)}_{II}
\end{bmatrix}
\begin{bmatrix}
u^{(i)}_\Gamma \\
u^{(i)}_\Gamma
\end{bmatrix} =
\begin{bmatrix}
0 \\
S^{(i)}_\Gamma u^{(i)}_\Gamma
\end{bmatrix}
\end{equation}

We then introduce a BDDC preconditioner to solve (3.2). We further decompose $\tilde{W}_\Gamma$ into the primal interface variables and introduce a partially assembled interface space as $\tilde{W} = W_\Pi \oplus W_\Delta$. We relax the continuity for the dual variables and introduce a partially assembled interface space as $\tilde{W}_\Gamma = \tilde{W}_\Pi \oplus W_\Delta = \tilde{W}_\Pi \oplus \left( \Pi_{i=1}^N W^{(i)}_\Delta \right)$.

Here the degrees of freedom in $W_\Delta$ may be discontinuous across the subdomain interfaces. The subspace $\tilde{W}_\Pi$ contains the coarse level with continuous primal interface degrees of freedom. Correspondingly, we define a partially sub-assembled problem matrix $\tilde{A}$ as the two by two block form

$$
\tilde{A} = \begin{bmatrix}
A_{II} & \tilde{A}_{II} \\
\tilde{A}_{II} & \tilde{A}_{II}
\end{bmatrix}
$$

where

$$
\tilde{A}_{II} = [A_{I\Delta} & A_{III}] , \quad \tilde{A}_{II} = [A_{\Delta I} & A_{III}], \quad \tilde{A}_{II} = [A_{\Delta I} & A_{III}].
$$

We note that $\tilde{A}_{II}$ is assembled only for the coarse-level primal degrees of freedom across the interface. We define the partially sub-assembled Schur complement operator $\tilde{S}_\Gamma$ as $\tilde{S}_\Gamma = \tilde{A}_{II} - \tilde{A}_{II} A_{II}^{-1} \tilde{A}_{II}$. We can obtain $\tilde{S}_\Gamma$ by partially assembling the subdomain local Schur complements $S^{(i)}_\Gamma$ defined in (3.3) with respect to the primal interface variables. We define the injection operator $\tilde{R}_\Gamma$ that maps the element in $\tilde{W}_\Gamma$ to $\tilde{W}_\Gamma$. By the definition of $S_\Gamma$ and $\tilde{S}_\Gamma$, we can obtain $S_\Gamma$ from $\tilde{S}_\Gamma$ by further assembling with respect to the dual interface variables, i.e.,

$$
S_\Gamma = \tilde{R}_\Gamma^T \tilde{S}_\Gamma \tilde{R}_\Gamma.
$$

We define $\tilde{R}_{D,\Gamma} = D \tilde{R}_\Gamma$, where $D$ is a scaling matrix. Different choices of the scaling matrix $D$ can be found in [53, 54], and it should provide a partition of unity:

\begin{equation}
\tilde{R}_{D,\Gamma}^T \tilde{R}_\Gamma = \tilde{R}_\Gamma^T \tilde{R}_{D,\Gamma} = I.
\end{equation}

We will discuss our choices of the primal variables $u_\Pi$ and $D$ in Section 4 in detail.

The BDDC preconditioned interface problem is given as

\begin{equation}
\tilde{R}_{D,\Gamma}^T \tilde{R}_\Gamma S_\Gamma u_\Gamma = \tilde{R}_{D,\Gamma}^T \tilde{S}^{-1}_\Gamma \tilde{R}_{D,\Gamma} g_\Gamma.
\end{equation}

Since the matrix $A$ from the FVEMs in (2.7) is asymmetric, $S_\Gamma$ is asymmetric as well. We need to use the GMRES method to solve (3.5). In each iteration, to multiply $S_\Gamma$ by a vector, we
need to solve subdomain Dirichlet boundary problems, while subdomain Neumann boundary value problems and a coarse-level problem need to be solved for multiplying \( S^{-1}_\Gamma \) by a vector; see [30] for more details. We will analyze the convergence of the BDDC preconditioned GMRES algorithm in Section 5.

We introduce similar Schur complement matrices for the finite element matrix \( A_E \). We rewrite the system (2.6) as

\[
\begin{bmatrix}
A_{E,II} & A_{E,IT} \\
A_{E,IT} & A_{E,GG}
\end{bmatrix}
\begin{bmatrix}
u_{E,I} \\
u_{E,G}
\end{bmatrix} =
\begin{bmatrix}
f_{E,I} \\
f_{E,G}
\end{bmatrix}.
\]

We define the subdomain local Schur complement \( S_{E,\Gamma}^{(i)} \) as \( S_{E,\Gamma}^{(i)} \) given in (3.3) for \( A_E \),

\[
S_{E,\Gamma}^{(i)} = A_{E,\Gamma}^{(i)} - A_{E,II}^{(i)} A_{E,II}^{-1} A_{E,\Gamma}^{(i)}.
\]

The global subdomain interface Schur complement \( S_{E,\Gamma} \) and the partial assembled Schur complement \( \tilde{S}_{E,\Gamma} \) for the finite element discretization are defined as

\[
S_{E,\Gamma} = A_{E,\Gamma} - A_{E,II} A_{E,II}^{-1} A_{E,\Gamma},
\]

\[
\tilde{S}_{E,\Gamma} = \tilde{A}_{E,\Gamma} - \tilde{A}_{E,II} \tilde{A}_{E,II}^{-1} \tilde{A}_{E,\Gamma}.
\]

Thus it holds

\[
S_{E,\Gamma} = \tilde{R}_I^T \tilde{S}_{E,\Gamma} \tilde{R}_I.
\]

4. Different choices of the scaling matrices and primal subspaces. The choices of the scaling matrix \( D \) and the primal variables \( u_e \) play crucial roles for the performance of the BDDC algorithms. Detailed discussions about the different choices for the symmetric positive definite problems are given in [53, Sections 4 and 5].

4.1. The scaling matrix \( D \). For our asymmetric system (3.5), we define two choices of the scaling matrix \( D \).

One choice of \( D \) is called \( \rho \)-scaling [39, equation (6.1)], which can be applied when the diffusion coefficient matrix has the form \( G = \rho G_0 \), where \( G_0 \) is a well-conditioned matrix (constant or changing very mildly) and \( \rho \) is assumed to be a constant in each subdomain, denoted by \( \rho_i \) for the subdomain \( \Omega_i \). Here we allow \( \rho \) having large jumps across subdomain interfaces. We define a positive scaling factor \( \delta_i^\gamma(x) \) as follows: for \( \gamma \in [1/2, \infty) \),

\[
\delta_i^\gamma(x) = \frac{\rho_i^\gamma(x)}{\sum_{j \in N_x} \rho_j^\gamma(x)}, \quad x \in \partial \Omega_i \cap \Gamma_b,
\]

where \( N_x \) is the set of indices \( j \) of the subdomains such that \( x \in \partial \Omega_j \). Since we assume that \( \rho_i(x) \) is constant in each subdomain, \( \delta_i^\gamma(x) \) is constant on each edge/face. The \( \rho \)-scaling matrix \( D \) is a diagonal matrix defined as

\[
D = \text{diag} \left( \delta_i^\gamma(x) \right).
\]

The second choice is called BDDC deluxe scaling, which was first introduced in [20] for the \( H(\text{curl}) \) problem and has been proved to be very robust for different problems that can be formulated as positive definite problems such as in [4, 5, 21, 35, 54]. The deluxe scaling is a block diagonal matrix with each block corresponding to an edge [53]. For a subdomain edge \( \mathcal{E}_{ij} \), which is shared by two subdomains \( \Omega_i \) and \( \Omega_j \), we define two Schur complements,

\[
S^{(k)}_{\mathcal{E}_{ij}} := A^{(k)}_{\mathcal{E}_{ij}} - A^{(k)}_{\mathcal{E}_{ij}I} A^{-1}_{II} A^{(k)}_{\mathcal{E}_{ij}}, \quad k = i, j,
\]
and the deluxe scaling $D$ is defined as
\begin{equation}
D_{E_{ij}^{(k)}} = S_{E_{ij}^{(k)}} \left( S_{E_{ij}^{(i)}} + S_{E_{ij}^{(j)}} \right)^{-1}, \quad k = i, j.
\end{equation}

We note that the Schur complements $S_{E_{ij}^{(i)}}$ can be obtained by restricting the subdomain Schur complement $S_t^{(i)}$, defined in (3.3), to the edge $E_{ij}$. Some economic variants can be found in [24] and the references therein.

### 4.2. The choices of the primal variables.

It is well-known that for (2.1) with the finite element discretization, if the diffusion coefficient matrix is constant in each subdomain, the vertex primal variable will be enough to ensure a good performance for two-dimensional problems. We will show in the next section that the BDDC preconditioned GMRES algorithm will perform well for our asymmetric system from the finite volume element discretization as well. Additional edge average constraints can enhance the convergence [30, 39]. However, when the diffusion coefficient matrix has a large variation inside subdomains, the BDDC algorithms with these standard primal variables can suffer considerably.

The adaptive choice of the primal spaces for the BDDC algorithm applied to symmetric positive definite problems has been a very active research area [53]. However, our system (3.5) is asymmetric, and we cannot apply the adaptive primal variable choices directly to our system. Here we use the finite element system $A_E$ defined in (2.6) to help us to choose the adaptive primal variables since $A_E$ is symmetric positive definite.

For a subdomain edge $E_{ij}$, which is shared by two subdomains $\Omega_i$ and $\Omega_j$, we have defined $S_{E_{ij}}^{(k)}$ for $k = i, j$. Similarly, we define the same Schur complements for $A_E$ as $S_{E_{ij}, E}^{(k)}$, for $k = i, j$. In addition, we need to define another matrix. In order to do that, we let $E_{ij}$ to be the complement of $E_{ij}$ in the set $\Gamma$, the subdomain interface of $\Omega_i$. We write the subdomain Schur complement $S_{E, \Gamma}^{(i)}$, defined in (3.6), as
\begin{equation}
S_{E, \Gamma}^{(i)} = \begin{bmatrix}
S_{E, E_{ij}, E_{ij}}^{(i)} & S_{E, E_{ij}, E_{ij}}^{(i)} \\
S_{E, E_{ij}, E_{ij}}^{(i)} & S_{E, E_{ij}, E_{ij}}^{(i)}
\end{bmatrix}.
\end{equation}

We define $S_{E, E_{ij}, E_{ij}}^{(i)} := S_{E, E_{ij}, E_{ij}}^{(i)} S_{E, E_{ij}, E_{ij}}^{(i)} S_{E, E_{ij}, E_{ij}}^{(i)} S_{E, E_{ij}, E_{ij}}^{(i)} S_{E, E_{ij}, E_{ij}}^{(i)} S_{E, E_{ij}, E_{ij}}^{(i)} S_{E, E_{ij}, E_{ij}}^{(i)} S_{E, E_{ij}, E_{ij}}^{(i)} .

The adaptive primal variables are obtained by solving the generalized eigenvalue problem
\begin{equation}
\left( S_{E, E_{ij}}^{(i)} : S_{E, E_{ij}}^{(j)} \right) x_m = \mu_m \left( S_{E, E_{ij}}^{(i)} : S_{E, E_{ij}}^{(j)} \right) x_m,
\end{equation}

where the parallel sum of two matrices is $A : B = A(A + B)^+ B$ and $(A + B)^+$ denotes a pseudoinverse with
\begin{align*}
(A + B)(A + B)^+ (A + B) &= A + B \quad \text{and} \\
(A + B)^+(A + B)(A + B)^+ &= (A + B)^+.
\end{align*}

The additional primal variables are defined as $S_{E, E_{ij}}^{(i)} : S_{E, E_{ij}}^{(j)} \Lambda$, where $\Lambda$ is the matrix with the eigenvectors associated with eigenvalues smaller than a given tolerance $\frac{1}{p}$ for the columns.

For the adaptive primal subspaces, the deluxe scaling is crucial. The analysis for the BDDC algorithm arising from the finite element discretization with the adaptive primal
subspace and deluxe scaling is provided in [24, 53, 54]. The implementation of the adaptive primal variables can be found in [5, 23].

We use the finite element matrix \( A_E \) to construct the deluxe scaling \( D \) as in (4.2) by replacing the corresponding matrices from \( A \) by those from \( A_E \). The adaptive primal subspaces are obtained using \( \left( S_{E,E,i}^{(i)}, S_{E,E,i}^{(j)} \right) \). In the next section, we will prove the convergence of the BDDC preconditioned GMRES algorithms for (3.5) with the adaptive primal subspace and the deluxe scaling.

We note that for the \( \rho \)-scaling defined in (4.1), we do not need to form the finite element matrix \( D \) and the adaptive primal variables \( u_{\Pi} \). This will require additional memory and computation. However, due to the asymmetry of \( A \), the spectral properties of \( A \) and the generalized eigenvalue problem (4.3) can be complicated. Therefore, we use \( A_E \) in the construction of our BDDC preconditioner. These subdomain \( A_E \) can be deleted as soon as the neighboring subdomain construction is completed.

5. Convergence rate of the GMRES iteration. In this section the convergence of the BDDC preconditioned GMRES for solving the interface problem (3.5) is analyzed.

We first define some useful norms. The partial sub-assembled finite element space \( \hat{W} \) is defined as

\[
\hat{W} = W_I \oplus \hat{W}_\Gamma.
\]

We have \( \hat{W} \subset \hat{W} \), and we denote the injection operator from \( \hat{W} \) to \( \hat{W} \) by \( \hat{R} \).

We define the bilinear forms on \( \hat{W} \) as: for all \( u, v \in \hat{W} \),

\[
\tilde{a}_h(u, v) = \sum_{i=1}^{N} a_h^{(i)}(u^{(i)}, v^{(i)}), \quad \tilde{a}_E(u, v) = \sum_{i=1}^{N} a_E^{(i)}(u^{(i)}, v^{(i)}),
\]

where \( a_h^{(i)} \) and \( a_E^{(i)} \) are the subdomain restrictions of \( a_h \) and \( a_E \) to the subdomain \( \Omega_i \), respectively. \( u^{(i)} \) and \( v^{(i)} \) represent restrictions of \( u \) and \( v \) to the subdomain \( \Omega_i \).

Denote the partially sub-assembled matrices corresponding to the bilinear forms \( \tilde{a}(\cdot, \cdot) \) and \( \tilde{a}_E(\cdot, \cdot) \) by \( \tilde{A} \) and \( \tilde{A}_E \), respectively. We have

\[
A = \tilde{R}^T \tilde{A} \tilde{R} \quad \text{and} \quad A_E = \tilde{R}_E^T \tilde{A}_E \tilde{R}.
\]

Since the subdomain bilinear forms \( a_E^{(i)}(\cdot, \cdot), i = 1, 2, \ldots, N \), are symmetric positive semi-definite on \( W^{(i)} \), we define

\[
\|u^{(i)}\|_{A_E^{(i)}}^2 = a_E^{(i)}(u^{(i)}, u^{(i)}) \quad \text{for any } u^{(i)} \in W^{(i)},
\]

\[
\|u\|_{A_E}^2 = \sum_{i=1}^{N} \|u^{(i)}\|_{A_E^{(i)}}^2, \quad \text{for any } u \in \hat{W}, \quad \text{and}
\]

\[
\|w\|_{A_E}^2 = \sum_{i=1}^{N} \|w^{(i)}\|_{A_E^{(i)}}^2, \quad \text{for any } w \in \hat{W}.
\]

In order to define the norms for \( u_\Gamma \in \hat{W}_\Gamma \), we provide two extensions for \( u_\Gamma \) to the interior of the subdomains. The first extension is the standard discrete harmonic extension
Given $u_{\Gamma}$, we obtain the discrete harmonic extension $u_{\mathcal{H},\Gamma}$ by solving subdomain Dirichlet problems corresponding to the finite element discretization, and $u_{\mathcal{H},\Gamma}$ has minimum energy norm under all finite element functions which have the trace $u_{\Gamma}$ on the interface.

The second discrete extension of $u_{\Gamma} \in \widetilde{W}_{\Gamma}$ to the interior of the subdomains is defined by

\begin{equation}
(5.2) \quad u_{V,\Gamma} = \left[ -A_{I}^{-1} \widehat{A}_{E,I} u_{\Gamma} \right]_{u_{\Gamma}} \in \widetilde{W}_{\Gamma}.
\end{equation}

Given $u_{V}$, we obtain $u_{V,\Gamma}$ by solving subdomain Dirichlet problems corresponding to the finite volume element discretization as shown in (5.2). However, $u_{V,\Gamma}$ does not have the energy minimization property.

We note that for any $u_{\Gamma} \in \widetilde{W}_{\Gamma}$, both $u_{\mathcal{H},\Gamma}$ and $u_{V,\Gamma}$ are also well defined. We have $u_{\mathcal{H},\Gamma} \in \widetilde{W}$ and $u_{V,\Gamma} \in \widetilde{W}$.

We use the notation $(p, q)_{M}$ to represent the product $q^{T} M p$ for any given matrix $M$ and vectors $p$ and $q$. With the extensions defined in (5.1) and (5.2), we have the following lemma.

**Lemma 5.1.** For any $v \in \widetilde{W}$, denote its restriction to $\Gamma$ by $v_{\Gamma} \in \widetilde{W}_{\Gamma}$. Then, for $u_{\Gamma} \in \widetilde{W}_{\Gamma}$ and any $v \in \widetilde{W}$ with $v_{\Gamma}$ on $\Gamma$, we have

\[ \langle u_{\Gamma}, v_{\Gamma} \rangle_{S_{\Gamma}} = \langle u_{V,\Gamma}, v \rangle_{\widehat{A}} \quad \text{and} \quad \langle u_{\Gamma}, v_{\Gamma} \rangle_{S_{E,\Gamma}} = \langle u_{\mathcal{H},\Gamma}, v \rangle_{\widehat{A}_{E}}. \]

The same results also hold for functions and the corresponding bilinear forms in the space $\widetilde{W}_{\Gamma}$.

Using Lemma 5.1, we can define $\|u_{\Gamma}\|^{2}_{S_{E,\Gamma}} = \langle u_{\Gamma}, u_{\Gamma} \rangle_{S_{E,\Gamma}}$, for any $u_{\Gamma} \in \widetilde{W}_{\Gamma}$, and $\|u_{\Gamma}\|^{2}_{S_{E,\Gamma}} = \langle u_{\Gamma}, u_{\Gamma} \rangle_{S_{E,\Gamma}}$, for any $u_{\Gamma} \in \widetilde{W}_{\Gamma}$. Let $T = R_{D,\Gamma} S_{\Gamma}^{-1} R_{D,\Gamma} S_{\Gamma}$ be the preconditioned operator in (3.5). We will use the $S_{E,\Gamma}$-norm to estimate the convergence rate of the GMRES iteration by employing the following result due to Eisenstat, Elman, and Schultz [22].

**Theorem 5.2.** Let $c_{0}$ and $C_{0}$ be two positive constants, independent of $H$, $h$, and $G$ in (2.1), such that

\[ c_{0} \langle u, u \rangle_{S_{E,\Gamma}} \leq \langle u, Tu \rangle_{S_{E,\Gamma}}, \quad \langle Tu, Tu \rangle_{S_{E,\Gamma}} \leq C_{0}^{2} \langle u, u \rangle_{S_{E,\Gamma}}. \]

Then,

\[ \frac{\|r_{m}\|_{S_{E,\Gamma}}}{\|r_{0}\|_{S_{E,\Gamma}}} \leq \left( 1 - \frac{c_{0}^{2}}{C_{0}^{2}} \right)^{m/2}, \]

where $r_{m}$ is the residual of the $m$-th iteration of GMRES.

In the rest of this section, we will estimate the lower bound $c_{0}$ and the upper bound $C_{0}^{2}$ in Theorem 5.2. We use $c$ and $C$ to denote constants that are independent of $H$, $h$, and $G$ in (2.1).

We first need to establish some useful connections between the systems from the finite element and the finite volume element discretizations. Using [14, Lemma 3.1], we have the following lemma.

**Lemma 5.3.**

\[ |\langle u, v \rangle_{A_{E}} - \langle u, v \rangle_{A_{E}}| \leq C h \|u\|_{A_{E}} \|v\|_{A_{E}}, \quad \forall u, v \in \widetilde{W}, \]
The result holds for the corresponding subdomain versions as well.

Using Lemma 5.3, we obtain the following two lemmas.

**Lemma 5.4.** There exists a positive constant $h_1 > 0$ such that for $h < h_1$,

$$c \langle u, u \rangle_A \leq \| u \|_{A_E}^2 \leq C \langle u, u \rangle_A,$$

and

$$c \langle u, u \rangle_{\overline{A}} \leq \| u \|_{\overline{A}_E}^2 \leq C \langle u, u \rangle_{\overline{A}}.$$  

The result holds for the corresponding subdomain versions as well.

**Proof.** By Lemma 5.3, we have

$$\| u \|_{A_E}^2 \leq \langle u, u \rangle_A + Ch \| u \|_{A_E}^2 \quad \text{and hence} \quad (1 - Ch) \| u \|_{A_E}^2 \leq \langle u, u \rangle_A.$$ 

There exists a positive constant $h_1 > 0$ such that $1 - Ch > 1/2$ for all $h < h_1$. Therefore, we have $\| u \|_{A_E}^2 \leq C \langle u, u \rangle_A$.

On the other hand, by Lemma 5.3, we have

$$\langle u, u \rangle_A \leq \langle u, u \rangle_{A_E} + Ch \| u \|_{A_E}^2 \leq (1 + Ch) \| u \|_{A_E}^2 \leq 2 \| u \|_{A_E}^2$$

if $h < h_1$. \(\Box\)

Similarly, using Lemma 5.3 and the Cauchy-Schwarz inequality, we can prove the following lemma.

**Lemma 5.5.** For $h < h_1$,

$$\langle u, v \rangle_A \leq C \| u \|_{A_E} \| v \|_{A_E}, \quad \forall u, v \in \widehat{W},$$

and

$$\langle u, v \rangle_{\overline{A}} \leq C \| u \|_{\overline{A}_E} \| v \|_{\overline{A}_E}, \quad \forall u, v \in \widehat{W}.$$  

The result holds for the corresponding subdomain versions as well.

**Lemma 5.6.** For $h < h_1$,

$$\left| \langle u, v \rangle_{S_\Gamma} - \langle u, v \rangle_{S_{E,\Gamma}} \right| \leq Ch \| u \|_{S_{E,\Gamma}} \| v \|_{S_{E,\Gamma}}, \quad \forall u, v \in \overline{W}_\Gamma,$$

and

$$\left| \langle u, v \rangle_{\overline{S}_\Gamma} - \langle u, v \rangle_{\overline{S}_{E,\Gamma}} \right| \leq Ch \| u \|_{\overline{S}_{E,\Gamma}} \| v \|_{\overline{S}_{E,\Gamma}}, \quad \forall u, v \in \overline{W}_\Gamma.$$

**Proof.** Given any $u_\Gamma, v_\Gamma \in \overline{W}_\Gamma$. Then, by Lemma 5.1,

$$\left| \langle u, v \rangle_{S_\Gamma} - \langle u, v \rangle_{S_{E,\Gamma}} \right| = \left| \langle u, v \rangle_{S_{E,\Gamma}} - \langle u_\Gamma, v_\Gamma \rangle_{A_E} \right|$$

(5.3)

$$= \left| \langle u, v \rangle_A - \langle u_\Gamma, v_\Gamma \rangle_{A_E} \right| \leq Ch \| u \|_{A_E} \| v \|_{A_E},$$
where Lemma 5.3 is used in the last step. Using Lemmas 5.4, 5.1, and 5.5, we have
\[
\|u_{V,G}\|_{A_E}^2 \leq C \langle u_{V,G}, u_{V,G} \rangle_A = C \langle u_{\Gamma}, u_{\Gamma} \rangle_{S_{\Gamma}} = \langle u_{V,G}, u_{\hat{H},G} \rangle_A \leq C \| u_{V,G} \|_{A_E} \| u_{\hat{H},G} \|_{A_E}.
\]

Dividing on both sides by \( u_{V,G} \) we obtain
\[
\|u_{V,G}\|_{A_E} \leq C \| u_{\hat{H},G} \|_{A_E}.
\]

Similarly, we can prove the result for any \( u_{\Gamma}, v_{\Gamma} \in \overline{W}_{\Gamma} \). □

Using Lemma 5.6, we obtain the following lemma.

**Lemma 5.7.** There exists a positive constant \( h_0 < h_1 \), \( c \), and \( C \) such that if \( h < h_0 \), then
\[
\langle u_{\Gamma}, v_{\Gamma} \rangle_{S_{\Gamma}} \leq C \| u_{\Gamma} \|_{S_{E,\Gamma}} \| v_{\Gamma} \|_{S_{E,\Gamma}},
\]
\[
c \langle u_{\Gamma}, u_{\Gamma} \rangle_{S_{\Gamma}} \leq \| u_{\Gamma} \|^2_{S_{E,\Gamma}} \leq C \langle u_{\Gamma}, u_{\Gamma} \rangle_{S_{\Gamma}}, \quad \forall u_{\Gamma}, v_{\Gamma} \in \overline{W}_{\Gamma},
\]
and
\[
\langle u_{\Gamma}, v_{\Gamma} \rangle_{\overline{S}_{\Gamma}} \leq C \| u_{\Gamma} \|_{\overline{S}_{E,\Gamma}} \| v_{\Gamma} \|_{\overline{S}_{E,\Gamma}},
\]
\[
c \langle u_{\Gamma}, u_{\Gamma} \rangle_{\overline{S}_{\Gamma}} \leq \| u_{\Gamma} \|^2_{\overline{S}_{E,\Gamma}} \leq C \langle u_{\Gamma}, u_{\Gamma} \rangle_{\overline{S}_{\Gamma}}, \quad \forall u_{\Gamma}, v_{\Gamma} \in \overline{W}_{\Gamma}.
\]

For any \( u_{\Gamma} \in \overline{W}_{\Gamma} \), we define \( E_{D,G} w_{\Gamma} = \overline{R}_{\Gamma} \overline{R}_{D,G} w_{\Gamma} \), which computes an average of \( w_{\Gamma} \) across \( \Gamma \). The estimate of \( E_{D,G} \) plays an important role in the analysis of the BDDC algorithms [26, 27, 29, 43, 47]. We make the following assumption.

**Assumption 5.8.** We assume that the coarse-level primal subspace \( \overline{W}_{\Omega} \) and the scaling \( D \) can ensure that there exists a positive constant \( C \), which is independent of the diffusion coefficient matrix \( G \), \( H \) and \( h \), such that it holds for all \( w_{\Gamma} \in \overline{W}_{\Gamma} \),
\[
\| E_{D,G} w_{\Gamma} \|^2_{S_{E,\Gamma}} \leq C \Phi^2(H, h) \| w_{\Gamma} \|^2_{S_{E,\Gamma}}.
\]

**Theorem 5.9 (Minimal coarse space).** The coarse-level primal subspace \( \overline{W}_{\Omega} \) includes all subdomain vertices. The BDDC algorithm with the deluxe scaling \( D \) gives \( \Phi^2(H, h) = C \max_{1 \leq i \leq N} \kappa_i \left( 1 + \log{(H/h)} \right)^2 \) in Assumption 5.8, where \( \kappa_i = \max_{x \in \Omega_i} \frac{\alpha(x)}{\alpha(x^*)} \) and \( \alpha_l \) and \( \alpha_u \) are the minimum and maximum eigenvalues of \( G^{(i)} \) in the subdomain \( \Omega_i \), respectively. When the diffusion coefficient matrix is constant in each subdomain, the BDDC algorithm with the \( p \)-scaling gives \( \Phi^2(H, h) = C \left( 1 + \log{(H/h)} \right)^2 \) in Assumption 5.8.

**Proof.** For the BDDC algorithms with the \( p \)-scaling defined in (4.1), the proof follows the analysis of the BDDC algorithms for two dimensions in [33].

For the BDDC algorithms with the deluxe scaling defined in (4.2) using the finite element matrix \( A_E \), the proof follows the analysis in [53, Section 4.2]. □

The following theorem is well established for the finite element discretization [24, 53].
THEOREM 5.10 (Adaptive coarse space with deluxe scaling). The coarse-level primal subspace $\tilde{W}_{\Omega}$ includes all the eigenvectors of the generalized eigenvalue problems (4.3) with corresponding eigenvalues smaller than $\frac{1}{\nu}$ for each edge on the subdomain interface. Moreover, if the deluxe scaling $D$ is calculated by the finite element matrix $A_E$, then $\Phi^2(H, h) = \nu$ in Assumption 5.8.

Given $u_T \in \hat{W}_T$, we define

$$w_T = \tilde{S}_T^{-1} \tilde{R}_{D,T} S_T u_T.$$  \hspace{1cm} (5.5)

LEMMA 5.11. $\|w_T\|^2_{\tilde{S}_{T,\Gamma}} \leq C \langle u_T, Tu_T \rangle_{S_T}$ if $h < h_0$.

Proof. Since $\tilde{R}_{D,T}^T w_T = \tilde{R}_{D,T}^T \tilde{S}_T^{-1} \tilde{R}_{D,T} S_T u_T = Tu_T$, we have, using Lemma 5.7,

$$\|w_T\|^2_{\tilde{S}_{T,\Gamma}} \leq C \langle u_T, w_T \rangle_{\tilde{S}_T} = C w_T^T \tilde{S}_T w_T = C w_T^T \tilde{S}_T \tilde{S}_T^{-1} \tilde{R}_{D,T} S_T u_T$$

$$= C w_T^T \tilde{R}_{D,T} S_T u_T = C \langle u_T, \tilde{R}_{D,T}^T w_T \rangle_{S_T} = C \langle u_T, Tu_T \rangle_{S_T}. \quad \Box$$

LEMMA 5.12. Let Assumption 5.8 hold and $h < h_0$. There exists a positive constant $C$, independent of $H$ and $h$, such that for all $u_T \in \hat{W}_T$,

$$\langle Tu_T, Tu_T \rangle_{S_{T,\Gamma}} \leq C \Phi^4 (H, h) \langle u_T, u_T \rangle_{S_{T,\Gamma}}.$$

Proof. We have, from Lemma 5.7,

$$\langle Tu_T, Tu_T \rangle_{S_{T,\Gamma}} \leq C \langle Tu_T, Tu_T \rangle_{S_T} = C \langle \tilde{R}_{D,T}^T \tilde{S}_T^{-1} \tilde{R}_{D,T} S_T u_T, \tilde{R}_{D,T}^T \tilde{S}_T^{-1} \tilde{R}_{D,T} S_T u_T \rangle_{S_T}$$

$$= C \langle \tilde{R}_{T} \tilde{R}_{D,T}^T w_T, \tilde{R}_{T} \tilde{R}_{D,T}^T w_T \rangle_{S_T} = C \langle E_D w_T, E_D w_T \rangle_{S_T}$$

$$\leq C \|E_D w_T\|^2_{S_{T,\Gamma}}. \quad \Box$$

Here we recall that $w_T$ is defined in (5.5). Now, using Assumption 5.8 and Lemmas 5.11 and 5.7, we find

$$\langle Tu_T, Tu_T \rangle_{S_{T,\Gamma}} \leq C \|E_D w_T\|^2_{S_{T,\Gamma}} \leq C \Phi^2 (H, h) \|w_T\|^2_{\tilde{S}_{T,\Gamma}} \leq C \Phi^2 (H, h) \langle u_T, Tu_T \rangle_{S_T}$$

$$\leq C \Phi^2 (H, h) \|Tu_T\|_{S_{T,\Gamma}} \|u_T\|_{S_{T,\Gamma}}.\quad \Box$$

Cancelling the common factor and squaring both sides, we obtain

$$\langle Tu_T, Tu_T \rangle_{S_{T,\Gamma}} \leq C \Phi^4 (H, h) \langle u_T, u_T \rangle_{S_{T,\Gamma}} \Box$$

THEOREM 5.13. Let Assumption 5.8 hold and $h < h_0$. The constants $c_0$ and $C_0$ in Theorem 5.2 can be chosen as $C_0^2 = C \Phi^4 (H, h)$ and $c_0 = 1 - C h \Phi^2 (H, h)$.

Proof. The upper bound $C_0^2$ is proved in (5.6). We only need to prove the lower bound $c_0$. Using $\tilde{R}_{T} \tilde{R}_{D,T} = I$ in (3.4) and Lemma 5.7, we have

$$\langle u_T, u_T \rangle_{S_{T,\Gamma}} \leq C \langle u_T, u_T \rangle_{S_T} = C w_T^T \tilde{R}_{T} \tilde{S}_T^{-1} \tilde{R}_{D,T} S_T u_T = C \langle u_T, \tilde{R}_{T} u_T \rangle_{S_T}$$

$$\leq C \|u_T\|^2_{S_{T,\Gamma}} \|u_T\|_{S_{T,\Gamma}} \leq C \langle u_T, Tu_T \rangle_{S_T}^{1/2} \|u_T\|_{S_{T,\Gamma}}.$$

where we have used Lemmas 5.7 and 5.11 for the last two inequalities. Cancelling the common term, we arrive at

$$\|u_T\|^2_{S_{T,\Gamma}} \leq C \langle u_T, Tu_T \rangle_{S_T}. \Box$$
Then, using Lemmas 5.7, 5.6, and 5.12, we obtain

\[
\langle u_\Gamma, u_\Gamma \rangle_{SE,\Gamma} \leq C \left( \langle u_\Gamma, Tu_\Gamma \rangle_{SE,\Gamma} + \left( \langle u_\Gamma, Tu_\Gamma \rangle_{S\Gamma} - \langle u_\Gamma, Tu_\Gamma \rangle_{SE,\Gamma} \right) \right)
\]

\[
\leq C \langle u_\Gamma, Tu_\Gamma \rangle_{SE,\Gamma} + C h \| u_\Gamma \|_{SE,\Gamma} \| Tu_\Gamma \|_{SE,\Gamma}
\]

\[
\leq C \langle u_\Gamma, Tu_\Gamma \rangle_{SE,\Gamma} + C h \left( \Phi^2(H, h) \right) \langle u_\Gamma, u_\Gamma \rangle_{SE,\Gamma}.
\]

Collecting the term \( \langle u_\Gamma, u_\Gamma \rangle_{SE,\Gamma} \) gives the desired estimate of \( c_0 \).

REMARK 5.14. If \( h \) is sufficiently small, then \( c_0 \) will be positive and bounded from zero independently of \( H \). Hence, from Theorem 5.2, the convergence rate of the GMRES algorithm for solving (3.5) becomes bounded independently of the number of subdomains.

6. Numerical experiments. We test our BDDC algorithms by solving three examples in the square domain \( \Omega = [0,1]^2 \). The domain \( \Omega \) is decomposed into several square subdomains, and each subdomain is triangulated uniformly as shown in Figure 6.1. Piecewise linear finite elements are used in our experiments.

A GMRES iteration with the \( L^2 \)-norm is used without restart to solve the preconditioned interface problem (3.5). The iteration stops if the \( L^2 \)-norm of the residual reaches a reduction of \( 10^{-8} \). We have found consistently that the convergence rate using the \( S_{E,\Gamma} \)-norm is quite similar to that using the \( L^2 \)-norm.

Our first test example considers \( G = \rho(2 + \sin(x\pi) \sin(y\pi)) \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \) as in [34], where \( \rho \) has checkerboard patterns as displayed in Figure 6.2. In the second example it is used \( G = \rho \begin{bmatrix} 2 + x & 0 \\ 0 & 2 + y \end{bmatrix} \). We note that the second example is similar to that in [55] except we add the factor \( \rho \) to consider possible jumps of the coefficient across subdomain interfaces. Our third example considers the coefficients from the permeability tensor from the SPE10 benchmark [37] as shown in Figure 6.3.

For the first two examples, the coefficient jumps only across the subdomain interfaces. We use the vertex constraint and the simple \( \rho \)-scaling \( D \) defined in (4.1). For this setup, we do not
need to form the finite element matrix $A_E$ and related Schur complements. We have two sets of numerical experiments for each example. We first change the number of subdomains and fix the subdomain local problem size. The second set is to change the subdomain local problem size with a fixed number of subdomains. In each set, we have chosen constant coefficients and a checkerboard pattern; see Figure 6.2. In our numerical experiments, we take $a = 1$ or $a = 1000$. The results are reported in Tables 6.1 and 6.2. We find, for both examples, that the number of GMRES iterations is independent of the number of subdomains and grows slowly with increasing ratio $H/h$. To compare, we also provide the number of GMRES iterations without any preconditioner for Example I in Table 6.1. From the results, we can see that the BDDC preconditioner controls the number of iterations as we have established in our theory. For the second example, we also test our BDDC algorithms with the deluxe scaling defined in (4.2). For this simple example, the deluxe scaling gives exactly the same numbers of GMRES iterations as those with $\rho$-scaling.

For the third example, since the coefficient has large jumps inside the subdomains, we expect the vertex constraints with a simple $\rho$-scaling $D$ not to work well (here we take one arbitrary value of $G$ in the subdomain $\Omega_i$ as $\rho_i$ to define $D$ in (4.1)). The results are reported
ETNA
Kent State University and
Johann Radon Institute (RICAM)

Table 6.1
Example I: GMRES iteration counts for the BDDC algorithm with vertex constraints (numbers in parentheses are for GMRES iterations without preconditioner and restart at 30).

<table>
<thead>
<tr>
<th>Num. of sub. ((\frac{H}{h} = 8))</th>
<th>(a = 1)</th>
<th>(a = 1000)</th>
<th>(\frac{H}{h}) (64 subs)</th>
<th>(a = 1)</th>
<th>(a = 1000)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 \times 4</td>
<td>12 (49)</td>
<td>10 (925)</td>
<td>4</td>
<td>12</td>
<td>11</td>
</tr>
<tr>
<td>8 \times 8</td>
<td>15 (115)</td>
<td>14 (&gt; 3000)</td>
<td>8</td>
<td>15</td>
<td>14</td>
</tr>
<tr>
<td>16 \times 16</td>
<td>16 (245)</td>
<td>15 (&gt; 3000)</td>
<td>16</td>
<td>19</td>
<td>17</td>
</tr>
<tr>
<td>32 \times 32</td>
<td>16 (681)</td>
<td>15 (&gt; 3000)</td>
<td>32</td>
<td>22</td>
<td>20</td>
</tr>
</tbody>
</table>

Table 6.2
Example II: GMRES iteration counts for the BDDC algorithm with vertex constraints and both \(\rho\)-scaling and deluxe scaling (numbers in parentheses are for deluxe scaling).

<table>
<thead>
<tr>
<th>Num. of sub. ((\frac{H}{h} = 8))</th>
<th>(a = 1)</th>
<th>(a = 1000)</th>
<th>(\frac{H}{h}) (64 subs)</th>
<th>(a = 1)</th>
<th>(a = 1000)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 \times 4</td>
<td>11 (11)</td>
<td>9 (9)</td>
<td>4</td>
<td>12 (12)</td>
<td>11 (11)</td>
</tr>
<tr>
<td>8 \times 8</td>
<td>15 (15)</td>
<td>14 (14)</td>
<td>8</td>
<td>15 (15)</td>
<td>14 (14)</td>
</tr>
<tr>
<td>16 \times 16</td>
<td>16 (16)</td>
<td>15 (15)</td>
<td>16</td>
<td>18 (18)</td>
<td>17 (17)</td>
</tr>
<tr>
<td>32 \times 32</td>
<td>17 (17)</td>
<td>15 (15)</td>
<td>32</td>
<td>21 (21)</td>
<td>20 (20)</td>
</tr>
</tbody>
</table>

Table 6.3
Example III: GMRES iteration counts for the BDDC algorithms.

<table>
<thead>
<tr>
<th>(\rho) scaling</th>
<th>Deluxe scaling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Num. of sub. ((\frac{H}{h} = 8))</td>
<td>vertex</td>
</tr>
<tr>
<td>4 \times 4</td>
<td>36</td>
</tr>
<tr>
<td>8 \times 8</td>
<td>61</td>
</tr>
<tr>
<td>16 \times 16</td>
<td>135</td>
</tr>
<tr>
<td>32 \times 32</td>
<td>199</td>
</tr>
</tbody>
</table>

Table 6.4
Example III: GMRES iteration counts for the BDDC algorithm with the deluxe scaling.

<table>
<thead>
<tr>
<th>Num. of sub. ((\frac{H}{h} = 8))</th>
<th>(\nu = 100)</th>
<th>(\nu = 20)</th>
<th>(\nu = 10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>vertex</td>
<td>edge+vertex</td>
<td>Iter.</td>
<td>nc</td>
</tr>
<tr>
<td>4 \times 4</td>
<td>17</td>
<td>9</td>
<td>12</td>
</tr>
<tr>
<td>8 \times 8</td>
<td>38</td>
<td>49</td>
<td>21</td>
</tr>
<tr>
<td>16 \times 16</td>
<td>72</td>
<td>225</td>
<td>34</td>
</tr>
<tr>
<td>32 \times 32</td>
<td>124</td>
<td>961</td>
<td>46</td>
</tr>
</tbody>
</table>

in Table 6.3. The number of GMRES iterations increases from 36 to 199 when increasing the number of subdomains from 16 to 1024. Additional edge average constraints are enforced to improve the performance. However, the number of iterations is still increasing. We repeat the same primal constraints with the deluxe scaling matrix \(D\) defined in (4.2). We recall that we need to form the deluxe \(D\) using the finite element matrix \(A_E\). With the deluxe scaling, the performance of the BDDC algorithms is improved for both vertex and edge average constraints. The number of GMRES iterations is still increasing quite a bit with an increasing number of subdomains.

We then have applied the adaptive BDDC algorithms with deluxe scaling using different choices of \(\nu\). The number of GMRES iterations and the number of used primal variables
(denoted as \( \text{nec} \)) are reported in Table 6.4. With a quite large \( \nu = 100 \), the number of selected primal variables is comparable with the vertex and edge average constraints (some cases are even smaller). However, the number of the GMRES iterations are much smaller. When decreasing the value of \( \nu \), the number of primal variables increases, but the number of GMRES iterations is well controlled.

Acknowledgments. The first and second authors’ work were supported in part by the NSF under Contract DMS-1723066.

REFERENCES


https://etna.ricam.oeaw.ac.at/vol.52.2020/pp553-570.dir/pp553-570.pdf


