Preconditioning of Saddle Point Systems by Substructuring and a Penalty Approach

Clark R. Dohrmann[1]

Sandia National Laboratories, crdohrm@sandia.gov. Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the U.S. Department of Energy under contract DE-AC04-94AL85000.

Summary. The focus of this paper is a penalty-based strategy for preconditioning elliptic saddle point systems. As the starting point, we consider the regularization approach of Axelsson in which a related linear system, differing only in the (2,2) block of the coefficient matrix, is introduced. By choosing this block to be negative definite, the dual unknowns of the related system can be eliminated resulting in a positive definite primal Schur complement. Rather than solving the Schur complement system exactly, an approximate solution is obtained using a preconditioner. The approximate primal solution together with the recovered dual solution then define the preconditioned residual for the original system. The approach can be applied to a variety of different saddle point problems.

Although the preconditioner itself is symmetric and indefinite, all the eigenvalues of the preconditioned system are real and positive if certain conditions hold. Stronger conditions also ensure that the eigenvalues are bounded independently of mesh parameters. An interesting feature of the approach is that conjugate gradients can be used as the iterative solution method rather than GMRES.

The effectiveness of the overall strategy hinges on the preconditioner for the primal Schur complement. Interestingly, the primary condition ensuring real and positive eigenvalues is satisfied automatically in certain instances if a Balancing Domain Decomposition by Constraints (BDDC) preconditioner is used. Following an overview of BDDC, we show how its constraints can be chosen to ensure insensitivity to parameter choices in the (2,2) block for problems with a divergence constraint. Examples for different saddle point problems are presented and comparisons made with other approaches.

1 Introduction

We consider linear systems

$$\begin{bmatrix} A & B^T \\ B & -C \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix} \tag{1}$$

arising from finite element discretizations of saddle point problems. The matrix A is assumed to be symmetric and positive definite on the kernel of B. The matrix

B is assumed to have full rank and C is assumed to be symmetric and positive semidefinite. The primal and dual vectors are denoted by $u \in \mathbb{R}^n$ and $p \in \mathbb{R}^m$, respectively. For example, in the case of Stokes flow and incompressible elasticity, the primal and dual variables are associated with velocity–pressure and displacement–pressure, respectively.

Several preconditioners for (1) have been investigated. Many approaches are based on preconditioning the dual Schur complement $C + BA^{-1}B^T$ by a matrix that is spectrally equivalent to the dual mass matrix. Examples include block diagonal preconditioners [17], block triangular preconditioners [9], and inexact Uzawa approaches [7]. Reformulation of the saddle point problem in (1) as a symmetric positive definite system was considered in [3] that permits an iterative solution using the conjugate gradient algorithm. Overlapping Schwarz preconditioners involving solutions of both local and coarse saddle point problems were investigated in [10]. More recently, substructuring preconditioners based on balancing Neumann-Neumann methods [15, 8] and FETI-DP [11] were studied.

Our approach builds on the idea of preconditioning indefinite problems using a regularization approach [1] introduced by Axelsson. Preconditioning based on regularization is motivated by the idea that the solution of a penalized problem is close to that of the original constrained problem. We present theory and numerical results that extends [1] to cases where the penalized primal Schur complement $S_A = A + B^T\tilde{C}^{-1}B$ is preconditioned rather than factored directly. Here, $\tilde{C}$ is a symmetric positive definite penalty counterpart of C in (1).

The preconditioner for (1) is most readily applied to discretizations employing discontinuous interpolation of the dual variable. In such cases the dual variable can be eliminated at the element level and S_A has the same sparsity structure as A. Consequently, preconditioning strategies available for A can also be used for S_A. As will be shown, the effectiveness of the overall approach hinges on the preconditioner for S_A.

Significant portions of this paper are based on two recent technical reports [6, 5]. Material taken directly from [6] includes a statement, without proof, of its main result in Section 2 and a form of the preconditioner suited for conjugate gradients in Section 3. New material related to [6] includes additional theory for the special case of $C = 0$ in Section 2 and an extension of numerical results in Tables 5.1 and 5.2 of the cited reference in Section 6. An overview of the BDDC preconditioner is provided in Section 4. In Section 5 we show how to choose the constraints in BDDC to accommodate problems with a divergence constraint. Numerical examples in Section 6 confirm the theory and demonstrate the excellent performance of the preconditioner. Comparisons are also made with block diagonal and block triangular preconditioners for saddle point systems.

2 Preconditioner and Theory

The penalized primal Schur complement S_A is defined as

$$S_A = A + B^T\tilde{C}^{-1}B \tag{2}$$

where $\tilde{C}$ is symmetric and positive definite. Since A is assumed to be positive definite on the kernel of B, it follows that S_A is positive definite. We consider a preconditioner $\mathcal{M}$ of the form

$$\mathcal{M} = \begin{bmatrix} I & B^T\tilde{C}^{-1} \\ 0 & -I \end{bmatrix} \begin{bmatrix} \hat{S}_A & 0 \\ 0 & -\tilde{C} \end{bmatrix} \begin{bmatrix} I & 0 \\ \tilde{C}^{-1}B & -I \end{bmatrix} \tag{3}$$

where $\hat{S}_A$ is a preconditioner for S_A. The action of the preconditioner on a vector r (with primal and dual subvectors r_u and r_p) is

$$\begin{bmatrix} z_u \\ z_p \end{bmatrix} = \begin{bmatrix} I & 0 \\ \tilde{C}^{-1}B & -I \end{bmatrix} \begin{bmatrix} \hat{S}_A^{-1} & 0 \\ 0 & -\tilde{C}^{-1} \end{bmatrix} \begin{bmatrix} I & B^T\tilde{C}^{-1} \\ 0 & -I \end{bmatrix} \begin{bmatrix} r_u \\ r_p \end{bmatrix} \tag{4}$$

leading to the two step application of $\mathcal{M}^{-1}r$ as

1. Solve $\hat{S}_A z_u = r_u + B^T\tilde{C}^{-1}r_p$ for z_u,
2. Solve $\tilde{C}z_p = Bz_u - r_p$ for z_p.

Each application of the preconditioner requires two solves with $\tilde{C}$ and one solve with $\hat{S}_A$.

Consider the eigenvalues ν of the generalized eigenproblem

$$\mathcal{A}z = \nu\mathcal{M}z \tag{5}$$

where $\mathcal{A}$ is the coefficient matrix in (1). Using a coordinate transformation, these eigenvalues are identical to those of the generalized eigenproblem

$$\mathcal{A}\mathcal{M}^{-1}\mathcal{H}w = \nu\mathcal{H}w \tag{6}$$

where $\mathcal{H}$ is defined as

$$\mathcal{H} = \begin{bmatrix} S_A - \hat{S}_A & 0 \\ 0 & \tilde{C} - C \end{bmatrix}. \tag{7}$$

The following theorem is taken from [6].

Theorem 1. *If* $\alpha_1 > 1,\ 0 \le \beta_1 < \beta_2 < 1,\ \gamma_1 > 0$, *and*

$$\alpha_1 x^T\hat{S}_A x \le x^T S_A x \le \alpha_2 x^T\hat{S}_A x \quad \forall x \in \mathbb{R}^n, \tag{8}$$

$$\beta_1 y^T\tilde{C}y \ \le y^T C y \le \ \beta_2 y^T\tilde{C}y \quad \forall y \in \mathbb{R}^m, \tag{9}$$

$$\gamma_1 y^T B\hat{S}_A^{-1}B^T y \ \le y^T\tilde{C}y \le \ \gamma_2 y^T B\hat{S}_A^{-1}B^T y \quad \forall y \in \mathbb{R}^m, \tag{10}$$

and

$$0 < y^T\tilde{C}y \quad \forall y \in \mathbb{R}^m, \tag{11}$$

then the eigenvalues of (6) are real and satisfy

$$\delta_1 \le \nu \le \delta_2$$

where

$$\delta_1 = \min\{\sigma_2(\alpha_1/\alpha_2), \beta_1 + \sigma_1(1-\beta_2)(\alpha_2\gamma_2)^{-1}\}$$
$$\delta_2 = \max\{2\alpha_2 - \sigma_2, \beta_2 + (1-\beta_1)(2-\sigma_1/\alpha_2)\gamma_1^{-1}\}$$

and σ_1, σ_2 *are arbitrary positive constants that satisfy* $\sigma_1 + \sigma_2 = 1$.

Notice in (8) that α_1 and α_2 depend on the preconditioner for S_A. In order to obtain bounds for γ_1 and γ_2 in (10), it proves useful to express A as

$$A = B^T A_1 B + B_\perp^T A_2 B_\perp + B^T A_3 B_\perp + B_\perp^T A_3^T B$$

where the columns of $B_\perp$ form an orthonormal basis for the null space of B and

$$A_1 = (BB^T)^{-1}BAB^T(BB^T)^{-1}, \quad A_2 = B_\perp AB_\perp, \quad A_3 = (BB^T)^{-1}BAB_\perp^T .$$

Using a similar expression for S_A^{-1} and the identity $S_A S_A^{-1} = I$ we obtain

$$BS_A^{-1}B^T = (\tilde{C}^{-1} + G)^{-1} \quad \text{where} \quad G = A_1 - A_3A_2^{-1}A_3^T = R^TR .$$

Notice that A_2 is nonsingular since A was assumed positive definite on the kernel of B. In addition, G is at least positive semidefinite since it is independent of $\tilde{C}$ and $BS_A^{-1}B$ is positive definite. Application of the Sherman-Morrison-Woodbury formula leads to

$$BS_A^{-1}B^T = \tilde{C} - \tilde{C}R^T(I + R\tilde{C}R^T)^{-1}R\tilde{C} . \tag{12}$$

We now consider the special case $C = 0$ and the parameterization $\tilde{C} = \zeta\bar{C}$. The positive scalar ζ is chosen so that

$$\zeta\|\bar{C}R^T(I + \zeta R\bar{C}R^T)^{-1}R\bar{C}\| < \epsilon\lambda_{\min}(\bar{C}) \tag{13}$$

where $\epsilon > 0$ and $\lambda_{\min}(\bar{C})$ is the smallest eigenvalue of $\bar{C}$. It then follows from (8), (12), and (13) that

$$(1/\alpha_2)y^TB\hat{S}_A^{-1}B^Ty \le y^T\tilde{C}y \le (1/\alpha_1)(1-\epsilon)^{-1}y^TBS_A^{-1}B^Ty \quad \forall y \in \mathbb{R}^m \tag{14}$$

Comparison of (10) and (14) reveals that

$$\gamma_1 \ge 1/\alpha_2 \quad \text{and} \quad \gamma_2 \le (1/\alpha_1)(1-\epsilon)^{-1} .$$

Notice from (9) for $C = 0$ that $\beta_1 = 0$ and β_2 can be chosen arbitrarily close to 0. The expressions for the eigenvalue bounds with σ_1 and σ_2 both chosen as $1/2$ then simplify to

$$\delta_1 = (1-\epsilon)(\alpha_1/\alpha_2)/2, \quad \delta_2 = 2\alpha_2 - 1/2 .$$

For very small values of ϵ we see that the eigenvalue bounds depend only on the parameters α_1 and α_2 which are related to the preconditioner. This result is purely algebraic and does not involve any inf-sup constants. For α_1 and α_2 both near 1 we see that all eigenvalues are bounded between $(1-\epsilon)/2$ and $3/2$. Numerical results in Section 6 suggest that these bounds could be made even tighter. In Section 5 we show how to choose the constraints of a BDDC preconditioner so that α_1 and α_2 are insensitive to mesh parameters and to values of ϵ near zero.

3 Preconditioned Conjugate Gradient Algorithm

We now consider a form of the preconditioner suitable for the conjugate gradient algorithm. The original linear system (1) can be expressed compactly as

$$\mathcal{A}w = d$$

where

$$w = \begin{bmatrix} u \\ p \end{bmatrix} \quad \text{and} \quad d = \begin{bmatrix} b \\ 0 \end{bmatrix} .$$

The associated residual r is defined as

$$r = d - \mathcal{A}w .$$

Because the matrices $\mathcal{H}\mathcal{M}^{-1}\mathcal{A}$ and $\mathcal{H}$ are both symmetric and positive definite (see [6] for details), the conjugate gradient algorithm can be used to solve the equivalent linear system

$$\tilde{\mathcal{A}}w = \tilde{d}$$

where

$$\tilde{A} = \mathcal{H}\mathcal{M}^{-1}\mathcal{A} \quad \text{and} \quad \tilde{d} = \mathcal{H}\mathcal{M}^{-1}d$$

using $\mathcal{H}$ as a preconditioner. Since the eigenvalues of (5) and (6) are identical, it follows that the eigenvalues of the preconditioned system are bounded below and above by δ_1 and δ_2. The preconditioned conjugate gradient algorithm for the equivalent linear system is summarized as follows:

1. $w_0 = 0$, $r_0 = d$, $z_0 = \mathcal{M}^{-1}r_0$, $\tilde{r}_0 = \mathcal{H}\mathcal{M}^{-1}r_0$, and $k = 1$.
2. If the norm of r_{k-1} is less than a specified value then exit. Otherwise,
3. $\beta_k = (z_{k-1}^T\tilde{r}_{k-1})/(z_{k-2}^T\tilde{r}_{k-2}) \qquad (\beta_1 = 0)$.
4. $p_k = z_{k-1} + \beta_k p_{k-1} \qquad (p_1 = z_0)$.
5. $\alpha_k = (z_{k-1}\tilde{r}_{k-1})/(p_k^T\mathcal{H}\mathcal{M}^{-1}\mathcal{A}p_k)$.
6. $w_k = w_{k-1} + \alpha_k p_k$.
7. $r_k = r_{k-1} - \alpha_k\mathcal{A}p_k$.
8. $z_k = z_{k-1} - \alpha_k\mathcal{M}^{-1}\mathcal{A}p_k$.
9. $\tilde{r}_k = \tilde{r}_{k-1} - \alpha_k\mathcal{H}\mathcal{M}^{-1}\mathcal{A}p_k$.
10. Return to Step 2.

The conjugate gradient algorithm described above is somewhat nonstandard in that two additional recurrences appear in Steps 7 and 8. Application of the algorithm requires calculations of the form $\mathcal{M}^{-1}a$ and $\mathcal{H}\mathcal{M}^{-1}a$. For $a^T = \begin{bmatrix} a_u^T & a_p^T \end{bmatrix}$ we see that

$$\mathcal{M}^{-1}a = \begin{bmatrix} b_u \\ b_p \end{bmatrix} = \begin{bmatrix} \hat{S}_A^{-1}(a_u + B^T\tilde{C}^{-1}a_p) \\ \tilde{C}^{-1}(Bb_u - a_p) \end{bmatrix}$$

and

$$\mathcal{H}\mathcal{M}^{-1}a = \begin{bmatrix} S_A b_u - (a_u + B^T\tilde{C}^{-1}a_p) \\ Bb_u - a_p - Cb_p \end{bmatrix}.$$

Notice that no calculations involving $\hat{S}_A$ are required. In addition, r_k is the residual of the original linear system at iteration k and can be used to assess convergence.

4 BDDC Preconditioner

A brief overview of the BDDC preconditioner is provided here for completeness. Additional details can be found in [4, 13, 14]. The domain of a finite element mesh is assumed to be decomposed into nonoverlapping substructures $\Omega_1, \ldots, \Omega_N$ so that each element is contained in exactly one substructure. The assembly of the substructure contributions to the linear system can be expressed as

$$\begin{bmatrix} A & B^T \\ B & -D \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix} = \sum_{i=1}^{N} \begin{bmatrix} R_i^T & P_i^T \end{bmatrix} \begin{bmatrix} A_i & B_i^T \\ B_i & -D_i \end{bmatrix} \begin{bmatrix} R_i \\ P_i \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix} \tag{15}$$

where each row of R_i and P_i contains exactly one nonzero entry of unity. Throughout this section several subscripted R matrices with exactly one nonzero entry of unity

in each row are used for bookkeeping purposes. For discontinuous pressure elements and compressible materials the matrices D and D_i are positive definite and block diagonal. Solving the second block of equations in (15) for p in terms of u and substituting the result back into the first block of equations leads to

$$Ku = f, \qquad p = D^{-1}Bu \tag{16}$$

where the displacement Schur complement K is given by

$$K = A + B^T D^{-1} B = \sum_{i=1}^{N} R_i^T K_i R_i \tag{17}$$

and

$$K_i = A_i + B_i^T D_i^{-1} B_i \,. \tag{18}$$

The coarse interpolation matrix Φ_i for Ω_i is obtained by solving the linear system

$$\begin{bmatrix} K_i & C_i^T \\ C_i & 0 \end{bmatrix} \begin{bmatrix} \Phi_i \\ \Lambda_i \end{bmatrix} = \begin{bmatrix} 0 \\ I \end{bmatrix} \tag{19}$$

where C_i is the constraint matrix for Ω_i and I is a suitably dimensioned identity matrix. A straightforward method to calculate Φ_i from (19) using solvers for sparse symmetric definite systems of equations is given in [4]. One option for constructing C_i is also described in [4].

Each row of the constraint matrix C_i is associated with a specific coarse degree of freedom (dof). Moreover, each coarse dof is associated with a particular set of nodes in Ω_i that appear in at least one other substructure. Let S_i denote the set of all such nodes. The set S_i is first partitioned into disjoint node sets $\mathcal{M}_{i1}, \ldots, \mathcal{M}_{iM_i}$ via the following equivalence relation. Two nodes are related if the substructures containing the two nodes are identical. In other words, each node of S_i is contained in exactly one node set, and all nodes in a given node set are contained in exactly the same set of substructures. Additional node sets called corners are used in [4] to facilitate the numerical implementation. Each corner is obtained by removing a node from one of the node sets described above. For notational convenience, we refer to $\{\mathcal{M}_{ij}\}_{j=1}^{M_i}$ as the set of all disjoint node sets for Ω_i including corners. Rows of the constraint matrix C_i associated with node set $\mathcal{M}_{ij}$ are given by $R_{ijr}C_i$. Similarly, columns of C_i associated with node set $\mathcal{M}_{ij}$ are given by $C_i R_{ijc}^T$. In this study all node sets are used in the substructure constraint equations.

Let u_{ci} denote a vector of coarse dofs for Ω_i. The dimension of u_{ci} equals the number of rows in the constraint matrix C_i. The vector u_{ci} is related to the global vector of coarse dofs u_c by

$$u_{ci} = R_{ci} u_c \,. \tag{20}$$

The coarse stiffness matrix of Ω_i is defined as

$$K_{ci} = \Phi_i^T K_i \Phi_i \tag{21}$$

and the assembled coarse stiffness matrix K_c is given by

$$K_c = \sum_{i=1}^{N} R_{ci}^T K_{ci} R_{ci} \,. \tag{22}$$

Consistent with (15), the vector of substructure displacement dofs u_i are related to u by

$$u_i = R_i u \,. \tag{23}$$

Let u_{Ii} denote a vector containing all displacement dofs in Ω_i that are not shared with any other substructures. The vector u_{Ii} is related to u_i by

$$u_{Ii} = R_{Ii} u_i \,. \tag{24}$$

In order to distribute residuals to the substructures, it is necessary to define weights for each substructure dof. In this study, the diagonal substructure weight matrix W_i is defined as

$$W_i = R_{Ii}^T R_{Ii} + \sum_{j=1}^{M_i} \alpha_{ij} R_{ijc}^T R_{ijc} \tag{25}$$

where

$$\alpha_{ij} = \text{trace}(R_{ijc} K_{ci} R_{ijc}^T)/\text{trace}(R_{ijc} R_{ci} K_c R_{ci}^T R_{ijc}^T) \tag{26}$$

and trace denotes the sum of diagonal entries. Notice that the weights of all dofs in a node set are identical. The substructure weight matrices form a partition of unity in the sense that

$$\sum_{i=1}^{N} R_i^T W_i R_i = I \,. \tag{27}$$

Given a residual vector r associated with the iterative solution of (16a), the preconditioned residual is obtained using the following algorithm.

1. Calculate the coarse grid correction v_1,

$$v_1 = \sum_{i=1}^{N} R_i^T W_i \Phi_i R_{ci} K_c^{-1} r_c \tag{28}$$

where

$$r_c = \sum_{i=1}^{N} R_{ci}^T \Phi_i^T W_i R_i r \,. \tag{29}$$

2. Calculate the substructure correction v_2,

$$v_2 = \sum_{i=1}^{N} R_i^T W_i z_i \tag{30}$$

where

$$\begin{bmatrix} K_i & C_i^T \\ C_i & 0 \end{bmatrix} \begin{bmatrix} z_i \\ \lambda_i \end{bmatrix} = \begin{bmatrix} W_i R_i r \\ 0 \end{bmatrix} . \tag{31}$$

3. Calculate the static condensation correction v_3,

$$v_3 = \sum_{i=1}^{N} R_i^T R_{Ii}^T (R_{Ii} K_i R_{Ii}^T)^{-1} R_{Ii} R_i r_1 \tag{32}$$

where

$$r_1 = r - K(v_1 + v_2) \,. \tag{33}$$

4. Calculate the preconditioned residual

$$M^{-1}r = v_1 + v_2 + v_3 \tag{34}$$

Residuals associated with displacement dofs in substructure interiors are removed prior to the first conjugate gradient iteration via a static condensation correction. These residuals then remain zero for all subsequent iterations.

5 BDDC Constraint Equations

In this section we show how to choose the constraint equations of BDDC so that it can be used effectively as a preconditioner for the primal Schur complement S_A. Recall at the end of Section 2 the goal of having a preconditioner that is insensitive to values of ϵ near zero. For problems with a divergence constraint like incompressible elasticity, this means that the performance of the preconditioner should not degrade as the norm of D in (15) approaches zero. Additional details and work related to this section can be found in [5] and [12].

The choice of constraints is guided by the goal to keep the volume change of each substructure relatively small in the presence of a divergence constraint. In particular, the volume change corresponding to a preconditioned residual should not be too large. Otherwise, the energy associated with the preconditioned residual will be excessively large and cause slow convergence of a Krylov iterative method.

Using the divergence theorem, the volume change of Ω_i resulting from u_i to first order is given by

$$\Delta V_i = \int_{\Omega_i} \nabla \cdot \mathbf{u} \, d\Omega = a_i^T u_i \tag{35}$$

where $\mathbf{u}$ is the finite element approximation of the displacement field. The vector a_i can be calculated in the same manner as the vector for a body force by summing element contributions to the divergence. All entries in a_i associated with nodes not on the boundary of Ω_i are zero.

The nodes in node set $\mathcal{M}_{ij}$ of substructure i are also contained in one or more node sets of other substructures. As such, define

$$\mathcal{N}_{ij} = \{(k,l) : \mathcal{M}_{kl} = \mathcal{M}_{ij}\}. \tag{36}$$

For notational convenience, assume that the rows of R_{ijc} are ordered such that $R_{ijc}u_i = R_{klc}u_k$ for all $(k,l) \in \mathcal{N}_{ij}$. Let E_{ij} denote the column concatenation of all vectors $R_{klc}a_k$ such that $(k,l) \in \mathcal{N}_{ij}$. Consider the singular value decomposition

$$\tilde{E}_{ij} = U_{ij} S_{ij} V_{ij}^T \tag{37}$$

where $\tilde{E}_{ij}$ is the matrix obtained by normalizing each column of E_{ij}. Assuming the singular values s_{ijm} on the diagonal of S_{ij} are in descending numerical order, let m_{ij} denote the largest value of m such that $s_{ikm}/s_{ij1} > tol$ where in this study $tol = 10^{-8}$. The singular values along with tol are used to determine a numerical rank of E_{ij}. Let F_{ij} denote the matrix obtained by normalizing each column of $(R_{ijr}C_i R_{ijc}^T)^T$ and define

$$\tilde{F}_{ij} = F_{ij} - \tilde{U}_{ij}\tilde{U}_{ij}^T F_{ij} = \bar{U}_{ij}\bar{S}_{ij}\bar{V}_{ij}^T \tag{38}$$

where $\tilde{U}$ contains the first m_{ij} columns of U_{ij}. The columns of $\tilde{U}$ are orthogonal and numerically span the range of E_{ij}. The singular values $\bar{s}_{ijm}$ on the diagonal of $\bar{S}_{ij}$ are assumed to be in descending numerical order and $\bar{m}_{ij}$ denotes the largest value of m such that $\bar{s}_{ijm} > tol$. Define

$$G_{ij} = \left[\tilde{U}_{ij} \; \hat{U}_{ij} \right] \tag{39}$$

where $\hat{U}_{ij}$ contains the first $\bar{m}_{ij}$ columns of $\bar{U}_{ij}$. The columns of $\hat{U}$ are orthogonal and numerically span the range of the projection of F_{ij} onto the orthogonal complement of $\tilde{U}_{ij}$. Thus, the columns of G_{ij} are orthogonal. Notice that G_{ij} contains a linearly independent set of vectors for the zero divergence constraints and the original BDDC constraints for node set $\mathcal{M}_{ij}$.

Finally, the original constraint matrix C_i is replaced by the row concatenation of the matrices $G_{ij}^T R_{ijc}$ for $j = 1, \ldots, M_i$. Use of the new substructure constraint matrices ensures that preconditioned residuals will not have excessively large values of volumetric energy. The final requirement needed to ensure good scalability with respect to the number of substructures is that the coarse stiffness matrix K_c be flexible enough to approximate well the low energy modes of K. This requirement is closely tied to an inf-sup condition, but is not analyzed here. Numerical results, however, indicate good scalability in this respect.

For 2D problems a node set consists either of a single isolated node called a corner or a group of nodes shared by exactly two substructures called a face. Furthermore, m_{ij}, the number of columns in $\tilde{U}_{ij}$, is at most two for a corner and one for a face. Similarly, for 3D problems m_{ij} is at most three for a corner and one for a face. The value of m_{ij} for the remaining 3D node sets, called edges here, depends on the mesh decomposition as well as the positions of nodes in the mesh. In any case, performance of the preconditioner should not degrade in the presence of nearly incompressible materials provided that all the columns of $\tilde{U}_{ij}$ are included in G_{ij}. Including columns of $\hat{U}_{ij}$ in G_{ij} as well will reduce condition numbers of the preconditioned equations, but is not necessary to avoid degraded performance for nearly incompressible materials.

Use of the modified constraints does not cause any difficulties when both nearly incompressible materials (e.g. rubber) and materials with smaller values of Poisson ratio (e.g. steel) are present. One can exclude the incompressibility constraint for substructures not containing nearly incompressible materials simply by setting all entries of a_i in (35) to zero. Doing so may lead to a slightly smaller coarse problem, but is not necessary.

6 Numerical Examples

In this section, (1) is solved to a relative residual tolerance of 10^{-6} using both right preconditioned GMRES [16] and preconditioned conjugate gradients (PCG) for an incompressible elasticity problem. For linear elasticity the shear modulus G and Lamé parameter λ for an isotropic material are related to the elastic modulus E and Poisson ratio ν by

$$G = \frac{E}{2(1+\nu)}, \qquad \lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}\,.$$

For incompressible problems λ is infinite with the result that $C = 0$ in (1). All the elasticity examples in this section use $G = 1$ and $\nu = 1/2$. We consider two different preconditioners for S_A in order to better understand the saddle point preconditioner. The first is based on a direct solver where $1.00001\hat{S}_A = S_A$ while the second is the BDDC preconditioner described in the previous two sections. Note that the leading constant 1.00001 is used to satisfy the assumption $\alpha_1 > 1$. The penalty matrix $\tilde{C}$ for the elasticity problems is chosen as the negative (2,2) block of the coefficient matrix in (1) for an identical problem with the same shear modulus but a value of ν less than 1/2.

Regarding assumption (8), we note that the BDDC preconditioner used for S_A has the attractive property that $\alpha_1 \geq 1$ and α_2 is mesh independent under certain additional assumptions [14]. For the conjugate gradient algorithm we scale the preconditioned residual associated with the primal Schur complement by 1.00001 to ensure that $\mathcal{H}$ is positive definite. We note, however, that such scaling was not necessary for any of the examples.

For purposes of comparison, we also present results for block diagonal and block triangular preconditioners for (1). Given the primal and dual residuals r_u and r_p, the preconditioned residuals z_u and z_p for the block diagonal preconditioner are given by

$$z_u = \hat{A}^{-1} r_u \quad \text{and} \quad z_p = M_p^{-1} r_p$$

where M_p is the dual mass matrix and either $\hat{A} = A$ (direct solver) or $\hat{A}$ is the BDDC preconditioner for A. Note that the shear modulus G was chosen as 1 to obtain proper scaling of z_p. Similarly, the preconditioned residuals for the block triangular preconditioner are given by

$$z_p = -M_p^{-1} r_p \quad \text{and} \quad z_u = M_A^{-1}(r_u - B^T z_p)\,.$$

We note that the majority of computations for the block preconditioners occur in forming and applying the BDDC preconditioner for A. Thus, the setup time and time for each iteration are very similar for the preconditioner of this study and the two block preconditioners.

The first example is for a 2D plane strain problem on a unit square with all displacement degrees of freedom (dofs) on the boundary constrained to zero. The entries of the right hand side vector b were chosen as uniformly distributed random numbers in the range from 0 to 1. For this simple geometry the finite element mesh consists of stable $Q_2 - P_1$ elements. This element uses biquadratic interpolation of displacement and discontinuous linear interpolation of pressure. In 2D the element has 9 nodes for displacement and 3 element pressure dofs. A description of the $Q_2 - P_1$ discontinuous pressure element can be found in [2].

Results are shown in Table 1 for the saddle point preconditioner (SPP) applied to a problem discretized by a 32 x 32 arrangement of square elements. Condition number estimates of the preconditioned equations are shown in parenthesis for the PCG results. The BDDC preconditioner is based on a regular decomposition of the mesh into 16 square substructures. The results shown in columns 2-5 are insensitive to changes in ν near the incompressible limit of 1/2. Notice that the use of a direct solver to precondition S_A results in very small numbers of iterations for values of ν near 1/2. The final two columns in Table 1 show results for BDDC contraint equations that are not modified to enforce zero divergence of each substructure. The condition number estimates grow in this case as ν appoaches 1/2.

Table 1. Iterations needed to solve incompressible 2D plane strain problem using the saddle point preconditioner. Results are shown for different values of ν used to define $\tilde{C}$. Results in parenthesis are condition number estimates from PCG. The S_A = no mod BDDC designation is for BDDC constraint equations that cannot enforce zero divergence of each substructure.

	$1.00001\hat{S}_A = S_A$		$\hat{S}_A$ = BDDC		$\hat{S}_A$ = no mod BDDC	
ν	GMRES	PCG	GMRES	PCG	GMRES	PCG
0.3	8	10 (4.8)	19	23 (16)	19	22 (16)
0.4	7	10 (2.4)	15	17 (7.2)	15	17 (7.1)
0.49	4	5 (1.1)	11	11 (3.0)	13	13 (3.6)
0.499	3	3 (1.01)	10	10 (2.7)	17	18 (8.5)
0.4999	3	3 (1.01)	9	9 (2.7)	23	28 (7.0e1)
0.49999	3	3 (1.01)	9	9 (2.6)	25	44 (6.9e2)

Table 2 shows results for a growing number of substructures with $H/h = 4$ where H and h are the substructure and element lengths, respectively. Very small growth in numbers of iterations with problem size is evident in the table for all the preconditioners. Notice that the iterations required by PCG either equal or are only slightly larger than those for GMRES. The primary advantage of PCG from a solver perspective is that storage of all search directions is not required as it is for GMRES. The SPP preconditioner is clearly superior to the two block preconditioners when a direct solver is used ($1.00001\hat{S}_A = S_A$ and $\hat{A} = A$). The performance of the SPP preconditioner compares very favorably with both of the block preconditioners when the BDDC preconditioner is used.

Table 2. Iterations needed to solve incompressible plane strain problems with increasing numbers of substructures (N) and $H/h = 4$. The value of ν used to define $\tilde{C}$ in the SPP preconditioner is 0.49999. Block diagonal and triangular preconditioners are denoted by M_d and M_t, respectively.

N	$1.00001\hat{S}_A = S_A$ and $\hat{A} = A$				$\hat{S}_A$ and $\hat{A}$ = BDDC			
	SPP		M_d	M_t	SPP		M_d	M_t
	GMRES	PCG	GMRES	GMRES	GMRES	PCG	GMRES	GMRES
4	3	3 (1.01)	17	9	6	6 (1.8)	26	16
16	3	3 (1.01)	17	9	8	8 (2.1)	30	20
36	3	3 (1.01)	17	9	9	9 (2.6)	35	23
64	3	3 (1.01)	17	9	9	10 (2.9)	38	26
100	3	3 (1.01)	17	9	10	10 (3.0)	40	28
144	3	3 (1.03)	17	9	10	10 (3.1)	42	29
196	3	3 (1.01)	17	9	10	11 (3.1)	45	30
256	3	3 (1.01)	17	9	10	11 (3.1)	47	30

References

1. O. Axelsson, *Preconditioning of indefinite problems by regularization*, SIAM J. Numerical Analysis, 16 (1979), pp. 58–69.
2. K. J. Bathe, *Finite Element Procedures*, Prentice Hall, Upper Saddle River, New Jersey, 1996.
3. J. H. Bramble and J. E. Pasciak, *A preconditioning technique for indefinite systems resulting from mixed approximations of elliptic problems*, Mathematics of Computation, 50 (1988), pp. 1–17.
4. C. R. Dohrmann, *A preconditioner for substructuring based on constrained energy minimization*, SIAM J. Scientific Computing, 25 (2003), pp. 246–258.
5. ———, *A substructuring preconditioner for nearly incompressible elasticity problems*, tech. rep., Sandia National Laboratories, SAND 2004-5393, 2004.
6. C. R. Dohrmann and R. B. Lehoucq, *A primal based penalty preconditioner for elliptic saddle point systems*, tech. rep., Sandia National Laboratories, SAND 2004-5964, 2004.
7. H. C. Elman and G. H. Golub, *Inexact and preconditioned Uzawa algorithms for saddle point problems*, SIAM J. Numerical Analysis, 31 (1994), pp. 1645–1661.
8. P. Goldfeld, L. F. Pavarino, and O. B. Widlund, *Balancing Neumann-Neumann preconditioners for mixed approximations of heterogeneous problems in linear elasticity*, Numerische Mathematik, 95 (2003), pp. 283–324.
9. A. Klawonn, *Block-triangular preconditioners for saddle point problems with a penalty term*, SIAM J. Scientific Computing, 19 (1998), pp. 172–184.
10. A. Klawonn and L. F. Pavarino, *Overlapping Schwarz methods for mixed linear elasticity and Stokes problems*, Computer Methods in Applied Mechanics and Engineering, 165 (1998), pp. 233–245.
11. J. Li, *A dual-primal feti method for incompressible Stokes equations*, Tech. Rep. 816, Courant Institute of mathematical Sciences, Department of Computer Sciences, 2001.
12. J. Li and O. B. Widlund, *BDDC algorithms for incompressible Stokes equations*, Tech. Rep. TR-861, Department of Computer Science, New York University, 2005.
13. J. Mandel and C. R. Dohrmann, *Convergence of a balancing domain decomposition by constraints and energy minimization*, Numerical Linear Algebra with Applications, 10 (2003), pp. 639–659.
14. J. Mandel, C. R. Dohrmann, and R. Tezaur, *An algebraic theory for primal and dual substructuring methods by constraints*, Applied Numerical Mathematics, (in press).
15. L. F. Pavarino and O. B. Widlund, *Balancing Neumann-Neumann methods for incompressible Stokes equations*, Comm. Pure Appl. Math., 55 (2002), pp. 302–335.
16. Y. Saad and M. H. Schultz, *Gmres: A generalized minimum residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 856–869.
17. D. Silvester and A. Wathen, *Fast iterative solution of stablized Stokes systems: part ii: using general block preconditioners*, SIAM J. Numerical Analysis, 31 (1994), pp. 1352–1367.