Electronic Transactions on Numerical Analysis.
Volume 15, pp. 94-105, 2003.

ISSN 1068-9613.

ETNA
Kent State University
etna@mcs.kent.edu

MULTILEVEL PRECONDITIONERS FOR LAGRANGE MULTIPLIERS IN DOMAIN IMBEDDING*

JANNE MARTIKAINEN†, TUOMO ROSSI†, AND JARI TOIVANEN†

Abstract. A domain imbedding method where the Dirichlet boundary conditions are treated using boundary supported Lagrange multipliers is considered. The discretization leads to a saddle-point problem which is solved iteratively by using either the PMINRES method with a block-diagonal preconditioner or the PCG method in an Uzawa type approach. In both cases, the preconditioning of the Schur complement related to Lagrange multipliers is based on a special sparse implementation of BPX/MDS method. The developed preconditioning technique is well-suited even for three-dimensional problems in domains with complicated shapes. Several numerical experiments for two-dimensional and three-dimensional problems demonstrate the efficiency and the applicability of the proposed method.

Key words. domain imbedding method, Lagrange multipliers, multilevel methods, preconditioning

AMS subject classifications. 65F10, 65N22, 65N55

1. Introduction. In this paper, we shall consider the efficient numerical solution of second-order elliptic Dirichlet boundary value problems by a domain imbedding method also known as a fictitious domain method. There exist several variants of these methods for such problems. One possibility is to use standard finite element approximation of the problem, and by suitably extending the arising coefficient matrix one gets an enlarged matrix for which it may be easier to construct an efficient solution procedure. We call them algebraic methods which include among others those considered in [12], [17]. However, they are typically restricted to quite a narrow class of problems.

More flexibility and better efficiency can be obtained by using mixed formulations with Lagrange multipliers to handle Dirichlet boundary value constraints. In such a way one can, for example, incorporate standard implementations of efficient multilevel techniques to the domain imbedding framework. This leads to an optimal preconditioning technique for problems posed on domains for which it is difficult to construct hierarchical sequences of meshes [20]. One possibility is to use volume distributed Lagrange multipliers. In [18], this approach has been applied to particulate flow problems governed by the Navier-Stokes equation. The optimal preconditioning of elliptic Dirichlet boundary value problems by the distributed Lagrange multiplier fictitious domain method was introduced in [19]. In [20], the method was extended to mixed boundary value problems, and its performance was demonstrated by a set of numerical experiments.

Another way is to use boundary supported Lagrange multipliers [2], [6]. In domain imbedding framework, they have been used in [14], [26], [32], [35], [38], for example. Typically, a $K^{1/2}$-type preconditioner [4], [7], [13], [15], [21], [36] is used for Lagrange multipliers. With this approach one can also obtain optimal computational cost, but the procedures for $K^{1/2}$-type preconditioners in the literature are restricted to two-dimensional problems. An alternative preconditioner for Lagrange multipliers in two-dimensional problems based on the $H(\mathrm{div})$-inner product was studied in [25].

Here, we consider a sparse multilevel preconditioner for boundary Lagrange multipliers. It is based on a multilevel nodal basis (BPX) preconditioner [8] or a multilevel diagonal scal-

*This work was supported by the Academy of Finland, grants #43066, #53588 and #66407. Received May 27, 2001. Accepted for publication September 12, 2001. Recommended by Panayot Vassilevski.

†Department of Mathematical Information Technology, University of Jyväskylä, P.O. Box 35 (Agora), FIN–40351 Jyväskylä, Finland (Janne.Martikainen@mit.jyu.fi, Tuomo.Rossi@mit.jyu.fi, Jari.Toivanen@mit.jyu.fi).

ing (MDS) preconditioner [43]. A sparse version of BPX was proposed as an edge preconditioner for domain decomposition problems in [42]. In an overlapping domain decomposition method, a preconditioner for Lagrange multipliers based on the sparse BPX was considered in [33]. A similar approach was used for the Mortar element method in [1]. A closely related idea to use a sparse version of an algebraic multigrid method was briefly considered in [31]. When a preconditioner is constructed for Lagrange multipliers, the action of the inverse of the sparse BPX or MDS is required. This can be accomplished by using an iterative procedure, such as the CG method or Chebyshev method. This approach was also suggested in [32].

There are two distinct advantages on a sparse multilevel preconditioner for Lagrange multipliers. They can be used for three-dimensional problems while other preconditioners for Lagrange multipliers seem to be restricted for two-dimensional problems. Secondly, boundaries can have quite complicated geometry and such a preconditioner is still easily applicable and optimal in quite many cases. As our numerical experiments seem to imply, the proposed domain imbedding method is well-suited for problems with complicated domains, excluding perhaps only domains with fractal boundaries. Special multigrid methods for such domains are rather involved and in many cases they are not optimal; see [9], [28], [29], for example.

The rest of this paper is organized as follows: In Section 2, we define the model saddle-point problem. In Section 2.1, we consider multilevel approximations for the inverse of the second-order elliptic operator. Some assumptions on Lagrange multipliers are given in Section 2.2. We use a block diagonal preconditioner and a Uzawa type algorithm in the solution of the model saddle-point problem in Section 3 and Section 4, respectively. In Section 5.1, we consider some of the possible preconditioners for primal variables. We introduce and study the sparse multilevel preconditioner in Section 5.2. Furthermore, the use of the CG method and the Chebyshev method in the inner iteration is considered. Finally, we perform numerical experiments using proposed methods with several two-dimensional and three-dimensional problems in Section 6. These experiments demonstrate the efficiency and the capabilities of the considered imbedding method.

2. Discrete problem. Let us consider the numerical solution of the second-order elliptic partial differential equation

$$-\nabla \cdot (E \nabla u) + eu = v \quad \text{in } \Omega \subset \mathbb{R}^d,\ d = 2, 3, \tag{2.1}$$

with a Dirichlet boundary condition on Γ which is $\partial\Omega$ or a part of it. We assume that the domain Ω can be extended to a larger polygonal domain Π over Γ. Furthermore, we assume that there is a quasiuniform hierarchical mesh for Π such that the mesh is conforming with $\partial\Omega$. Then, the equation (2.1) is extended to Π in such a way that everywhere in Π the matrix E is symmetric and its eigenvalues are bounded from below and above by positive constants and e is non-negative and bounded. We require that the extended problem has a Dirichlet boundary condition on part of the boundary $\partial\Pi$.

The extended problem is discretized with Lagrangian finite elements by using the hierarchical mesh in Π. The resulting $N \times N$ matrix A is symmetric and positive definite. In order to obtain an approximation for the solution of the original problem (2.1), we enforce the Dirichlet boundary condition on Γ by adding Lagrange multipliers to the discrete problem. This leads to the saddle-point problem

$$\mathcal{A}x = \begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} v \\ q \end{pmatrix} = b. \tag{2.2}$$

The possible values of the entries of the $M \times N$ constraint matrix B in (2.2) are zero and one. In those columns of B corresponding to Γ there is one entry having the value one while the other columns contain zeros and, in each row, there is one entry having the value one.

Remark 2.1. When introducing the boundary Lagrange multipliers to a continuous problem and then discretizing it with finite elements as in [2], [6], the matrix B does not have the previously defined form. After a simple change of variables the desired form can be obtained. With suitable permutation the matrix B has the block form $B = (0\ B_c)$ and the square matrix B_c is non-singular. Then, by replacing p with $r = B_c^{-1}p$ in (2.2) and by multiplying the second block row of (2.2) by B_c^{-1}, the problem will have the required form.

2.1. Multilevel approximation for A^{-1}. Under the previous assumptions there exists a multilevel nodal basis (BPX) preconditioner [8] or a multilevel diagonal scaling (MDS) preconditioner [43] denoted by C^{-1} which is spectrally equivalent with A^{-1}. The spectral equivalence means that there exist positive constants $\check{c}$ and $\hat{c}$ independent of N such that

$$\check{c}\, v^T C^{-1} v \leq v^T A^{-1} v \leq \hat{c}\, v^T C^{-1} v \qquad \forall v \in \mathbb{R}^N. \tag{2.3}$$

2.2. Assumptions on Lagrange multipliers. We assume for two-dimensional problems that there exists a curve formed by connecting the nodes corresponding to the Lagrange multipliers on Γ which has bounded length when N tends to infinity. Similarly, for three-dimensional problems, we assume that there exists a surface formed by connecting the nodes corresponding to the Lagrange multipliers on Γ which has bounded area when N tends to infinity. These assumptions exclude fractal boundaries and without them the preconditioner for Lagrange multipliers, which is introduced in Section 5.2, might be computationally too expensive.

Let the step size of the mesh which was used to obtain A be denoted by h. Since this mesh is quasiuniform, we have that h is the order of $N^{-1/d}$ for d-dimensional problems. From this and the previous assumption it follows that the number of Lagrange multipliers denoted by M is at most $\mathcal{O}(N^{1/2})$ for two-dimensional problems and $\mathcal{O}(N^{2/3})$ for three-dimensional problems.

With distributed Lagrange multipliers the number of multipliers M is $\mathcal{O}(N)$, but it is possible to reduce this to $\mathcal{O}(N^{(d-1)/d})$. This can be done mechanically in the following way: Let the matrix B^T be given by $B^T = (b_1\, b_2 \cdots b_M)$. Now, the ith Lagrange multiplier can be eliminated if it holds for i that $\mathrm{im}(Ab_i) \subset \mathrm{im}(B^T)$. The saddle-point problem (2.2) has the block form

$$\begin{pmatrix} A_{11} & A_{1\Gamma} & 0 & 0 & 0 \\ A_{\Gamma 1} & A_{\Gamma\Gamma} & A_{\Gamma 2} & I & 0 \\ 0 & A_{2\Gamma} & A_{22} & 0 & I \\ 0 & I & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 \end{pmatrix} \begin{pmatrix} u_1 \\ u_\Gamma \\ u_2 \\ p_\Gamma \\ p_2 \end{pmatrix} = \begin{pmatrix} v_1 \\ v_\Gamma \\ v_2 \\ q_\Gamma \\ q_2 \end{pmatrix}, \tag{2.4}$$

where the primal variables are split into three groups. The first group corresponds the variables without constraints. The second and third groups are related to the Lagrange multipliers which will be kept and eliminated, respectively. A simple manipulation of (2.4) yields an equivalent saddle-point problem

$$\begin{pmatrix} A_{11} & A_{1\Gamma} & 0 & 0 \\ A_{\Gamma 1} & A_{\Gamma\Gamma} & A_{\Gamma 2} & I \\ 0 & A_{2\Gamma} & A_{22} & 0 \\ 0 & I & 0 & 0 \end{pmatrix} \begin{pmatrix} u_1 \\ u_\Gamma \\ u_2 \\ p_\Gamma \end{pmatrix} = \begin{pmatrix} v_1 \\ v_\Gamma \\ A_{2\Gamma} q_\Gamma + A_{22} q_2 \\ q_\Gamma \end{pmatrix} \tag{2.5}$$

and the values of the eliminated Lagrange variables are $p_2 = v_2 - A_{2\Gamma} q_\Gamma - A_{22} q_2$.

ETNA
Kent State University
etna@mcs.kent.edu

3. Iterative solution of saddle-point problem. We use a block diagonal preconditioner $\mathcal{B}$ which has the form

$$\mathcal{B} = \begin{pmatrix} D & 0 \\ 0 & S \end{pmatrix} \tag{3.1}$$

in the iterative solution of the saddle-point problem (2.2). The choice of preconditioners D and S and their application are considered in Section 5. Let us assume for now that the preconditioners fulfill the conditions

$$\check{d}\, v^T D v \leq v^T A v \leq \hat{d}\, v^T D v \qquad \forall v \in \mathbb{R}^N \tag{3.2}$$

and

$$\check{c}\, q^T S q \leq q^T B A^{-1} B^T q \leq \hat{c}\, q^T S q \qquad \forall q \in \mathbb{R}^M \tag{3.3}$$

with some positive constants $\check{d}$, $\hat{d}$, $\check{c}$ and $\hat{c}$ independent of N. Actually, it is shown in Section 5.2 that the constants $\check{c}$ and $\hat{c}$ in (2.3) can be also used in (3.3). It is easy to see that $BA^{-1}B^T$ and $\mathcal{B}$ are symmetric and positive definite.

Now, a result by Yu. A. Kuznetsov, 1990 [30], gives information on the eigenvalues of $\mathcal{B}^{-1}\mathcal{A}$; see [27], [40]. It states that

$$\lambda(\mathcal{B}^{-1}\mathcal{A}) \in \left[(\hat{d} - \hat{a})/2,\ (\check{d} - \check{a})/2\right] \cup \left[\check{d},\ \hat{d}\right] \cup \left[(\check{d} + \check{a})/2,\ (\hat{d} + \hat{a})/2\right], \tag{3.4}$$

where the constants $\check{a}$ and $\hat{a}$ are given by $\check{a} = \sqrt{\check{d}^2 + 4\check{d}\check{c}}$ and $\hat{a} = \sqrt{\hat{d}^2 + 4\hat{d}\hat{c}}$. Hence, the eigenvalues of $\mathcal{B}^{-1}\mathcal{A}$ belong to intervals which are independent of N, since the constants defining them are also independent of N. Moreover, zero does not belong to any of the intervals and, thus, $\mathcal{A}$ and $\mathcal{B}$ can be said to be spectrally equivalent.

Since the saddle-point matrix $\mathcal{A}$ is indefinite, we must use an iterative method suitable for symmetric indefinite problems. For this purpose, we have chosen the minimal residual (MINRES) method and, especially, we employ a variant with a preconditioner. In our case the well-known result for the residual vectors in the MINRES method [23], [37] gives the error bound

$$\|r_k\|_{\mathcal{B}^{-1}} \leq 2 \left(\frac{\sqrt{(\hat{d}^2 - \hat{e}^2)/4} - \sqrt{\check{d}(\check{d} - \check{e})/2}}{\sqrt{(\hat{d}^2 - \hat{e}^2)/4} + \sqrt{\check{d}(\check{d} - \check{e})/2}} \right)^{\lfloor k/2 \rfloor} \|r_0\|_{\mathcal{B}^{-1}}, \tag{3.5}$$

where r_k is the kth residual vector.

4. Uzawa type algorithm for saddle-point problem. When the solution with the matrix A can be obtained cheaply with a good accuracy, it might be more efficient to solve the saddle-point problem (2.2) using a Uzawa type algorithm [16]. The idea is first to eliminate the primal variable u from (2.2) and get the formula $u = A^{-1}(v - B^T p)$. Using that elimination the system of linear equations

$$BA^{-1}B^T p = BA^{-1}v - q \tag{4.1}$$

is obtained. A suitable preconditioner for the coefficient matrix $BA^{-1}B^T$ is the preconditioner block S, the preconditioner for the Lagrange multipliers, considered in Section 5.2.

The solution of (4.1) can be computed using, for example, the preconditioned CG method. In each iteration one multiplication by the matrix $BA^{-1}B^T$ is performed and, thus, one linear problem with the matrix A must be solved with a rather good accuracy [16]. The fast

ETNA
Kent State University
etna@mcs.kent.edu

direct solvers are ideal in this kind of situation if they can be used to solve directly a problem with A. Furthermore, the right-hand side vectors belong to the subspace $\mathrm{im}(B^T)$ and the solution is only required in the same subspace. For this kind of problems the so-called partial solution technique [3], [34] which is a special implementation of a direct solver based on the separation of variables is very well suited. It requires $\mathcal{O}(N)$ and $\mathcal{O}(N \log N)$ floating point operations in the case of two-dimensional problems and three-dimensional problems, respectively.

5. Preconditioning. The application of the preconditioner (3.1) to a vector consists of the following two steps: The vector block corresponding to the primal variables is given to a multilevel method or to a fast direct solver as a right hand side and then the solution given by the method is the corresponding preconditioned vector block. Then, the vector block corresponding to the Lagrange multipliers is given to an iterative method, such as Conjugate Gradient or Chebyshev method, as a right hand side and the solution is taken as the preconditioned vector block. The matrix-vector product in the iterative method corresponds to a sparse implementation of BPX or MDS, considered in Subsection 5.2.

When the problem (4.1) is solved iteratively, only the second step of the preconditioner is used.

5.1. Preconditioner for primal variables. A natural choice for the preconditioner block D in (3.1) is to use the BPX or MDS algorithm as a preconditioner. However, in many cases it is more efficient to use some multiplicative multilevel method. They usually give better approximations for A^{-1} and, hence, they lead to a faster convergence of the iterative solution.

When the mesh underlying the discretization is topologically equivalent with an orthogonal mesh, fast direct methods such as FFT methods or cyclic reduction algorithms [41] can be used in the preconditioning. In some cases, these methods give the exact inverse for A.

We assume the preconditioner D for the primal variables to be spectrally equivalent with A. Thus, there exist positive constants $\check{d}$ and $\hat{d}$ independent of N such that the condition (3.2) is fulfilled.

5.2. Preconditioner for Lagrange multipliers. It is well-known that the preconditioner for the Lagrange multipliers, the block S in (3.1), should be close to the inverse of $BA^{-1}B^T$. Under the assumptions given in earlier sections, key observations are that $BA^{-1}B^T$ is close to $BC^{-1}B^T$ and that vectors can be multiplied cheaply by $BC^{-1}B^T$; see [32], [33], [42]. Thus, the approximation of the inverse of $BC^{-1}B^T$ obtained using an iterative method leads to a good preconditioner for the Lagrange multipliers, provided that it is computationally not too expensive when compared to other parts of the solution procedure. In the following, we study in more detail the ideas laid out here and show that we can obtain a good preconditioner which is optimal in terms of computational cost. When C^{-1} corresponds to the BPX preconditioner, the resulting preconditioner will be equivalent to that one suggested in [32].

Let us first denote

$$S = BC^{-1}B^T. \tag{5.1}$$

The inequalities (2.3) state the spectral equivalence of the matrices A^{-1} and C^{-1}. As a direct consequence of this we obtain the inequalities (3.3). Thus, $BA^{-1}B^T$ and S are spectrally equivalent with the same constants as A^{-1} and C^{-1}. Actually, the inequalities in (3.3) are not sharp, since the inequalities (2.3) could be made tighter in the subspace $\mathrm{im}(B^T) \subset \mathbb{R}^N$. ¿From (3.3) it also follows that

$$\mathrm{cond}\left(S^{-1}BA^{-1}B^T\right) \le \hat{c}/\check{c}, \tag{5.2}$$

where $\mathrm{cond}(\cdot)$ gives the spectral condition number.

Let us now consider the multiplication of a vector $p \in \mathbb{R}^M$ by S. Let us remark that it follows from our assumption that M is at most $\mathcal{O}(N^{(d-1)/d})$ for d-dimensional problems, $d = 2, 3$. Due to the structure of B the vector $B^T p$ is rather sparse, that is, it contains only M nonzero components. Thus, the multilevel preconditioner C^{-1} is applied to a vector which contains mostly zeros. For the multiplication of the vector $C^{-1}B^T p$ by B only the vector components corresponding to the subspace $\mathrm{im}(B^T) \subset \mathbb{R}^N$ are required. Since the dimension of $\mathrm{im}(B^T)$ is M, only M components are needed. Hence, we have a very special case in which the multilevel preconditioner is applied to a vector having M nonzero values and only the same M components of the resulting vector are required. For such a case, a sparse BPX/MDS operator can be easily created in an initialization phase by storing the grid transfer operations corresponding to nonzero components to sparse matrices.

Let us further study the computational cost of a special implementation taking advantage of the sparsity described above. We assume that the multilevel preconditioner is based on linear, bilinear or trilinear finite elements. By identifying the vectors v and $C^{-1}v$ with the associated finite element functions, the multilevel preconditioner C^{-1} is defined by

$$C^{-1}v = \sum_{l=1}^{L} \sum_{i=1}^{N_l} (v, \phi_i^l)_l \phi_i^l, \tag{5.3}$$

where ϕ_i^l is the ith basis function on the level l and $(\cdot,\cdot)_l$ is a scaled L_2-inner product. Furthermore, L is the number of levels in the hierarchical mesh and N_l is the number of basis functions on the level l. In our case, v is a linear combination of M different basis functions ϕ_i^L and we need to know only the coefficients for the same M basis functions for $C^{-1}v$.

For two-dimensional problems, the nonzero terms in the sum $\sum_{i=1}^{N_l} (v, \phi_i^l)_l \phi_i^l$ correspond to those basis functions whose support lies within a strip around the curve formed by connecting the nodes corresponding to the Lagrange multipliers. The width of this strip is $\mathcal{O}(2^{L-l}h)$ at the level l. The strip intersects with $\mathcal{O}(1/(2^{L-l}h))$ supports of the basis functions ϕ_i^l. Thus, the number of nonzero terms in the sum is $\mathcal{O}(N^{1/2}/2^{L-l})$ at the level l. By summing these up, we obtain that the total number of nonzero terms in (5.3) is $\mathcal{O}(N^{1/2})$ and this is also the computational cost for the sparse multilevel preconditioner for two-dimensional problems. In Figure 5.1, an example of sparsity patterns for all levels is shown. Similarly, we can obtain the computational cost $\mathcal{O}(N^{2/3})$ for three-dimensional problems. Hence, our sparse multilevel preconditioner requires $\mathcal{O}(M)$ operations. Since in our case $M \ll N$, this is a crucial observation in order to keep the computational cost, of the iterative solution of linear systems with the coefficient matrix S, affordable.

If the nodes related to the image subspace of B^T do not obey any surface-like pattern one can easily deduce a slightly weaker estimate for the cost of the sparse multilevel preconditioner. It is straightforward to see that the cost at each level is at most $\mathcal{O}(M)$. Since there are $\mathcal{O}(\log N)$ levels, the total cost is not higher than $\mathcal{O}(M \log N)$. In Section 6, we present the numerical results also for such settings of the problem.

5.3. CG for inner iteration. In the solution of linear systems with the symmetric and positive definite matrix S the CG method can be used. According to the well-known error bounds for CG, the number of iterations required to reduce the error in the norm induced by the coefficient matrix by the factor ε is of order $\sqrt{\kappa}\log\frac{1}{\varepsilon}$. Here, κ is the spectral condition number of S. The condition number for C^{-1} is $\mathcal{O}(N)$ for two-dimensional problems and $\mathcal{O}(N^{2/3})$ for three-dimensional problems. Due to properties of B we have the obvious inequality

$$\mathrm{cond}\,(S) \leq \mathrm{cond}\left(C^{-1}\right), \tag{5.4}$$

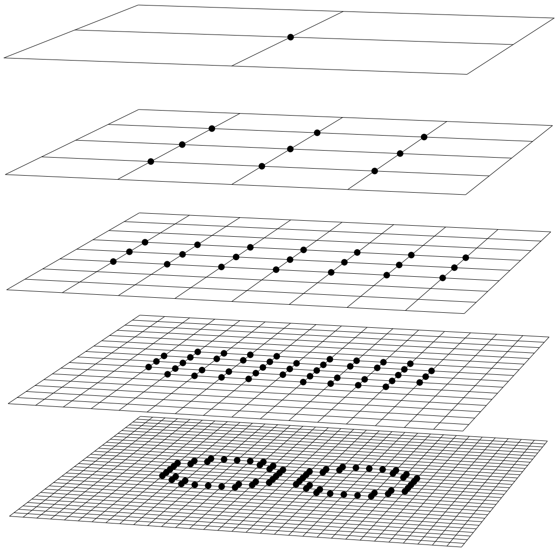

FIG. 5.1. *An example of sparsity patterns at different levels of BPX/MDS method*

which is not sharp, but it is still sufficient for our purposes. Thus, the number of required iterations is at most $\mathcal{O}(N^{1/d}\log\frac{1}{\varepsilon})$ for d-dimensional problems.

By taking the product of the cost of multiplication of a vector by S and the number of iterations required, we obtain that the total cost of the Lagrange preconditioner is at most $\mathcal{O}(N\log\frac{1}{\varepsilon})$ floating point operations for both two-dimensional and three-dimensional problems. The CG method gives an approximation for the solution of linear problem which depends nonlinearly on the right-hand side vector. Usually the methods used in the outer iteration can only cope with linear preconditioners. In order to diminish the nonlinearities caused by CG to the level which is tolerable for the outer iteration the error reduction factor ε must be sufficiently small. Due to that fact the required number of iterations might grow to be unnecessarily large. In the following section we try to alleviate this by considering the use of the Chebyshev method.

Remark 5.1. Often it holds that the condition number of S is of the same order as the square root of the condition number of C^{-1}; see [10]. In this case, the total cost of the Lagrange multiplier preconditioner reduces to $\mathcal{O}(N^{(2d-1)/2d}\log\frac{1}{\varepsilon})$ floating point operations for d-dimensional problems. Hence, it is possible to place a slightly larger proportion of mesh nodes than $\mathcal{O}(N^{(d-1)/d})$ onto the boundary in order to improve the accuracy of approximation and still keep the computational cost below $\mathcal{O}(N\log\frac{1}{\varepsilon})$.

5.4. Chebyshev method for inner iteration. In order to solve linear problems with the Chebyshev method, we must estimate the eigenvalues of the coefficient matrix. More precisely, we must compute the bounds $\lambda_{\min}$ and $\lambda_{\max}$ for the eigenvalues of S which satisfy the inequalities $\lambda_{\min} \le \lambda(S) \le \lambda_{\max}$. For example, this can be accomplished by performing a few Lanczos iterations and then by using the error bounds for the attained eigenvalue estimates [22].

The actual solution of the system of linear equations $Sp = q$ is performed in the following way: First, the initial values $p_0 = 0$, $\delta_1 = 2$ and $p_1 = \gamma q$ are set. Then, the kth

ETNA
Kent State University
etna@mcs.kent.edu

Chebyshev iterate p_k, $k = 2, 3, \ldots$, is given by the recurrence formula

$$\begin{aligned} \delta_k &= 4/(4 - \sigma^2 \delta_{k-1}), \\ p_k &= \delta_k(\gamma(q - Sp_{k-1}) + p_{k-1}) + (1 - \delta_k)p_{k-2}. \end{aligned} \tag{5.5}$$

The bounds $\lambda_{\min}$ and $\lambda_{\max}$ appear in the Chebyshev method via the constants δ and γ which are given by

$$\sigma = (\lambda_{\max} - \lambda_{\min})/(\lambda_{\max} + \lambda_{\min}) \quad \text{and} \quad \gamma = 2/(\lambda_{\max} + \lambda_{\min}). \tag{5.6}$$

According to the well-known error bounds for the Chebyshev method, the number of iterations required to reduce the error in the Euclidean norm by the factor ε is of order $\sqrt{\kappa} \log \frac{1}{\varepsilon}$. The upper estimate for the spectral condition number of S denoted by κ is given by $\lambda_{\max}/\lambda_{\min}$. Again the same analyses can be performed as in the case of the CG method. Hence, the cost of the Lagrange preconditioner is at most $\mathcal{O}(N \log \frac{1}{\varepsilon})$ floating point operations for both two-dimensional and three-dimensional problems. Unlike the CG method the Chebyshev method with a fixed number of iterations gives a linear operator. Thus, it is not necessary to compute the solution with a high precision in order to obtain a linear and spectrally equivalent preconditioner. With reasonably accurate eigenvalue bounds $\lambda_{\min}$ and $\lambda_{\max}$ it is often possible considerably reduce the number of inner iterations when the Chebyshev method is used instead of the CG method. On the other hand the implementation becomes more complicated, since the bounds for the eigenvalues must be computed.

6. Numerical experiments. In the experiments, we study the solution of a constrained Poisson problem in the unit square and in the unit cube. Thus, the matrix A is always the discretized Laplace operator in the unit square or in the unit cube. Discretizations are performed using linear finite elements and fully rectangular meshes with many different mesh step sizes. Note that we could use as well meshes which are locally fitted to the boundaries [5]. The sparse BPX is based on bilinear finite elements for two-dimensional problems and on trilinear finite elements for three-dimensional problems.

For two-dimensional problems, the diagonal block D in the preconditioner $\mathcal{B}$ is based on a multigrid method using bilinear interpolation operations and Gauss-Seidel smoother [11], [24]. A symmetric version is obtained by performing the smoothing in reverse order when the multigrid is moving from coarser to finer meshes. For three-dimensional problems, the diagonal block D in $\mathcal{B}$ is exactly A and it is based on a fast direct solver called PSCR method; see [39]. In the Uzawa type algorithm with the PCG method, the Euclidean norm of the residual vector is reduced by the factor 10^{-6}. When the saddle-point problem (2.2) is solved with the PMINRES method, the $\mathcal{B}^{-1}$-norm of the residual vector is reduced by the same factor.

In the first set of test problems, the domain Ω has two circular holes in two-dimensional problems and two spherical holes in three-dimensional problems. The diameter of holes is 0.375 and we have varied their distance between zero and 0.125 in the experiments.

In Table 6.1, we have reported results of numerical experiments for two-dimensional problems with two holes. The number of degrees of freedom is given by N, the number of Lagrange multipliers is M. The spectral condition number of S^{-1} defined by the sparse BPX is given by $\operatorname{cond} S$ column. The two last columns give the number of PMINRES iterations for saddle-point problems and outer PCG iterations for the Uzawa type algorithm. Each row corresponds to a sequence of problems with the distance between the holes growing from zero to 0.125; see Figure 6.1. This is also the reason why the number of Lagrange multipliers and the number of iterations are given as a range instead of single number.

ETNA
Kent State University
etna@mcs.kent.edu

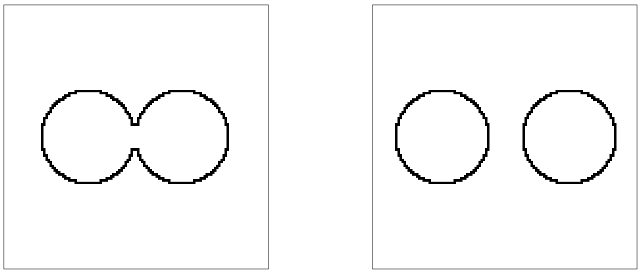

FIG. 6.1. *The holes for two-dimensional problems with the distance zero and 0.125*

TABLE 6.1
Results for two-dimensional problems with holes

N	M	cond S	PMINRES	Uzawa&PCG
31^2	58– 64	31.5– 39.4	23–24	9–10
63^2	146– 160	63.9– 86.6	23–25	9–10
127^2	330– 352	136.9– 193.2	23–26	9–10
255^2	702– 736	276.1– 399.1	24–26	9–10
511^2	1454–1504	554.9– 812.1	24–26	9–11
1023^2	2966–3040	1113.2–1639.2	24–26	9–11

For three-dimensional problems, the same results are given in Table 6.2. In Figure 6.2 and Figure 6.3, condition numbers of preconditioned Schur complement matrices $S^{-1}BA^{-1}B^T$ are shown with respect to different distances between the holes for two-dimensional and three-dimensional problems.

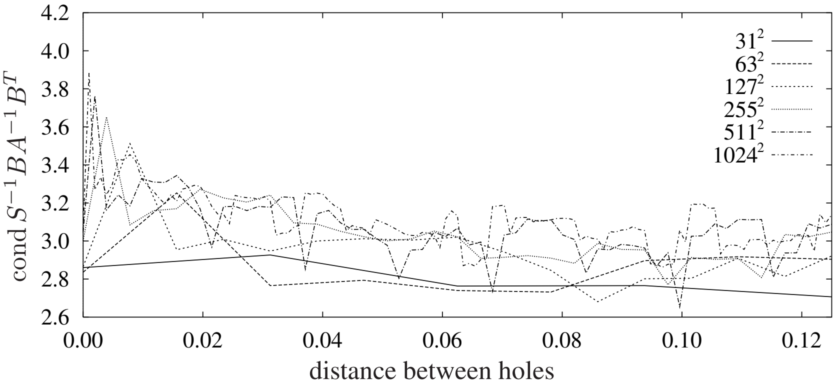

FIG. 6.2. *The condition number of preconditioned Schur complement matrices for two-dimensional problems*

The solution of a two-dimensional problem with the holes and $N = 1023^2$ requires less than two seconds with the PCG method when the multiplications by S^{-1} are approximated by 32 Chebyshev iterations and the multiplications by A^{-1} are based on a special fast direct partial solution technique [3], [34]. The multiplications by S^{-1} require about 30% of time. The computations have been performed on an HP J5600 workstation which has PA-8600 processor at 552 MHz and 1.5MB on-chip cache.

The second set of test problems is defined by randomly choosing the unknowns which have constraints. The number of constraints M is $N^{1/2}$ for two-dimensional problems and $N^{2/3}$ for three-dimensional problems. The nodes corresponding to the constrained unknowns

ETNA
Kent State University
etna@mcs.kent.edu

TABLE 6.2
Results for three-dimensional problems with holes

N	M	cond S	PMINRES	Uzawa&PCG
15^3	19– 20	4.4	17–19	10–11
31^3	206– 212	12.8–13.8	23	11–12
63^3	1307–1307	28.3–31.8	23	12–13

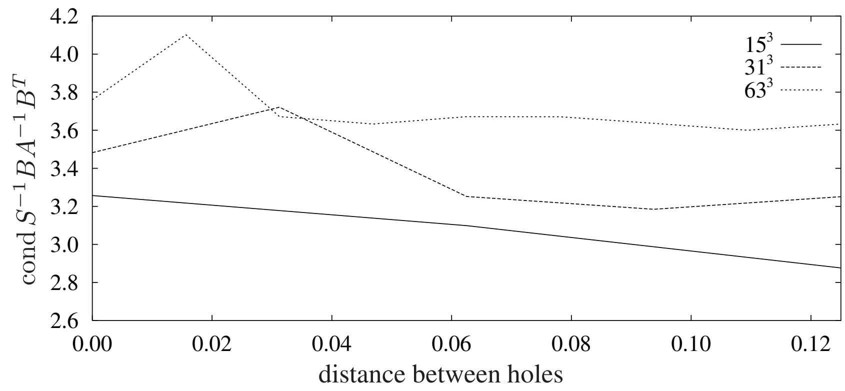

FIG. 6.3. *The condition number of preconditioned Schur complement matrices for three-dimensional problems*

are shown in Figure 6.4 for two-dimensional problems with $N = 31^2$ and $N = 127^2$. The results are reported in Table 6.3 and Table 6.4 in the same way as with the problems with two holes. The solution of the three-dimensional problem with a random domain and $N = 127^3$ requires about five minutes when the multiplications by S^{-1} are based on the CG method. The multiplications by S^{-1} require about 1% of time.

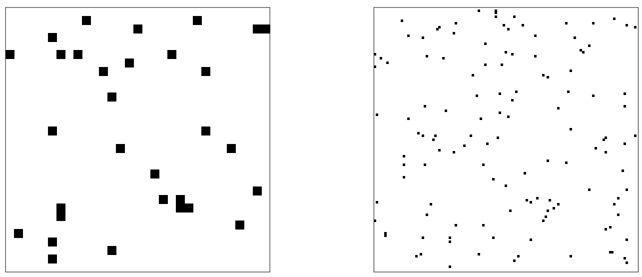

FIG. 6.4. *Randomly fixed unknowns for the two-dimensional problems with the sizes* 31^2 *and* 127^2

7. Conclusions. We have presented a domain imbedding method with Lagrange multipliers to enforce the Dirichlet boundary conditions. In the iterative solution, the primary variables are preconditioned with the standard domain imbedding approach using a multilevel method or a fast direct solver. A special implementation of BPX or MDS multilevel method is used in preconditioning the Lagrange multipliers. This special implementation takes into account the sparsity of the right hand side vector and the components of the solution which are actually needed in preconditioning. It has been shown that the whole preconditioner is spectrally optimal, the number of iterations needed to solve a problem with a fixed accuracy is bounded from above. It has also been shown that when using a multilevel method in preconditioning the primary variables, the computational complexity of the method can be optimal

TABLE 6.3
Results for two-dimensional problems with randomly fixed unknowns

N	M	cond S	PMINRES	Uzawa&PCG
31^2	31	7.8	22	9
63^2	63	12.9	23	9
127^2	127	25.9	22	9
255^2	255	49.1	22	10
511^2	511	93.8	22	10
1023^2	1023	201.4	24	10

TABLE 6.4
Results for three-dimensional problems with randomly fixed unknowns

N	M	cond S	PMINRES	Uzawa&PCG
15^3	15^2	3.2	17	8
31^3	31^2	5.4	17	8
63^3	63^2	9.9	17	8
127^3	127^2	18.5	17	8

with respect to the discretization mesh step size. The use of a fast direct solver results in a slightly weaker estimate. The numerical results show that the method is efficient in practice and the geometry of the problems can be complex.

Acknowledgments. The authors are grateful to Professor Yuri A. Kuznetsov for fruitful discussions about the considered solution procedures.

REFERENCES

[1] G. S. ABDOULAEV, Y. ACHDOU, YU. A. KUZNETSOV, AND C. PRUD'HOMME, *On a parallel implementation of the Mortar element method*, M2AN Math. Model. Numer. Anal., 33 (1999), pp. 245–259.

[2] I. BABUŠKA, *The finite element method with Lagrangian multipliers*, Numer. Math., 20 (1973), pp. 179–192.

[3] A. BANEGAS, *Fast Poisson solvers for problems with sparsity*, Math. Comp., 32 (1978), pp. 441–446.

[4] P. E. BJØRSTAD AND O. B. WIDLUND, *Solving elliptic problems on regions partitioned into substructures*, in Elliptic problem solvers, II (Monterey, Calif., 1983), Academic Press, Orlando, FL, 1984, pp. 245–255.

[5] C. BÖRGERS, *A triangulation algorithm for fast elliptic solvers based on domain imbedding*, SIAM J. Numer. Anal., 27 (1990), pp. 1187–1196.

[6] J. H. BRAMBLE, *The Lagrangian multiplier method for Dirichlet's problem*, Math. Comp., 37 (1981), pp. 1–11.

[7] J. H. BRAMBLE, J. E. PASCIAK, AND A. H. SCHATZ, *The construction of preconditioners for elliptic problems by substructuring. I*, Math. Comp., 47 (1986), pp. 103–134.

[8] J. H. BRAMBLE, J. E. PASCIAK, AND J. XU, *Parallel multilevel preconditioners*, Math. Comp., 55 (1990), pp. 1–22.

[9] A. BRANDT AND V. MIKULINSKY, *On recombining iterants in multigrid algorithms and problems with small islands*, SIAM J. Sci. Comput., 16 (1995), pp. 20–28.

[10] S. C. BRENNER, *The condition number of the Schur complement in domain decomposition*, Numer. Math., 83 (1999), pp. 187–203.

[11] W. L. BRIGGS, V. E. HENSON, AND S. F. MCCORMICK, *A multigrid tutorial*, SIAM, Philadelphia, PA, second ed., 2000.

[12] B. L. BUZBEE, F. W. DORR, J. A. GEORGE, AND G. H. GOLUB, *The direct solution of the discrete Poisson equation on irregular regions*, SIAM J. Numer. Anal., 8 (1971), pp. 722–736.

[13] T. F. CHAN, *Analysis of preconditioners for domain decomposition*, SIAM J. Numer. Anal., 24 (1987), pp. 382–390.

[14] Q. V. DINH, R. GLOWINSKI, J. HE, V. KWOCK, T.-W. PAN, AND J. PÉRIAUX, *Lagrange multiplier approach to fictitious domain methods: Application to fluid dynamics and electromagnetics*, in Fifth International Symposium on Domain Decomposition Methods for Partial Differential Equations, T. Chan, D. Keyes, G. Meurant, J. Scroggs, and R. Voigt, eds., Philadelphia, 1992, SIAM, pp. 151–194.

[15] M. DRYJA, *A capacitance matrix method for Dirichlet problem on polygonal region*, Numer. Math., 39 (1982), pp. 51–64.

[16] H. C. ELMAN AND G. H. GOLUB, *Inexact and preconditioned Uzawa algorithms for saddle point problems*, SIAM J. Numer. Anal., 31 (1994), pp. 1645–1661.
[17] S. A. FINOGENOV AND YU. A. KUZNETSOV, *Two–stage fictitious components method for solving the Diriclet boundary value problem*, Sov. J. Numer. Anal. Math. Modelling, 3 (1988), pp. 301–323.
[18] R. GLOWINSKI, T. I. HESLA, D. D. JOSEPH, T. W. PAN, AND J. PÉRIAUX, *A distributed Lagrange multiplier/fictitious domain method for flows around moving rigid bodies: Application to particulate flow*, Internat. J. Numer. Methods Fluids, 30 (1999), pp. 1043–1066.
[19] R. GLOWINSKI AND YU. KUZNETSOV, *On the solution of the Dirichlet problem for linear elliptic operators by a distributed Lagrande multiplier method*, C. R. Acad. Sci. Paris Sér. I Math., 327 (1998), pp. 693–698.
[20] R. GLOWINSKI, YU. KUZNETSOV, T. ROSSI, AND J. TOIVANEN, *A ficitious domain method with distributed lagrange multipliers*, in Numerical Mathematics and Advanced Applications Proceedings of the 3rd European Conference, P. Neittaanmäki, T. Tiihonen, and P. Tarvainen, eds., World Scientific, 2000, pp. 733–742.
[21] G. H. GOLUB AND D. MAYERS, *The use of preconditioning over irregular regions*, in Computing methods in applied sciences and engineering, VI (Versailles, 1983), North-Holland, Amsterdam, 1984, pp. 3–14.
[22] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, North Oxford Academic Publishers Ltd, 1986.
[23] A. GREENBAUM, *Iterative Methods for Solving Linear Systems*, vol. 17 of Frontiers in Applied Mathematics, SIAM, Philadelphia, PA, 1997.
[24] W. HACKBUSCH, *Multigrid methods and applications*, Springer-Verlag, Berlin, 1985.
[25] E. HAUG AND R. WINTHER, *A domain embedding preconditioner for the Lagrange multiplier system*, Math. Comp., 69 (2000), pp. 65–82.
[26] E. HEIKKOLA, T. ROSSI, P. TARVAINEN, AND YU. A. KUZNETSOV, *Efficient preconditioners based on fictitious domains for elliptic FE–problems with Lagrange multipliers*, in ENUMATH 97, Proceedings of the Second European Conference on Numerical Mathematics and Advanced Applications, H. G. Bock, F. Brezzi, R. Glowinski, G. Kanschat, Yu. A. Kuznetsov, J. Périaux, and R. Rannacher, eds., Singapore, 1998, World Scientific, pp. 646–661.
[27] YU. ILIASH, T. ROSSI, AND J. TOIVANEN, *Two iterative methods for solving the Stokes problem*, Tech. Rep. 2, Laboratory of Scientific Computing, Department of Mathematics, University of Jyväskylä, Jyväskylä, Finland, 1993.
[28] T. KÄRKKÄINEN AND J. TOIVANEN, *Building blocks for odd-even multigrid with applications to reduced systems*, J. Comput. Appl. Math., 131 (2001), pp. 15–33.
[29] R. KORNHUBER AND H. YSERENTANT, *Multilevel methods for elliptic problems on domains not resolved by the coarse grid*, in Domain decomposition methods in scientific and engineering computing (University Park, PA, 1993), Amer. Math. Soc., Providence, RI, 1994, pp. 49–60.
[30] YU. A. KUZNETSOV. Private communication, 1992.
[31] ———, *Efficient iterative solvers for elliptic finite element problems on nonmatching grids*, Russ. J. Numer. Anal. Math. Modelling, 10 (1995), pp. 187–211.
[32] ———, *Iterative analysis of finite element problems with Lagrange multipliers*, in Computational Science for the 21st Century, M.-O. Bristeau, G. Etgen, W. Fitzgibbon, J. L. Lions, J. Périaux, and M. F. Wheeler, eds., Wiley, Chichester, 1997, pp. 170–178.
[33] ———, *Overlapping domain decomposition with non–matching grids*, East–West J. Numer. Math., 6 (1998), pp. 299–308.
[34] YU. A. KUZNETSOV AND A. M. MATSOKIN, *On partial solution of systems of linear algebraic equations*, Sov. J. Numer. Anal. Math. Modelling, 4 (1989), pp. 453–467.
[35] R. A. E. MÄKINEN, T. ROSSI, AND J. TOIVANEN, *A moving mesh fictitious domain approach for shape optimization problems*, M2AN Math. Model. Numer. Anal., 34 (2000), pp. 31–45.
[36] M. MU AND J. R. RICE, *Preconditioning for domain decomposition through function approximation*, SIAM J. Sci. Comput., 15 (1994), pp. 1452–1466.
[37] C. C. PAIGE AND M. A. SAUNDERS, *Solution of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617–629.
[38] T. ROSSI, *Fictitious Domain Methods with Separable Preconditioners*, PhD thesis, Department of Mathematics, University of Jyväskylä, 1995.
[39] T. ROSSI AND J. TOIVANEN, *A parallel fast direct solver for block tridiagonal systems with separable matrices of arbitrary dimension*, SIAM J. Sci. Comput., 20 (1999), pp. 1778–1796.
[40] D. SILVESTER AND A. WATHEN, *Fast iterative solution of stabilised Stokes systems part II: using general block preconditioners*, SIAM J. Numer. Anal., 31 (1994), pp. 1352–1367.
[41] P. N. SWARZTRAUBER, *The methods of cyclic reduction and Fourier analysis and the FACR algorithm for the discrete solution of Poisson's equation on a rectangle*, SIAM Review, 19 (1977), pp. 490–501.
[42] C. H. TONG, T. F. CHAN, AND C. C. J. KUO, *A domain decomposition preconditioner based on a change to a multilevel nodal basis*, SIAM J. Sci. Stat. Comput., 12 (1991), pp. 1486–1495.
[43] X. ZHANG, *Multilevel Schwarz methods*, Numer. Math., 63 (1992), pp. 521–539.