A DOMAIN DECOMPOSITION PRECONDITIONER BASED ON A CHANGE TO A MULTILEVEL NODAL BASIS*

CHARLES H. TONG[†], TONY F. CHAN[‡], AND C.C. JAY KUO[§]

Abstract. A domain decomposition method based on a simple change of basis on the interfaces and vertices is presented. It is shown that this leads to an effective preconditioner compared to the ones previously considered, such as the preconditioner by Bramble, Pasciak, and Schatz (BPS) [*Math. Comp.*, 47 (1986), pp. 103–134], and the hierarchical basis domain decomposition (HBDD) preconditioner by Smith and Widlund [*SIAM J. Sci. Statist. Comput.*, 11 (1990), pp. 1212–1226]. This domain-decomposed preconditioner is based on Bramble, Pasciak, and Xu's multilevel nodal basis preconditioner [*Math. Comp.*, to appear]. It is shown that analytically this method and the HBDD method give the same order of condition number, namely, $O(\log^2(H/h))$ for problems with smooth coefficients. Numerically this method appears to be more effective with little additional cost and for the model Poisson problem, the condition numbers appear to be $O(1)$.

Key words. domain decomposition, hierarchical basis, multilevel nodal basis, preconditioned conjugate gradient methods, Schur complement

AMS(MOS) subject classifications. 65F10, 65N30

1. Introduction. We consider second-order, selfadjoint, uniformly elliptic partial differential equations on a two-dimensional polygonal domain Ω. The problems are solved numerically by using piecewise linear finite elements. The domain is first divided into nonoverlapping subregions Ω_i's, which are further divided into triangular finite elements. We use H to denote the diameter of a typical subregion and h to denote the diameter of its elements.

We begin with the linear system arising from a discretization of the problem and we first eliminate the variables interior to the subregions Ω_i. The resulting reduced system, the Schur complement, involves only the variables associated with Γ, the set of edges, and vertices of the subregions. This system is then solved by a preconditioned conjugate gradient method, where the preconditioner is constructed from certain problems associated with the interfaces $\Gamma_{ij} = \partial\Omega_i \cap \partial\Omega_j$ and vertices and a global coarse problem associated with the vertices.

Many preconditioners have been proposed for the subproblems associated with the edges Γ_{ij}. For example, the method by Bramble, Pasciak, and Schatz (BPS) [2] uses an operator similar to the square root of the Laplacian operator as the subproblem. Recently Smith and Widlund [8] proposed a computationally more efficient hybrid preconditioning method which involves only a simple change of basis (between nodal and hierarchical basis) with the unknowns on the edges Γ_{ij}. They show that the new method has a condition number which grows no faster than $C(1 + \log(H/h))^2$, which is comparable to that of the BPS method.

† Sandia National Laboratories, Livermore, California 94551-0969 (chtong@snll-arpagw.llnl.gov).

‡ Department of Mathematics, University of California, Los Angeles, California 90024 (chan@math.ucla.edu).

§ Department of Electrical Engineering-Systems, University of Southern California, Los Angeles, California 90089. This work was also supported by the University of Southern California Faculty Research and Innovation Fund.

The domain-decomposed preconditioner we consider in this paper is inspired by the work of Smith and Widlund [8]. In the same way that [8] uses the hierarchical basis on the edges to obtain a domain decomposition method, we use the multilevel nodal basis of Bramble, Pasciak, and Xu [3] applied to the reduced system on the interfaces (i.e., the edges *and the vertices*) to obtain our domain decomposition method. We call this preconditioner the multilevel nodal basis domain decomposition (MNBDD) preconditioner. We derive a proof, similar to the proof by Smith and Widlund [8], such that the condition number is bounded by $O(\log^2(H/h))$ for smooth coefficient problems. Numerical results are included to confirm the theoretical results. In particular, numerical experiments show that the condition number appears to be $O(1)$ for the model Poisson problem. We show that the MNBDD preconditioner also achieves convergence for variable and discontinuous problems in fewer iterations than the HBDD precondtioner, and with little extra cost.

2. The multilevel nodal basis algorithm and domain decomposition methods. In this section we provide the necessary background to define the MNBDD algorithm. We consider a second-order, selfadjoint, uniformly elliptic, bilinear form $a(u,v)$ on Ω with Dirichlet condition on $\partial\Omega$:

$$a(u,v) = (f,v) \quad \forall v \in H_0^1(\Omega), \quad u \in H_0^1(\Omega).$$

Let $V^H(\Omega)$ and $V^h(\Omega)$ be the spaces of continuous, piecewise linear functions, on the two triangulations, which vanish on $\partial\Omega$. We use elements which obey certain regularity assumptions, and obtain the following discrete variational problem:

$$a(u_h, v_h) = (f, v_h) \quad \forall v_h \in V^h(\Omega), \quad u_h \in V^h(\Omega).$$

By introducing the standard nodal basis $\{\phi_i\}$ for the space V^h, the above finite dimensional variational problem is reduced to a linear system:

$$Kx = b.$$

Here x is the vector of unknowns x_i, b is the vector of components (f, ϕ_i), and K is the stiffness matrix where $K_{ij} = a(\phi_i, \phi_j)$.
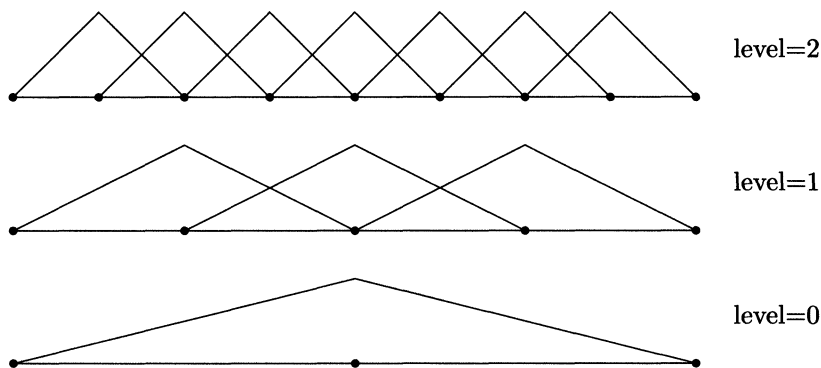

FIG. 1. *Multilevel nodal basis functions.*

2.1. The multilevel nodal basis algorithm. The multilevel nodal basis method [3] is given in terms of a set of nested sequence of finite element spaces,

$$V_0^h \subset V_1^h \subset \cdots \subset V_J^h \equiv V^h, \quad J = \log_2(H/h) \geq 1,$$

which are successive refinements by a factor of two of $V_0^h = V^H$. Here V_i^h is the set of piecewise linear finite element functions after i levels of refinement from the original coarse triangulation. In other words, V_i^h is the set of piecewise linear functions $\{\phi_i^j\}_{j=1}^{n_i}$ ($n_i = \dim V_i^h$) in $V^{2^{J-i}h}$ that satisfies

$$\phi_i^j(y_i^l) = \delta_{jl}, \quad \forall j, l = 1, \cdots, n_i,$$

where $\{y_i^l : l = 1, \cdots, n_i\}$ is the set of all interior nodal points of the triangulation on which V_i^h is defined. In short, $\{\phi_i^j\}_{j=1}^{n_i}$ is the standard nodal basis for the space V_i^h. Figure 1 shows the multilevel nodal basis functions in one dimension. The multilevel nodal basis preconditioner M of Bramble, Pasciak, and Xu [3], [6] applied to $v \in V^h$ takes the following form:

$$(1) \qquad\qquad M^{-1}v = \alpha A_0^{-1} Q_0 v + \sum_{l=1}^{J} \sum_{i=1}^{n_l} (v, \phi_l^i) \phi_l^i,$$

where the operator A_0 is a discretization of the elliptic operator $-\triangle$ on V_0^h and Q_0 is the standard orthogonal L^2 projection from V_J^h to V_0^h. The parameter α is not present in the original formulation of this preconditioner. Its inclusion here is inspired by the work of Smith and Widlund [8]. It can be shown that this modified version has the same order of condition number bound as the unmodified one (i.e., $\alpha = 1$), provided that α is independent of h and H.

The operator M^{-1} involves transformations between standard nodal basis and multilevel nodal basis (the second term in the above equation), as well as the solution of the problem corresponding to the original coarse triangulation (the first term in the equation). In matrix form, this preconditioner $\hat{M}$ can be written as:

$$\hat{M}^{-1} = GD^{-1}G^T,$$

where G and G^T represent the transformation matrices from multilevel nodal basis to nodal basis and vice versa, respectively; and $D^{-1} = \text{blockdiag}[I_{n_J}, I_{n_{J-1}}, \cdots, I_{n_1}, \alpha A_0^{-1}]$ solves the elliptic problem on the coarse level while leaving the other levels unchanged. It was proved in [3] that for smooth coefficient problems, the condition number of the preconditioned system $\kappa(\hat{M}^{-1}K) \leq CJ^2$, where C is a constant independent of h and H. In addition, this algorithm requires only $O(n_J)$ operations, where n_J is the dimension of the finite element space V^h.

The matrix G transforms the input vector from multilevel nodal basis to nodal basis. The dimension of the multilevel nodal basis is $m \equiv n_J + n_{J-1} + \cdots + n_1 + n_0$, whereas that of the standard nodal basis is $n = n_J$. Thus, G is a rectangular matrix of size $n \times m$ (which is unlike the square transformation matrices for the hierarchical basis). Let $v = \{v_l^i, i = 1, \cdots, n_l\}_{l=0}^{J} \in R^M$, where v_l^i is the value at the nodal point y_l^i corresponding to level l, then

$$Gv = \sum_{l=0}^{J} \sum_{i=1}^{n_l} v_l^i \phi_l^i,$$

where $\phi_l^i, i = 1, 2, \cdots, n_l$ is the set of basis functions in V_l^h.

The algorithm for G as applied to a vector v (of dimension m) is as follows:

Algorithm $G : v_i \rightarrow u, i = 0, \cdots, J$
    for $l = 0, \cdots, J - 1$
        $v_{l+1} := v_{l+1} + I_l^{l+1} v_l$
    end for
    $u = v_J$
end G

Here the I_l^{l+1} matrix is obtained from the choice of ϕ_i's. On a two-dimensional uniform domain using triangular elements, it corresponds to a seven-point interpolation operator.

2.2. Domain decomposition methods. Domain decomposition methods generally split the space V^h into $N + 1$ subspaces

$$V^h = V_{har}^h \oplus V_0^h(\Omega_1) \oplus \cdots \oplus V_0^h(\Omega_N).$$

For each subregion Ω_i, we thus have a subspace $V_0^h(\Omega_i) = V^h \cap H_0^1(\Omega_i)$. The elements of V_{har}^h are piecewise, discrete harmonic functions, i.e., they are orthogonal, in the sense of the bilinear form $a(\cdot, \cdot)$, to all the other subspaces.

First we partition the stiffness matrix K and vector x into those corresponding to the interior of the subregions and the edges and vertices. We then have

$$Kx = \begin{bmatrix} K_I & K_{IB} \\ K_{IB}^T & K_B \end{bmatrix} \begin{bmatrix} x_I \\ x_B \end{bmatrix} = \begin{bmatrix} b_I \\ b_B \end{bmatrix}.$$

If we apply block Gaussian elimination to eliminate the interior points, we obtain the following reduced system or Schur complement for the edges and interfaces

$$S_B x_B = (K_B - K_{IB}^T K_I^{-1} K_{IB}) x_B = b_B - K_{IB}^T K_I^{-1} b_I = \hat{b}_B.$$

The Schur complement matrix S_B is generally dense. However, it is not necessary to generate this matrix since, in the conjugate gradient iteration, this matrix is needed only in terms of matrix-vector products which can be computed by solving each subregion once and collecting the solution on the interfaces and vertices.

2.3. Multilevel nodal basis domain decomposition preconditioner. In this section, we combine the ideas from previous sections to derive a new domain decomposition algorithm. The symmetric form of the preconditioned system using the multilevel nodal basis preconditioner can be written as:

$$D^{-\frac{1}{2}} G^T K G D^{-\frac{1}{2}} \tilde{x} = \tilde{b},$$

where $G D^{-\frac{1}{2}} \tilde{x} = x$ and $\tilde{b} = D^{-\frac{1}{2}} G^T b$. Let us partition the unknowns corresponding to the multilevel nodal basis $\tilde{x}$ into those on the subregion interior $\tilde{x}_I$ and those on the interface $\tilde{x}_B$ and we eliminate the subregion interior variables $\tilde{x}_I$, we obtain the reduced system:

$$\tilde{S}_B \tilde{x}_B = \tilde{b}_B,$$

where $\tilde{S}_B$ is the Schur complement of $D^{-\frac{1}{2}} G^T K G D^{-\frac{1}{2}}$ after eliminating x_I. Here we can also decompose G according to the interior and interface unknowns so that

$$G = \begin{bmatrix} G_I & G_{IB} \\ G_{BI} & G_B \end{bmatrix},$$

where G_I and G_B are the transformation (rectangular) matrices involving subregion interior points and interface points, respectively.

The above formulation requires many arithmetic operations when the Schur complement is applied to a vector during the conjugate gradient iterations. By using the following lemmas, the above system can be reduced to a simpler form.

LEMMA 2.1. *G represents a change of basis which leaves the space of variables on Γ invariant (i.e., $G_{BI} = 0$).*

Proof. G_{BI} represents the contribution of the multilevel nodal bases in the subdomain interior to the nodal basis on the interfaces during the transformation. Recall from previous sections that

$$(2) \qquad u \equiv Gv = \sum_{l=0}^{J} \sum_{i=1}^{n_J} v_l^i \phi_l^i,$$

where $\{\phi_l^i, i = 1, 2, \cdots, n_l\}$ is the set of basis functions in V_l^h, and $v_l^i, i = 1, \cdots, n_l$ is the set of values defined on the nodal points x_l^i. Let $u = (u_I, u_B)^T$ and $v = (v_I, v_B)^T$ be the partitionings according to subdomain interior and interfaces, where $u \in R^n$ and $v \in R^m$. If we evaluate the above expression at node y_j^i on the interface Γ, we obtain

$$u_B^i = (G_B v_B)^i + (G_{BI} v_I)^i = \sum_{l=0}^{J} \left\{ \sum_{j \in \Gamma} (v_B)_l^j (\phi_l^j)_i + \sum_{j \notin \Gamma} (v_I)_l^j (\phi_l^j)_i \right\},$$

where $(\phi_l^j)_i$ is the value of the basis function at node j evaluated at node i on level l. It can be verified that

$$(\phi_l^j)_i = 0 \quad \forall j \in \Omega_k \text{ and } i \in \Gamma,$$

since all multilevel nodal basis functions at the nodes interior to the subregions vanish on the interface. In other words, the second term in the summation of (2) is identically equal to 0. Thus $G_{BI} = 0$ and $u^i = (G_B v_B)^i$. $\quad\square$

The following lemma is based on a similar one from Smith and Widlund's paper [8]. The proof can be obtained by a straightforward matrix manipulation.

LEMMA 2.2. *Let $G_{BI} = 0$ and $D = \text{blockdiag}(D_I, D_B)$, then*

$$\tilde{S}_B = D_B^{-\frac{1}{2}} G_B^T S_B G_B D_B^{-\frac{1}{2}}.$$

The above two lemmas imply that if we first eliminate the interior variables and then transform to the multilevel nodal basis, we will be solving the same linear system as before (transformation to multilevel nodal basis and then do the elimination). As a result, we are solving the following simpler and smaller system:

$$D_B^{-\frac{1}{2}} G_B^T S_B G_B D_B^{-\frac{1}{2}} \tilde{x}_B = \tilde{b}_B,$$

which is equivalent to solving the Schur complement system $S_B x_B = b_B$ with the preconditioner $M^{-1} = G_B D_B^{-1} G_B^T$. We call the preconditioner using this new formulation the multilevel nodal basis domain decomposition (MNBDD) preconditioner. This method offers several possible advantages over the standard multilevel nodal basis method by Bramble, Pasciak, and Xu [3]. The conjugate gradient iteration is

carried out over a much smaller set of unknowns and we will show that the condition number is smaller. The solution of the subproblems on the interfaces is easily parallelizable since they are independent. One possible drawback, however, is that now it is necessary to solve each subregion exactly in each iteration which adds more computational overhead.

Here the algorithm for G_B is similar to the one for G shown before, except now the basis functions used in the evaluation are restricted only to those on the interfaces. The operation of G_B on a vector v_B is defined by:

$$G_B v_B = \sum_{j=0}^{J} \sum_{i=1}^{m_j} (v_B)_j^i \phi_j^i,$$

where $\phi_j^i, i = 1, 2, \cdots m_j$ is the set of basis functions on Γ.

The MNBDD algorithm can be summarized as follows:

Algorithm MNBDD : input $= r$, output $= z = M_B^{-1} r$
   Perform partial change of basis to the MN basis $v = G_B^T r$
   solve the coarse grid problem $y = D_B^{-1} v$
   Perform change of basis back to the nodal basis $z = G_B y$
end MNBDD

We need one more lemma from [8].

LEMMA 2.3. *Let K be symmetric and positive definite. Then the condition numbers of K and its Schur complement satisfy*

$$\kappa(Schur(K)) \leq \kappa(K).$$

Using Lemma 2.3, as well as the condition number bounds from [3], we arrive at the following main theorem.

THEOREM 2.4. $\kappa(M_B^{-1} S_B) \leq O(\log^2(H/h))$ *for smooth coefficient problems.*

Proof. By using Lemma 2.1 and 2.2 we obtain

$$\tilde{S}_B = D_B^{-\frac{1}{2}} G_B^T S_B G_B D_B^{-\frac{1}{2}}.$$

By using Lemma 2.3, we obtain

$$\kappa(D_B^{-\frac{1}{2}} G_B^T S_B G_B D_B^{-\frac{1}{2}}) = \kappa(\tilde{S}_B)$$
$$\leq \kappa(D^{-\frac{1}{2}} G^T K G D^{-\frac{1}{2}}),$$

which is bounded by $O(\log^2(H/h))$, see [3]. □

It is also proved in [3] that the condition number bound in Theorem 2.4 depends also on the elliptic regularity of the problem. For example, for smooth coefficient problems on convex polygonal domains, the condition number is bounded by $O(\log(H/h))$; and for certain discontinuous problems, the condition number is bounded by $O(\log^3(H/h))$.

The hierarchical basis domain-decomposed (HBDD) [8] algorithm is similar to the BPS algorithm by Bramble, Pasciak, and Schatz [2]. The only difference is that the HBDD uses the hierarchical basis preconditioner for the edges while the BPS uses variants of Dryja's preconditioner [5]. The MNBDD algorithm, however, has

one important difference; namely, in addition to the use of multilevel nodal basis as preconditioners on the edges, the MNBDD algorithm also implements the multilevel nodal basis on the vertices. This introduces some redundancy on the vertices. This redundancy may be the reason for its improved performance (see next section).

3. Numerical results.

3.1. Two-subdomain example. For the two-subdomain case, since there is one edge and no vertex, it is not necessary to solve a coarse problem. We use the two-subdomain example of [8] with a different right-hand side, and we compare our results with those reported in [8]. We also include results from using the Dryja ($l_0^{1/2}$) [5] preconditioner. We use the domain $\bar{\Omega} = \bar{\Omega}_1 \cup \bar{\Omega}_2$, where $\bar{\Omega}_1$ and $\bar{\Omega}_2$ are unit squares aligned along an edge $\Gamma = \bar{\Omega}_1 \cap \bar{\Omega}_2$. We use the standard uniform mesh, and the usual five-point discretization for the Laplacian and the iteration counts are listed in Table 1 (where n is the number of unknowns on the interface). The right-hand side is such that the solution is $x(x-1)y(y-1)$ and the stopping criterion is when the relative 2-norm of the residual falls below 10^{-6}. The initial guess used is $u^{(0)} = 1.0$. We observe that while the iteration count for the HBDD continues to grow with larger n, the other preconditioners seem to be bounded independent of n. While the iteration counts using the HBDD preconditioners grow with n, we see that the iteration counts using MNBDD seem to gradually level off. Overall, we see that the MNBDD preconditioner performs relatively well compared to the others.

TABLE 1
Iteration count versus n.

n	No precond	Dryja	MNBDD	HBDD
8	4	4	4	4
16	8	6	7	7
32	16	6	9	8
64	27	6	9	10
128	39	6	9	12

3.2. Many-subdomain example. Next we consider the case of many subdomains. The unit square Ω is subdivided uniformly into $k \times k$ square subdomains and the same model problem is solved using uniform meshes. We compare our results with a set of experiments reported in [8]. We also compare the condition numbers as well as iteration counts between our method, Smith and Widlund's method, and Bramble, Pasciak, and Schatz's method (BPS).

3.2.1. Two-dimensional Poisson equation. A two-dimensional Poisson equation is solved in this case. The right-hand side f is constructed such that the solution is $u(x, y) = x(x-1)y(y-1)$ and the stopping criterion is 10^{-5}. Again, the initial guess is $u^{(0)} = 1.0$. The results are shown in Table 2.

Our first observation is that our condition number results for the HBDD method agree very well with those reported in [8]. We also observe that the condition numbers using the MNBDD method are much lower than the BPS and HBDD methods. The condition numbers grow very slowly with n; this is not the case with the BPS and HBDD methods.

TABLE 2
Condition numbers and iteration counts for the Poisson problem.

Grid	Number of subdomains	HBDD($\alpha = 3.6$)		BPS		MNBDD	
		κ	iter	κ	iter	κ	iter
32x32	2x2	9.62	11	11.85	11	2.24	7
32x32	4x4	7.96	11	8.75	14	2.19	8
32x32	8x8	5.30	10	6.08	12	2.10	7
64x64	2x2	12.68	13	16.47	12	2.32	8
64x64	4x4	11.84	13	13.03	15	2.28	8
64x64	8x8	8.52	12	9.79	15	2.21	8
64x64	16x16	5.41	10	6.32	13	2.11	7
128x128	4x4	16.49	15	17.92	18	2.35	8
128x128	8x8	12.54	15	14.18	16	2.35	8
128x128	16x16	8.69	13	10.21	15	2.24	8
128x128	32x32	5.42	10	6.36	13	2.11	7
256x256	4x4	21.90	17	23.45	19	2.39	8
256x256	8x8	17.30	17	19.33	18	2.43	8
256x256	16x16	12.72	15	14.79	17	2.36	8
256x256	32x32	8.69	13	10.27	15	2.24	8
256x256	64x64	5.37	10	6.37	12	2.09	7

3.2.2. Two-dimensional variable coefficient problem.

Next, we use the following variable coefficient boundary value problem:

$$-\frac{\partial}{\partial x}\left(e^{-xy}\frac{\partial u}{\partial x}\right) - \frac{\partial}{\partial y}\left(e^{xy}\frac{\partial u}{\partial y}\right) = f(x,y) \quad \text{in} \quad \Omega = [0,1]^2.$$

In order to account for the variation of coefficients in this case, we use a simple diagonal scaling in the HBDD, MNBDD, and BPS preconditioners. The right-hand side is constructed such that the solution is $u(x,y) = x(x-1)y(y-1)$, and the stopping criterion is achieved when the relative 2-norm of the residual drops below 10^{-5}. The number of iterations are shown in Table 3. Here we observe a small deterioration in convergence rates. With the use of strongly varying coefficients, all the methods used show rapidly decaying convergence rates.

TABLE 3
Iteration counts for variable coefficient case.

Grid	No. subdomains	HBDD	BPS	MNBDD
32x32	2x2	14	13	9
32x32	4x4	16	17	10
32x32	8x8	15	15	11
64x64	4x4	19	19	10
64x64	8x8	18	19	11
64x64	16x16	16	16	12
128x128	4x4	21	21	11
128x128	8x8	21	21	12
128x128	16x16	19	20	12
128x128	32x32	16	17	13

3.2.3. Two-dimensional discontinuous coefficient problem.

The discontinuous coefficient example used in this experiment is as follows:

$$-\frac{\partial}{\partial x}\left(\mu(x,y)\frac{\partial u}{\partial x}\right) - \frac{\partial}{\partial y}\left(\mu(x,y)\frac{\partial u}{\partial y}\right) = f(x,y) \quad \text{in} \quad \Omega = [0,1]^2,$$

where μ is a constant within each subdomain. In the following, we show the numerical results for the 16-subdomain case with μ in each subdomain as shown in Fig. 2.

10^{-1}	10^3	10^{-2}	10^2
10^{-2}	10^2	10^{-3}	10
10^{-3}	10	10^{-4}	1
10^{-4}	1	10^4	10^{-1}

FIG. 2. *Discontinuous coefficients used.*

Again, diagonal scaling is used to account for the variation of coefficients in all preconditioners used. The right-hand side is constructed such that the solution is $u(x,y) = x(x-1)y(y-1)$, and the stopping criterion is the same as in the previous cases. The number of iterations are shown in Table 4, which shows that all the methods used are effective in solving the above discontinuous coefficient problem, with the MNBDD preconditioner slightly better.

TABLE 4
Iteration counts for discontinuous coefficient case.

Grid	No. subdomains	HBDD	BPS	MNBDD
32x32	4x4	15	14	12
64x64	4x4	18	16	15
128x128	4x4	22	18	18
256x256	4x4	25	20	21

Through the numerical experiments, we have shown that the MNBDD precondi-tioner offers good convergence rates (better than BPS and HBDD for the problems used in our experiment), as well as low computational cost ($O(n)$ for MNBDD and HBDD and $O(n \log n)$ for BPS).

REFERENCES

[1] R. E. BANK, T. F. DUPONT, AND H. YSERENTANT, *The hierarchical basis multigrid method*, Numer. Math., 52 (1988), pp. 427–458.
[2] 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.
[3] J. H. BRAMBLE, J. E. PASCIAK, AND J. XU, *Parallel multilevel preconditioners*, Math. Comp., 55 (1990), pp. 1–22.
[4] T. F. CHAN, R. GLOWINSKI, J. PERIAUX, AND O. B. WIDLUND, *Proceedings of the Third In-ternational Symposium on Domain Decomposition Methods for Partial Differential Equa-tions*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1989.
[5] M. DRYJA, *A capacitance matrix method for Dirichlet problem on polygon region*, Numer. Math., 39 (1982), pp. 51–54.
[6] C.-C. J. KUO, T. F. CHAN, AND C. H. TONG, *Multilevel filtering elliptic preconditioners*, UCLA CAM Report 89-23, August 1989; SIAM J. Matrix Anal. Appl., 11 (1990), pp. 403–429.
[7] B. F. SMITH, *An optimal domain decomposition preconditioner for the finite element solution of linear elasticity problems*, preprint.

[8] B. SMITH AND O. WIDLUND, *A domain decomposition algorithm using a hierarchical basis*, SIAM J. Sci. Statist. Comput., 11 (1990), pp. 1212–1226.

[9] H. YSERENTANT, *On the multi-level splitting of finite element spaces*, Numer. Math., 49 (1986), pp. 379–412.

[10] J. XU AND J. QIN, *On some multilevel preconditioners*, SIAM J. Sci. Statist. Comput., submitted.

[11] J. XU, *Theory of multilevel methods*, Ph.D. thesis, Department of Mathematics, Cornell University, NY, 1989.