We study domain decomposition techniques for the solution of partial differential equations on a domain divided into several subdomains. These techniques have special applications in the solution of elliptic problems on irregular domains and parallel computation. A unifying feature of these techniques is the use of preconditioned conjugate gradient method in solving for the unknowns on the interfaces of the subdomains, or in some cases, on the whole domain. Since each iteration involves solving problems on each subdomain, it is essential to keep the number of iterations low. For this reason, much effort has been devoted recently to the construction of good preconditioners for the conjugate gradient methods. In this paper, we survey the most common preconditioners that have appeared in the literature, including a new class that we have developed recently. One of our objectives is to illuminate the relationships among these preconditioners.

A Survey of Preconditioners for Domain Decomposition

Tony F. Chan and Diana C. Resasco †

Research Report YALEU/DCS/RR-414
September 1985

The authors were supported in part by the Department of Energy under contract DE-AC02-81ER10996, by the Army Research Office under contract DAAG-83-0177, and by a BID-CONICET fellowship from Argentina.

Keywords: *domain decomposition, substructuring, capacitance matrix, elliptic partial differential equations, preconditioning, parallel computation*

†Dept. of Computer Science, Yale Univ., Box 2158, Yale Station, New Haven, Conn. 06520.

This paper is based on an invited talk at the IV Coloquio de Matemáticas del CINVESTAV, Workshop in Numerical Analysis and its applications, Taxco, Mexico, 18-24 Aug. 1985.

1. Introduction

We consider the problem of solving an elliptic partial differential equation on a domain Ω that is broken up into subregions Ω_i. By using *domain decomposition* or *substructuring* techniques, the problem reduces to separately solving approximate problems in the subdomains and updating the solution at the interfaces between two or more subregions. There are several reasons why these techniques might be attractive:

1. Special solution techniques, like fast direct solvers, might exist to solve the problems on the subdomains that cannot be applied efficiently to the entire domain. This is often the case, for example, when the subdomains Ω_i have very regular geometry, but Ω does not.
2. The equations in the different subdomains might have different parameters or even be of different nature, in which case the idea of substructuring comes very naturally.
3. The idea is attractive for parallel processing, since the problem can be decoupled into independent subproblems and the communication needed is limited to the boundaries of the subdomains.

For the class of domain decomposition methods considered in this paper, the basic idea consist of the following: the domain is discretized and partitioned into several subregions, then, by applying block elimination to the discretized equations, a system is derived for the unknowns on the interfaces between subregions. This system is sometimes called the capacitance system. Forming the right hand side for the interface system and solving it requires the solution of independent elliptic problems on the subdomains. For certain constant coefficient problems, fast direct methods can be applied to the solution of the interface system. Such is not the case, however, for more general operators on irregular domains. For efficiency reasons the system must then be solved by iterative methods, such as the preconditioned conjugate gradient method (PCG). Once the solution is known on the interfaces, one more elliptic problem must be solved on each subdomain with the computed values as boundary conditions. The method is particularly suited to problems for which the subproblems can be solved efficiently, for example, when the operator has separable coefficients and the domain is a union of rectangles. On the other hand, when the subdomain problems cannot be solved efficiently but they can be approximated by simpler operators, it is possible to derive block preconditioners for the original system based on preconditioners for the capacitance matrix.

In section 2, we illustrate the method for the case of a domain that is the union of two rectangles. In section 3, we consider the Poisson and Helmholtz equations on a rectangular domain divided into parallel strips and derive the capacitance system for the interface variables. For these simple and regular cases, the capacitance system can be solved by fast direct methods. Such is not the case for irregular domains. In section 4, we summarize the various preconditioners that were proposed in the literature for use with the CG method. For the case of variable coefficient problems, when fast direct methods are not applicable to the solution of the problems on the subdomains, the system for the whole domain must be solved by iterative methods. Using the results of the previous sections, preconditioners for the large system can be derived from preconditioners for the capacitance matrix. We discuss this case in section 5. Finally, in section 6, we propose a new family of row-sum preserving banded preconditioners for the capacitance matrix. These preconditioners have the advantage that they can be applied to a more general class of problems, since as opposed to most of the other preconditioners, they do not depend on special properties of the differential operator.

2. Domain Decomposition

We will first consider the problem:

$$Lu = f \qquad \text{on} \quad \Omega \tag{2.1}$$

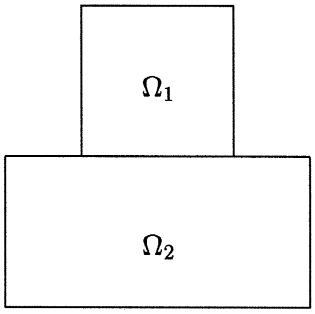

Figure 1: The domain Ω and its partition.

with boundary conditions

$$u = u_b \qquad \text{on} \quad \partial\Omega$$

where L is a linear elliptic operator and the domain Ω is as illustrated in Fig. 1 . We will call the interface between Ω_1 and Ω_2, Γ.

If we order the unknowns for the internal points of the subdomains first and then those in the interface Γ, then the discrete solution vector $u = (u_1, u_2, u_3)$ satisfies the linear system

$$Au = b \quad , \tag{2.2}$$

that can be expressed in block form as:

$$\begin{pmatrix} A_{11} & & A_{13} \\ & A_{22} & A_{23} \\ A_{13}^T & A_{23}^T & A_{33} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} \tag{2.3}$$

The system (2.3) can be solved by Block-Gaussian Elimination as follows:

Step 1: Compute

$$C = A_{33} - A_{13}^T A_{11}^{-1} A_{13} - A_{23}^T A_{22}^{-1} A_{23}, \tag{2.4}$$

$$w_1 = A_{11}^{-1} b_1 \tag{2.5}$$

$$w_2 = A_{22}^{-1} b_2 \tag{2.6}$$

and solve

$$C u_3 = b_3 - A_{13}^T w_1 - A_{23}^T w_2 \tag{2.7}$$

Step 2: Compute

$$u_1 = w_1 - A_{11}^{-1} A_{13} u_3 \tag{2.8}$$

and

$$u_2 = w_2 - A_{22}^{-1} A_{23} u_3 \tag{2.9}$$

Note that, except for (2.7), the algorithm only requires the solution of problems with A_{11} and A_{22}, which corresponds to solving independent problems on the subdomains. This technique of reducing the problem on Ω to the solution of decoupled problems on the subdomains and a smaller system for the interface is usually called *domain decomposition* or *substructuring.* The matrix C (2.4) is the Schur complement of A_{33} in A and it is sometimes called the *capacitance matrix* in this context.

3. Poisson and Helmholtz Equations on a rectangle

We now consider the case where L is the Laplacian operator and Ω is a rectangle divided into two or more strips like is shown in Fig. 2. For this case, the exact eigenvectors and eigenvalues of C are known [2, 6, 7].

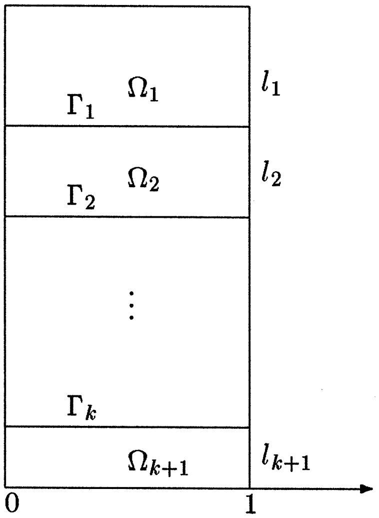

Figure 2: Rectangular domain divided into strips.

For the case of two strips, C has the following eigenvalue decomposition:

$$W \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{pmatrix} W^T \quad , \tag{3.1}$$

where W is the matrix whose columns are

$$w_j = \sqrt{\frac{2}{n+1}}(\sin j\pi h, \sin 2j\pi h, \cdots, \sin nj\pi h)^T \tag{3.2}$$

and

$$\lambda_j = -\left(\frac{1+\gamma_j^{m_1+1}}{1-\gamma_j^{m_1+1}} + \frac{1+\gamma_j^{m_2+1}}{1-\gamma_j^{m_2+1}}\right)\sqrt{\sigma_j + \frac{\sigma_j^2}{4}}. \tag{3.3}$$

for $j = 1, \ldots, n$, where

$$\sigma_j = 4 \sin^2 \left(\frac{j \pi h}{2} \right) \quad , \tag{3.4}$$

$$\gamma_j = \left(1 + \frac{\sigma_j}{2} + \sqrt{\sigma_j + \frac{\sigma_j^2}{4}} \right)^2 \tag{3.5}$$

h is the grid size, and m_1 and m_2 are the number of rows of grid points in the y-direction in Ω_1 and Ω_2 respectively. By using the decomposition (3.1), the capacitance system (2.7) can be solved by fast Fourier transforms. Once the solution u_3 on the interface is computed, we can compute u_1 and u_2 by (2.8) and (2.9), which correspond to solving two independent problems on the subdomains with boundary condition u_3 on Γ.

In the multistrip case, the matrix C has the block-tridiagonal structure:

$$C = \begin{pmatrix} C_1 & B_2 & & \\ B_2 & C_2 & \ddots & \\ & \ddots & \ddots & B_k \\ & & B_k & C_k \end{pmatrix} \quad ; \tag{3.6}$$

all blocks C_i and B_i have the same matrix of eigenvectors W, i.e. for $i = 1, \ldots, k$, we have

$$W^T C_i W = \Lambda_i = \ \mathrm{diag}(\lambda_{i1}, \ldots, \lambda_{in}) \tag{3.7}$$

and for $i = 2, \ldots, k$, we have

$$W^T B_i W = D_i = \ \mathrm{diag}(\delta_{i1}, \ldots, \delta_{in}) \tag{3.8}$$

where

$$\lambda_{ij} = - \left(\frac{1 + \gamma_j^{m_i + 1}}{1 - \gamma_j^{m_i + 1}} + \frac{1 + \gamma_j^{m_{i+1} + 1}}{1 - \gamma_j^{m_{i+1} + 1}} \right) \sqrt{\sigma_j + \frac{\sigma_j^2}{4}}, \tag{3.9}$$

and

$$\delta_{ij} = \sqrt{\gamma_j^{m_i}} \left(\frac{1 - \gamma_j}{1 - \gamma_j^{m_i + 1}} \right) . \tag{3.10}$$

The capacitance system can then be solved by fast Fourier transforms and the solution of n decoupled tridiagonal systems of dimension k, where $k + 1$ is the number of subdomains [7].

Although it first appears that the algorithm requires the solution of two problems on each subdomain, one for computing the right hand side and one for computing the solution on each subdomain, some computations can be saved. We refer the interested reader to [8], where a detailed operation count is derived for the sequential and parallel implementations.

Formulas (3.6) to (3.10) can be generalized to two particular operators other than the Laplacian: the linear elliptic operator

$$u_{xx} + \beta u_{yy} \quad , \tag{3.11}$$

where the coefficient β takes constant values β_i on each subdomain Ω_i and the Helmholtz operator

$$\Delta u + \alpha u \quad . \tag{3.12}$$

The capacitance matrix for the operator (3.11) has the same form as (3.6). The eigenvalues of C_i and B_i are given by

$$\lambda_{ij} = -\left(\frac{1+\gamma_{ij}^{m_i+1}}{1-\gamma_{ij}^{m_i+1}}\right)\sqrt{\frac{\sigma_j^2}{4}+\beta_i\sigma_j} \quad - \quad \left(\frac{1+\gamma_{i+1,j}^{m_{i+1}+1}}{1-\gamma_{i+1,j}^{m_{i+1}+1}}\right)\sqrt{\frac{\sigma_j^2}{4}+\beta_{i+1}\sigma_j} \tag{3.13}$$

and

$$\delta_{ij} = \beta_i\sqrt{\gamma_{ij}^{m_i}}\left(\frac{1-\gamma_{ij}}{1-\gamma_{ij}^{m_i+1}}\right) \quad ,$$

where

$$\gamma_{ij} = \frac{1}{\beta_i^2}\left(\frac{\sigma_j}{2}+\beta_i-\sqrt{\frac{\sigma_j^2}{4}+\beta_i\sigma_j}\right)^2 \quad . \tag{3.14}$$

The operator (3.11) can be used as a preconditioner for more general variable coefficient problems.

The Helmholtz operator (3.12) has important applications in the solution of time dependent problems. The capacitance matrix for this operator also has the form (3.6), where the eigenvalues of C_i and B_i are

$$\lambda_{ij} = -\left(\frac{1+\gamma_j^{m_i+1}}{1-\gamma_j^{m_i+1}}+\frac{1+\gamma_j^{m_{i+1}+1}}{1-\gamma_j^{m_{i+1}+1}}\right)\sqrt{\frac{\mu_j^2}{4}-1} \tag{3.15}$$

and

$$\delta_{ij} = \sqrt{\gamma_j^{m_i}}\left(\frac{1-\gamma_j}{1-\gamma_j^{m_i+1}}\right) \quad , \tag{3.16}$$

where

$$\mu = -\sigma_j - 2 + \alpha h^2 \tag{3.17}$$

and

$$\gamma_j = \left(-\frac{\mu_j}{2}-\sqrt{\frac{\mu_j^2}{4}-1}\right)^2 \quad . \tag{3.18}$$

4. Poisson Equation on Irregular Domains

In general, when Ω has irregular shape like in Fig. 1, the eigenvalues and eigenvectors of the capacitance matrix are not known. The computation of the capacitance matrix is expensive, since it requires the solution of $m+1$ systems with A_{11} and A_{22}, and it is also expensive to invert for m large, because it is dense.

Instead of solving the system (2.7) directly, preconditioned conjugate gradient methods (PCG) can be applied, where only matrix vector products Cy for arbitrary $y \in R^m$ are required. This product can be computed by solving one Poisson problem on each subdomain with boundary condition on Γ given by y.

Since each iteration involves the solution of problems on the subdomains, keeping the number of iterations small is very important for the efficiency of the method. This can be achieved by choosing a good preconditioner for C, several of which are given in the literature [3, 10, 13, 5]. We summarize these here:

1. In [10], Dryja proposes

$$M_D = \sqrt{K} \quad ,$$

as a preconditioner for (2.4), where K is the one-dimensional Laplacian. He proved that the condition number of the preconditioned system, $\mathcal{K}(M_D^{-1}C)$ is bounded independently of the mesh size h. Since M_D has the following factorization

$$M_D = W diag(\lambda_1^D, \lambda_2^D, \cdots, \lambda_n^D) W^T \quad , \tag{4.1}$$

where the columns of W are given by (3.2) and

$$\lambda_j^D = -2\sqrt{\sigma_j} \tag{4.2}$$

with σ_j given by (3.4), M_D can be inverted by FFT's.

2. Golub and Mayers [13] propose a preconditioner given by

$$M_G = \sqrt{K^2 + 4K} \quad ,$$

which has the following decomposition:

$$M_G = W diag(\lambda_1^G, \lambda_2^G, \cdots, \lambda_n^G) W^T \quad , \tag{4.3}$$

where

$$\lambda_j^G \equiv -2\sqrt{\sigma_j + \frac{\sigma_j^2}{4}} \quad . \tag{4.4}$$

Empirical results in [13] show that M_G performs better than M_D.

3. Another interesting preconditioner was given by Björstad and Widlund [3] and has the following form:

$$M_B = A_{33} - 2A_{13}^T A_{11}^{-1} A_{13} \quad .$$

It is easy to show that the eigenvalue decomposition of M_B is

$$M_B = W diag(\lambda_1^B, \lambda_2^B, \cdots, \lambda_n^B) W^T \quad , \tag{4.5}$$

where

$$\lambda_j^B = -2\left(\frac{1+\gamma_j^{m_1+1}}{1-\gamma_j^{m_1+1}}\right)\sqrt{\sigma_j + \frac{\sigma_j^2}{4}}$$

for $j = 1, \ldots, n$. When Ω_1 and Ω_2 are identical, M_B is an exact preconditioner. To implement the method, Björstad and Widlund solve a mixed Neumann-Dirichlet problem in one of the subdomains and a Dirichlet problem in the other one. Their method has the advantage that it can be applied to more general operators and domain shapes, but in the particular case of the Laplacian operator on a union of rectangles, it is less efficient than applying a single FFT computation on the interface grid points, as the factorization (4.5) sugests.

4. Although M_D, M_G and M_B were derived independently of the factorization (3.1), they can be viewed as progressively better approximations to the capacitance matrix C [1]. The factorization (3.1) is exact for the case of a rectangular Ω, while M_D and M_G are not. It can be easily observed that (4.2) is a first order approximation to (3.3), while (4.4) is a second order approximation. On the other hand, M_B is exact only for the case of a rectangular domain divided into two

[1] Anderson [1] gives an interpretation of the various discrete preconditioners as approximations to a continuous operator on the interface

identical subregions. All this sugests that (3.1) might be a better preconditoner for the case of an irregular domain [6].. We will call this preconditioner M_C.

In Fig. 3 we compare the preconditioners M_D, M_G and M_C for the Poisson equation on a T-shaped region Ω as given in Fig. 1, where we vary the aspect ratio of the subdomain Ω_1. We consider a uniform grid on Ω with 15 grid points on the interface Γ. By varying m_1, the number of interior grid points in the y direction on the subdomain Ω_1, we computed the condition number of the preconditioned capacitance system for different aspect ratios $\frac{m_1+1}{n+1}$. As we can see, M_C performs very well, even when Ω_1 becomes very narrow, while the others deteriorate. All M_C, M_G and M_W are indistinguishable for aspect ratios larger than one and they are all better than M_D. In [6], Chan analyzes and compares these preconditioners on rectangular regions. By his analysis, we can see that M_G is always better than M_D on a rectangle and both preconditoners perform poorly when the aspect ratio for the dimension of the rectangles is small. See [14] for a careful numerical comparison of these and other preconditioners for constant and variable coefficients operators.

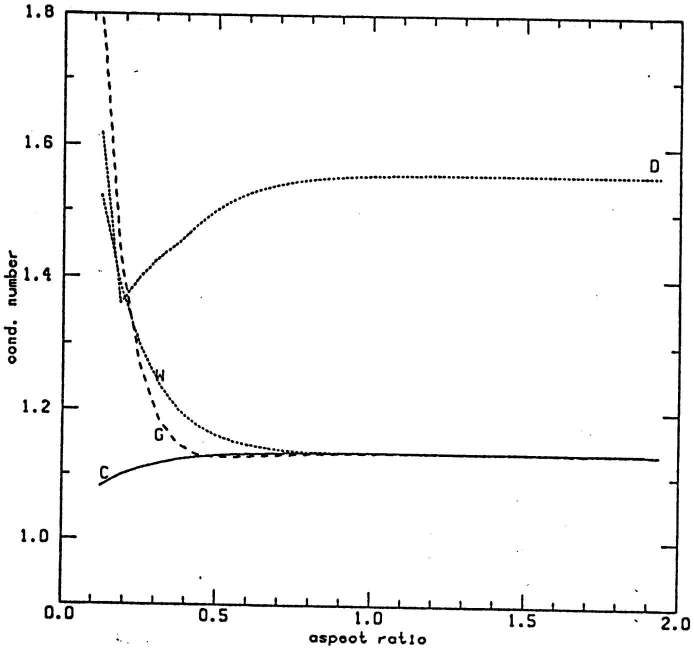

Figure 3: T-shaped region. Condition number of the preconditioned capacitance matrix with Chan's (C), Dryja's(D), Björstad and Widlund's (W) and Golub and Mayers' (G) preconditoners.

5. Variable Coefficient Problems

In the case of non-constant coefficient problems, there usually are no fast solvers available for A_{11} and A_{22} and therefore the solutions to systems with these matrices are to be avoided. In that case, a Krylov subspace method can be applied to solve the system (2.3) on the whole domain instead of just the capacitance system on the interface. Therefore, we must now be concerned with the problem of finding preconditioners for (2.3) that make use of the regularity of the subdomains. We will show that preconditioners for (2.3) can be derived from preconditioners for the capacitance matrix. Assume that the variable coefficient operator can be approximated by operators with constant coefficients on each subdomain. In particular, let B_{11} and B_{22} be approximations to A_{11} and A_{22}, corresponding to the discretization of linear elliptic operators with constant coefficients on Ω_1 and Ω_2 respectively. Based on the following decomposition of the matrix A in (2.3):

$$A = \begin{pmatrix} A_{11} & & \\ & A_{22} & \\ A_{31} & A_{32} & C \end{pmatrix} \begin{pmatrix} I & & A_{11}^{-1}A_{13} \\ & I & A_{22}^{-1}A_{23} \\ & & I \end{pmatrix} , \qquad (5.1)$$

where C is the Schur complement (2.4), we can derive a preconditoner for A given by:

$$\tilde{M} = \begin{pmatrix} B_{11} & & \\ & B_{22} & \\ A_{31} & A_{32} & M \end{pmatrix} \begin{pmatrix} I & & B_{11}^{-1}A_{13} \\ & I & B_{22}^{-1}A_{23} \\ & & I \end{pmatrix} , \qquad (5.2)$$

where M is a good preconditioner for the matrix C. We can see that $\tilde{M}$ is easily invertible by block-elimination, since fast solvers can be applied to solve systems with B_{11} and B_{22}.

Preconditioners of the form (5.2) were first used by Bramble, Pasciak and Schatz's [4, 5]. In [4], Dryja's preconditioner is used as the matrix M in (5.2). The second preconditioner in [5] corresponds to chosing the matrix M given by Björstad and Widlund [3]. As a generalization of their idea, any of the preconditioners given for the constant coefficients case can be applied here as M. In fact, a theorem by Eisenstat in [14] shows that, when $B_{ii} = A_{ii}$, the PCG algorithm applied to (2.7) with preconditioner M and initial guess u_3^0 is equivalent to the PCG algorithm applied to (2.3) with preconditioner given by (5.2) and initial guess $(A_{11}^{-1}(b_1 - A_{13}u_3^0), A_{22}^{-1}(b_2 - A_{23}u_3^0), u_3^0)$. In [14], numerical experiments were performed with these and other preconditioners.

6. A new class of banded, row-sum preserving preconditioners

We now present a new family of preconditioners for the capacitance matrix C. These preconditioners are motivated by the empirical observation that the elements of the matrix C decay away from the main diagonal. It is, therefore, reasonable to consider k-diagonal approximations to C. It would not, however, be efficient to compute the elements of C in order to do this. We now present a method for computing a k-diagonal approximation to C without requiring the computation of C explicitly. The idea is motivated by sparse Jacobian evaluation techniques [9]. For example, for the case $k = 3$, the approximant M to C can be computed in compact form by evaluating the three products $Cu_i, i = 1, 2, 3$, where $u_3 = (1, 0, 0, 1, 0, \ldots)^T$, $u_3 = (0, 1, 0, 0, 1, \ldots)^T$ and $u_3 = (0, 0, 1, 0, 0, \ldots)^T$. The motivation is clear, for if C were indeed tridiagonal, $(k = 3)$, then all of its nonzero elements can be found in the three vectors $Cu_i, i = 1, 2, 3$. Note that the computation of each product Cu_i involves solving one problem on each subdomain with u_i as boundary condition on the interface.

The generalization to other values of k is obvious. Moreover, it can be easily verified that the matrix M computed this way preserves the row-sums of C. The case $k = 1$, however, deserves special mention. The method described above would compute a diagonal approximation to C, with diagonal entries given by Ce, where $e = (1, 1, \ldots, 1)^T$. However, since the first term A_{33} in the definition of C in (2.4) is already known explicitly (and it is tridiagonal), it is only necessary to apply the above approximation procedure to the last two terms in (2.4). The resulting matrix M is thus tridiagonal, namely, A_{33} with the diagonal entries modified in such a way that the row sums of C are preserved. Viewed this way, the case $k = 1$ is similar in spirit to the Dupont-Kendall-Rachford procedure [11] for obtaining an easily invertible banded approximant for C. This special procedure for the case $k = 1$ was sugested independently by Eisenstat [12]. See [14] for numerical experiments with this class of preconditioners.

In general, for a k-diagonal approximation to C, k problems on each subdomain must be solved, which may seem prohibitively expensive except for small values of k. However, the main advantage of this family of preconditioners is that they are less dependent on special properties (e.g. eigenstructures) of the differential operator underlying A. Moreover, for nonlinear problems where a Newton type outer iteration may be involved, one preconditioner can be reused several times and the cost of computing it can be amortized over the overall iteration. Further details will be reported elsewhere.

References

[1] C. R. Anderson, *On Domain Decomposition.,* Technical Report Manuscript CLaSSiC-85-09, Center for Large Scale Scientific Computation , Stanford University, October 1985.

[2] R. Bank and D. Rose, *Marching Algorithms for Elliptic Boundary Value Problems. I: The Constant Coefficient Case,* SIAM J. Numer. Anal., 14/5 (1977), pp. 792–828.

[3] P. E. Bjorstad and O. B. Widlund, *Iterative Methods for the Solution of Elliptic Problems on Regions Partitioned into Substructures,* Technical Report 136, Courant Institute of Mathematical Sciences, NYU, September 1984.

[4] J. H. Bramble, *The Construction of Preconditioners for Elliptic Problems by Substructuring,* manuscript, 1984.

[5] J.H. Bramble, J.E. Pasciak and A.H. Schatz, *Preconditioners for Interface Problems on Mesh Domains,* manuscript.

[6] T.F. Chan, *Analysis of Preconditioners for Domain Decomposition,* Technical Report YALEU/ DCS/RR-408, Yale Computer Science Department, 1985.

[7] T.F. Chan and D.C. Resasco, *A Domain-Decomposed Fast Poisson Solver on a Rectangle.,* Technical Report YALEU/DCS/RR-409, Yale Computer Science Department, 1985.

[8] T.F. Chan, D.C. Resasco and F. Saied, *Implementation of Domain Decomposed Fast Poisson Solvers on Multiprocessors,* Technical Report YALE/DCS/RR-456, Yale Computer Science Department, 1985.

[9] A.R. Curtis, M.J.D. Powell and J.K. Reid, *On the Estimation of Sparse Jacobian Matrices,* J. Inst. Maths. Applics., 13 (1974), pp. 117–119.

[10] M. Dryja, *A Capacitance Matrix Method for Dirichlet Problem on Polygonal Region,* Numer. Math., 39 (1982), pp. 51–64.

[11] T.Dupont, R.P. Kendall and H.H. Rachford, *An Approximate Factorization Procedure for Solving Self-Adjoint Elliptic Difference Equations,* SIAM Journal on Numerical Analysis, 6 (1968), pp. 753–782.

[12] S. Eisenstat, *personal communication,* 1985.

[13] G. H. Golub and D. Mayers, *The Use of Pre-Conditioning over Irregular Regions,* 1983. Lecture at Sixth Int. Conf. on Computing Methods in Applied Sciences and Engineering, Versailles, Dec. 1983.

[14] D. Keyes and W. Gropp, *A Comparison of Domain Decomposition Techniques for Elliptic Partial Differential Equations,* Technical Report YALEU/DCS/RR-448, Yale Computer Science Department, 1985.