Numerical Analysis Project
Manuscript NA-83-27

The Use of Pre-Conditioning over Irregular Regions

by

Gene H. Golub and David Mayers

Numerical Analysis Project
Computer Science Department
Stanford University
Stanford, California 94305

The Use of Pre-Conditioning over Irregular Regions

Gene H. Golub*
Computer Science Department
Stanford University
Stanford, California 94305
USA

David Mayers
Oxford University Computing Laboratory
19 Parks Road
Oxford OX1 3PL
England

Abstract

Some ideas and techniques for solving elliptic pde.'s over irregular regions are discussed. The basic idea is to break up the domain into subdomains and then to use the pre-conditioned conjugate gradient method for obtaining the solution over the entire domain. The solution of Poisson's equaton over a T-shaped region is described in some detail and a numerical example is given.

1. Introduction.

In the last several years, there has been great interest in solving a variety of problems using ***domain decomposition*** or ***substructuring*** (cf. [1], [2], [3], [4], [9]). The basic idea is to break up a domain into subdomains and then to solve for an approximate solution on each subdomain. There is a need then to "paste" the approximate solutions together to get the solution to the original problem.

It turns out that a very effective numerical procedure for obtaining the solution is the pre-conditioned conjugate gradient method (cf. [3]). It provides **a** particularly good method because advantage is taken of the eigenvalue structure of certain matrices.

It is obvious that the ideas of domain decomposition are quite applicable to parallel processing. We will not delve into that subject at this time.

We will concentrate on solving Poisson's equation on a T-shaped region. The general ideas expressed here will be applicable to a much wider class of problems.

2. The Pre-Conditioned Conjugate Gradient Method.

Our discussion in this section closely follows that given in [3]. Consider the system of equations

$$\mathbf{A\,x = b} \tag{2.1}$$

where A is a symmetric, positive definite matrix. We re-write (2.1) as

$$\mathbf{M\,x = N\,x + b}$$

where M is symmetric and positive definite. We assume that given a vector $\mathbf{d}$ it is "easy" to solve the system

$$M\mathbf{z} = \mathbf{d}. \tag{2.2}$$

* The work of this author was in part supported by NSF and the DOE. Invited Lecture at Sixth International Conference on Computing Methods in Applied Sciences and Engineering, Versailles, sponsored by INRIA, December 12-16, 1983.

The **pre-conditioned** conjugate gradient (CG) method proceeds as follows.

Algorithm.

Let $\mathbf{x}^{(0)}$ be a given vector and arbitrarily define $\mathbf{p}^{(-1)}$. For $k = 0$, 1, . . .

(1) Solve $M\mathbf{z}^{(k)} = \mathbf{b} - A\mathbf{x}^{(k)} \equiv \mathbf{r}^{(k)}$.

(2) Compute

$$b_0 = o\,,$$

$$b_k = \frac{\mathbf{z}^{(k)^T} M\, \mathbf{z}^{(k)}}{\mathbf{z}^{(k-1)^T} M\, \mathbf{z}^{(k-1)}} \qquad k \geq 1$$

$$\mathbf{p}^{(k)} = \mathbf{z}^{(k)} + b_k\, \mathbf{p}^{(k-1)}\,.$$

(3) Compute

$$a_k = \frac{\mathbf{z}^{(k)^T} M\, \mathbf{z}^{(k)}}{\mathbf{p}^{(k)^T} A\, \mathbf{p}^{(k)}}$$

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + a_k\, \mathbf{p}^{(k)}\,.$$

Note that in the computation of the numerator of a_k and b_k one need not compute $M\,\mathbf{z}^{(k)}$ since $\mathbf{M}\ \mathbf{z}^{(k)} = \mathbf{r}^{(k)}$. Furthermore, the matrix A need not be known explicitly since we need compute the vector $\mathbf{q}^{(k)} = A\ \mathbf{p}^{(k)}$. Finally, we note that the residual $\mathbf{r}^{(k+1)}$ can be computed by the relationship

$$\mathbf{r}^{(k+1)} = \mathbf{r}^{(k)} - a_k\, A\,\mathbf{p}^{(k)}\,.$$

The convergence properties of the conjugate gradient method are fairly well understood. Let

$$K = M - ‘A$$

and

$$\kappa = \frac{\lambda_{\max}(M^{-1}A)}{\lambda_{\min}(M^{-1}A)}$$

where $\lambda_{\max}(M^{-1}A)$ is the largest eigenvalue of the argument and $\lambda_{\min}(M^{-1}A)$ is the smallest.

The following is known about the CG method.

$$(A) \qquad \frac{(\mathbf{x}^{(\ell)} - \mathbf{x})^T A\ (\mathbf{x}^{(\ell)} - \mathbf{x})}{(\mathbf{x}^{((0)} - \mathbf{x})^T A\ (\mathbf{x}^{(0)} - x)} \leq 4\left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^{2\ell}.$$

(B) If $K = M^{-1}A$ has p distinct eigenvalues then the CG method converges in at most p iterations. ·

Thus if we wish to solve the system

$$(A + \mathbf{u}\mathbf{u}^T)\,\mathbf{x} = \mathbf{b} \qquad (\mathbf{u} \neq \mathbf{0})$$

with $M = A$ and $N = -\mathbf{u}\,\mathbf{u}^T$, the CG method will converge in at most 2 iterations since $M^{-1}N$ is a matrix of rank one and hence has at most two distinct eigenvalues.

Our aim will be to devise matrices M so that κ is as small as possible and that the matrix N be of low rank.

In many cases the matrix A can be written as

$$A = \begin{pmatrix} M_1 & F \\ F^T & M_2 \end{pmatrix}$$

where the systems

$$M_1\, \mathbf{z}_1 = \mathbf{d}_1 \qquad \text{and} \qquad M_1\, \mathbf{z}_2 = \mathbf{d}_2$$

are easy to solve and for such matrices, it is convenient to choose

$$M = \begin{pmatrix} M_1 & 0 \\ 0 & M_2 \end{pmatrix} .$$

Suppose we are given $\mathbf{x}_1^{(0)}$ and we compute $\mathbf{x}_2^{(0)}$ so that

$$M_2\, \mathbf{x}_2^{(0)} = \mathbf{r}_2^{(0)} .$$

A short calculation shows that $\mathbf{z}_1^{(1)} = 0$ and hence $\mathbf{a}_1 = 1$. Generalizing a result of Reid [7], it is shown in [3], then for j = 0, 1, . . .

$$a_j = 1 , \quad \mathbf{z}_1^{(2j+1)} = \mathbf{0} , \quad \mathbf{z}_2^{(2j)} = \mathbf{0} . \tag{2.3}$$

Using (2.3), we are able to eliminate roughly half the-number of operations (cf. [7]).

3. Some Preliminary Results.

In many applications (cf. [5], [6]), matrices of the following structure arise quite naturally:

$$A = \begin{pmatrix} A_1 & & \bigcirc & & B_1 \\ & \cdot & & & B_2 \\ & & \cdot & & \vdots \\ \bigcirc & & & \cdot & \\ & & & A_r & B_r \\ B_1 & \cdot & \cdot & B_r & Q \end{pmatrix} .$$

The matrix A is symmetric and positive definite so that each $A;$ $(n_i \text{ x } n_i)$ is also positive definite and so is the $p \times p$ matrix Q.

Suppose we wish to solve the system

$$\mathbf{A}\, \mathbf{x} = \mathbf{b} \tag{3.1}$$

where

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_r \\ \xi \end{pmatrix} \qquad \text{and} \qquad \mathbf{b} = \begin{pmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_r \\ c \end{pmatrix} .$$

One possibility is to apply the CG method to (3.1) with

$$M = \begin{pmatrix} A_1 & & \bigcirc & \\ & \cdot & & \\ \bigcirc & & A_r & \\ & & & Q \end{pmatrix} .$$

Note the matrix $M^{-1}N$ is 2-cyclic and as indicated in Section 2 various simplifications can be made in the procedure and about half the computations can be eliminated.

Let us examine the eigenvalues of $M^{-1}N$. Consider the matrix equation

$$M^{-1}N\mathbf{u} = \lambda\mathbf{u}.$$

Then if u = $\begin{pmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_r \\ \mathbf{v} \end{pmatrix}$, a short manipulation shows

$$\sum_{i=1}^{r} B_i^T A_i^{-1} B_i \mathbf{v} = \lambda^2 Q \mathbf{v}.$$

Instead of solving (3.1) directly, we could eliminate $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_r$ and solve for ξ. This leads to the equation

$$\left(Q - \sum_{i=1}^{r} B_i^T A_i^{-1} B_i\right)\xi = \mathbf{c} - \sum_{i=1}^{r} B_i^T A_i^{-1} \mathbf{b}_i. \tag{3.3}$$

We can apply the CG method to (3.3) and if Q is a p x p matrix we will obtain the solution in at most p iterations. The matrix

$$G \equiv Q - \sum_{i=1}^{r} B_i^T A_i^{-1} B_i$$

is the Schur complement of Q in A and hence G is symmetric and positive definite.

Associated with (3.3), we can choose a pre-conditioner $\tilde{M}$. There are various choices of $\tilde{M}$. One possibility is to choose $\tilde{M} = I$. Perhaps a more natural pre-conditioner is $\tilde{M} = Q$. Note if

$$\tilde{M}\xi = \tilde{N}\xi + \tilde{\mathbf{b}},$$

then the convergence properties of the algorithm are determined by the eigenvalues of $\tilde{M}^{-1}\tilde{N}$. Let

$$\tilde{M}^{-1}\tilde{N}\mathbf{w} = \gamma\mathbf{w}.$$

Then if $\tilde{M} = Q$, we have

$$\sum_{i=1}^{r} B_i^T A_i^{-1} B_i \mathbf{w} = \gamma Q \mathbf{w}. \tag{3.4}$$

Hence the cigenvalues of (3.2) are the squares of the eigcnvalues of (3.4). The rate of convergence of the two procedures are essentially the same bccause we arc able to eliminate half of the numerical operations. In Section 4, wc shall indicate how we can further improve the convcrgence properties by choosing various pre-conditioners related to the differential equations.

In ordcr tc solve (3.3) with $\tilde{M}$ = Q, we need to solve the system

$$Q\,\mathbf{z}^{(k)} = \mathbf{c} + \sum_{i=1}^{r} B_i^T A_i^{-1}(B_i\eta - \mathbf{b}_i)$$

where η is an approximation to the solution. Note that one need not compute the matrix $B_i^T A_i^{-1} B_i$ but instead one solves the **system**

$$A_i \varphi_i = B_i \eta - \mathbf{b}_i .$$

If *A;* has a simple structure this can be solved without much difficulty.

4. The Model Problem.

We shall explain the method by applying it to the problem **studicd** in [3]. We wish to solve

$$\text{(4.1)} \qquad \begin{cases} -\mathrm{A}\, u = f & (x,y) \in T \\ u = g & (x,y) \in \partial T \end{cases}$$

where T is the T-shaped domain shown in Figure 1. We use a uniform square grid of size h, where $1/2h$ is an integer N, and $0 < \ell < N$. Using the standard 5-point finite difference approximation we obtain the system of equations in matrix form

$$\text{(4.2)} \qquad \begin{pmatrix} P & J & O \\ J^T & R & K^T \\ 0 & K & Q \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{w} \\ \mathbf{v} \end{pmatrix} = \begin{pmatrix} \mathbf{a} \\ \mathbf{b} \\ \mathbf{c} \end{pmatrix} .$$

The vector u comprises the unknown values of the solution at the interior points of the lower square, v comprises the unknowns in the interior of the upper square, and w comprises the unknown values at grid points on the dividing line. Thus *P, Q*, and *R* are square matrices of orders $(2N-1)^2$, $(2\ell-1)^2$, and $(2\ell-1)$ respectively.

There are several possible pre-conditioners one can use for solving (4.2). The obvious choices **-are**

$$M^{(1)} = \begin{pmatrix} P & J & O \\ J^T & R & O \\ O & O & Q \end{pmatrix}$$

and

$$M^{(2)} = \begin{pmatrix} P & O & O \\ O & R & O \\ O & O & Q \end{pmatrix} . \qquad 0$$

The pre-conditioner $M^{(1)}$ requires that for each iteration two fast Poisson solvers are used whereas for pre-conditioner $M^{(2)}$ two fast Poisson solvers are used and a system of equations involving the matrix Q must be solved. Which pre-conditioner is to bc preferred? Using a **thcorcm** of Varga [8, pg. 90], we see that. the spectral radius is smaller when $M^{(1)}$ is used as the pre-conditioner and hence $M^{(1)}$ is to be preferred in many situations.

Theorem (Varga):

Let $A = M^{(1)} - N^{(1)} = M^{(2)} - N^{(2)}$ be two regular splittings of A where $A^{-1} > 0$. If $N^{(2)} \geq N^{(1)} \geq 0$, equality excluded, then

$$1 > \rho({M^{(2)}}^{-1} N^{(2)}) \geq \rho({M^{(1)}}^{-1} N^{(1)}) > 0 .$$

Here $\rho({M^{(i)}}^{-1} N^{(i)})$ denotes the spectral radius of tlic argument. Even though it appears that $M^{(1)}$ is to be preferred over $M^{(2)}$, we will considcr bclow a pre-conditioner rclated to $M^{(2)}$!

WC obtain thc **capacitance matrix** by eliminating u and v from these equations, giving

$$\text{(4.3)} \qquad (R - J^T P^{-1} J - K^T Q^{-1} K) w = \mathbf{b} - J^T P^{-1} \mathbf{a} - K^T Q^{-1} \mathbf{c}$$

which we shall write

$$\text{(4.4)} \qquad \mathbf{C\,w} = \mathbf{d}\,.$$

As indicated in Section 3, it follows at once that C is symmetric and positive definite.

We now wish to solve (4.4) by a pre-conditioned conjugate gradient method. If we can find a suitable pre-conditioning matrix this iterative method converges very rapidly, and it will certainly terminate in at most $(2\ell - 1)$ iterations, this being thc order of the matrix C.

We now consider the computation of $C\mathbf{p}^{(k)}$. Returning to the original equation (4.2) it will be seen that the matrices P and Q correspond to the solution of the discrete form of Poisson's equation on the lower and upper squares respectively, given values on the boundaries of these squares. The matrices J and K consist of unit matrices, augmcnted by blocks of zeros. Hence to compute $C\mathbf{p}^{(k)}$ we need to solve the discrete Laplace equation in the two squares, given zero boundary conditions at all boundary points, except at the points on the dividing line between the two squares, where the given boundary values are the elements of the vector $\mathbf{p}^{(k)}$. The required vector $C\mathbf{p}^{(k)}$ then consists of the residuals of the 5-point formula evaluated at each of these dividing points. The two discrete Laplace problems are both Dirichlet problems on a rectangle, and can be solved very efficiently by a fast Poisson Solver, using some form of Fast Fourier Transform.

For the most rapid rate of convergence we wish the matrix M to be close to C, and in particular for the eigenvalues of $M^{-1}C$ to be clustered as closely as possible. If we examine the elements of C in some particular cases we find that it is quite close to a Toeplitz form, with the element C_{ij} being mainly a function of $|i-j|$ only, and with the largest element on the diagonal, the elements decreasing quite rapidly as $|i-j|$ increases.

This suggests that the elements of C do not depend very much on the shape and size of the -two rectangles in Figure 1, and that we could find a useful approximation to C by letting the boundaries of the two squares move away to infinity. We must then find solutions of Laplace's equation in the two half planes. In this approximation the elements C_{ij} are in fact dependent only on $|i-j|$, and the elements in each row are obtained by solving the discrete Laplace equation in a half plane, the solution being required to vanish at infinity, and also at all points on the axis except at the origin, at which it is equal to 1. We thus wish to solve

$$\begin{gathered} u_{r,s+1} + u_{r,s-1} + u_{r+1,s} + u_{r-1,s} - 4u_{rs} = 0, \qquad s > 0 \\ u_{rs} \to 0 \qquad as \qquad r \to \pm\infty, \qquad \text{and as} \qquad s \to \infty \\ u_{r,0} = 0 \qquad (r \neq 0) \\ u_{0,0} = 1. \end{gathered}$$

Defining the generating function

$$\phi_s(t) = \sum_{r=-\infty}^{\infty} t^r u_{rs}$$

we thus obtain

$$\begin{gathered} \phi_{s+1} + \phi_{s-1} + \left(t + \frac{1}{t} - 4\right)\phi_s = 0, \qquad (s > 0) \\ \phi_0 = 1. \end{gathered}$$

The general solution of the recurrence relation for ϕ_s is

$$\phi_s = A_1\lambda_1^s + A_2\lambda_2^s$$

where A_1 and A_2 are arbitrary constants, and λ_1 and λ_2 are the roots of

$$\lambda^2 + \left(t + \frac{1}{t} - 4\right)\lambda + 1 = 0.$$

The condition at infinity eliminates one of these two terms, and the condition $\phi_0 = 1$ determines the remaining arbitrary constant, giving the solution

$$\phi_s(t) = \left[2 - \frac{1}{2}\left(t + \frac{1}{t}\right) - \left(\left\{2 - \frac{1}{2}\left(t + \frac{1}{t}\right)\right\}^2 - 1\right)^{1/2}\right]^s.$$

The solution in the other half plane is of course determined by symmetry, and the residuals at the grid points on the axis are given by

$$\begin{aligned}\rho_r &= u_{r-1,0} + u_{r+1,0} + u_{r,1} + u_{r,-1} - 4u_{r,0}\\ &= u_{r-1,0} + u_{r+1,0} + 2u_{r,1} - 4u_{r,0}\end{aligned}$$

for which the generating function is •

$$\begin{aligned}\psi(t) &= \left[t + \frac{1}{t} - 4\right|\phi_0 + 2\phi_1\\ &= -2\left\{ \qquad \left[2 - \frac{1}{2}\left(t + \frac{1}{t}\right)\right]^2 - 1\right\}^{1/2}.\end{aligned} \tag{4.5}$$

To determine ρ_r we now expand $\psi(t)$ in positive and negative powers of t; ρ_r is then the coefficient of t^r. The simplest way to do this is to write $t = e^{i\theta}$, $t + \frac{1}{t} = 2\cos\theta$, and then expand $\psi(t)$ as a Fourier cosine series in $\cos\theta$. This gives

$$\begin{aligned}\psi &= -2\{[2 - \cos\theta]^2 - 1\}^{1/2}\\ &= \frac{1}{2}\rho_0 + 2\Sigma\,\rho_r \cos r\theta.\end{aligned}$$

It is then convenient to make the substitution $\theta = 2\alpha$ and obtain the final result

$$\begin{aligned}\rho_r &= \frac{1}{2\pi}\int_{-\pi}^{\pi} -2\cos r\theta[(2 - \cos\theta)^2 - 1]^{1/2}\,d\theta\\ &= -\frac{4}{\pi}\int_0^{\pi}\cos 2r\alpha \sin\alpha[1 + \sin^2\alpha]^{1/2}\,da.\end{aligned} \tag{4.6}$$

Thcse integrals are easily evaluated numerically for small values of r. For moderate and large values of r, it is sufficient to integrate twicc by parts and obtain the asymptotic approximation

$$\rho_r = \frac{2}{\pi r^2} + O\left(\frac{1}{r^4}\right).$$

A possiblc preconditioning matrix M is then obtained by writing

$$M_{ij} = \rho_{|i-j|}.$$

This matrix has in fact been found to give good convergence, but a minor modification improves it still further. Dryja [4] used as pre-conditioning matrix

$$M = K^{1/2}$$

where K is the matrix

$$K = \begin{pmatrix} 2 & -1 & & \\ -1 & 2 & -1 & \\ & -1 & 2 & -1 \\ & & & \cdots \end{pmatrix}.$$

In our notation this corresponds to a different generating function

$$\psi_D = \left[-t + 2 - \frac{1}{t}\right]^{1/2}.$$

This suggests replacing $2 - t - \frac{1}{t}$ by K in the generating function. A multiplying scale factor is unimportant for our purpose, so for comparison we use instead of Dryja's form, $M = -(4K)^{1/2}$, and from(4.5) we suggest trying also

$$-2\left\{\left(1 + \frac{1}{2}K\right)^2 - 1\right\}^{1/2} = -\{4K + K^2\}^{1/2}.$$

The eigenvalues and eigenvectors of the matrix K are well known, so it is a simple matter to write down the elements of these matrices giving three possible pre-conditioners:

$$M^{(1)}_{ij} = \rho_{|i-j|}$$

where ρ_r is given by (4.6)

$$M^{(2)}_{ij} = -\frac{8}{n}\sum_{k=1}^{n-1} \sin\left(\frac{ik\pi}{n}\right) \sin\left(\frac{k\pi}{2n}\right) \sin\left(\frac{jk\pi}{n}\right)$$

$$M^{(3)}_{ij} = -\frac{8}{n}\sum_{k=1}^{n-1} \sin\left(\frac{ik\pi}{n}\right) \sin\left(\frac{k\pi}{2n}\right) \left[1 + \sin^2\left(\frac{k\pi}{2n}\right)\right] \mathrm{sin}(\$)$$

where n is the order of the matrix, in our case $(2\ell - 1)$. Here $M^{(2)}$ is Dryja's pre-conditioner, and $M^{(3)}$ is our modification. Owing to the particular form of $M^{(2)}$ and $M^{(3)}$, thc solution of a system of equations $M\mathbf{z} = \mathbf{r}$ is particularly simple in these two cases, and can be done by a simple application of the Fast Fourier Transform.

For the model problem of Figure 1 we can now evaluate the eigenvalues of the matrix $M^{-1}C$ for each of the three prc-conditioning matrices M. We have used $\ell = \frac{1}{2}N$, with $N = 8$ and 16, giving two sets of matrices, of orders 7 and 15 respectively, for which the eigenvalues are given in Table 1. It will be seen that in each case the eigcnvalucs are clustered quite closely about. 1, the clustering being most marked for the matrix $M^{(3)}$.

Tablc 2 gives somc rcsults which illustrate the rate of convergence of thc method. The domain is as in Figure 1, and the function f and the boundary conditions g are chosen to give a smooth solution as in [3]. The entries in Tablc 2 are the values of $(\mathbf{z}^k, M\mathbf{z}^k)$. which give an indication of how fast z^k, and hence r^k, is tending to zero. The initial approximation was constructed by simply taking $\mathbf{w}^{(0)} = 0$, corresponding to making u = 0 at these grid points.

Acknowledgment. Much of thc work of the first author was performed while a guest at the Oxford University Computing Lab. He is pleased to thank Professor Leslie Fox for his generous hospitality.

The authors wish to thank Gérard Mcurant and Pettcr Bjøstad for their helpful comments.

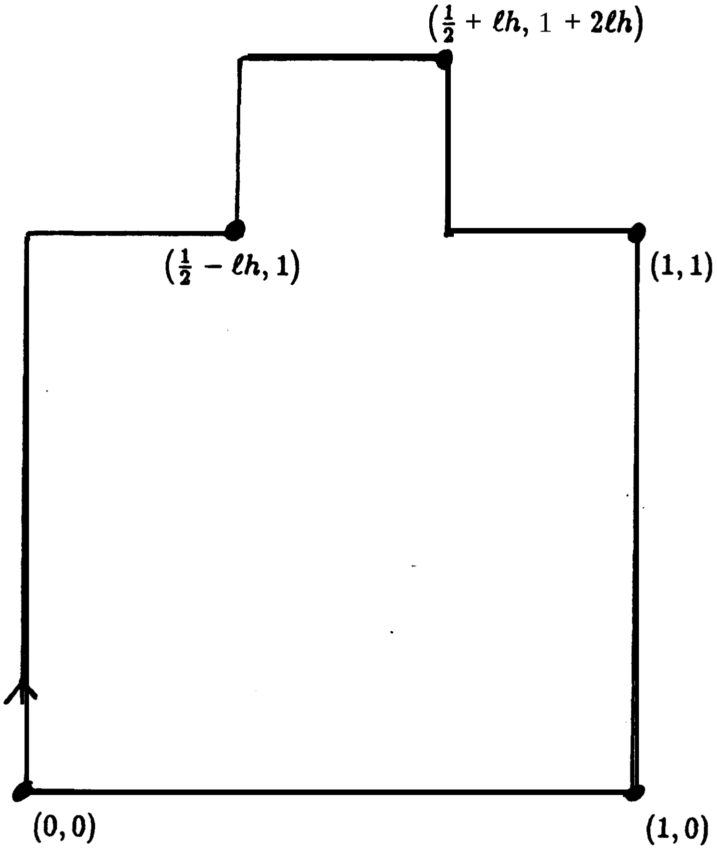
$\left(\frac{1}{2} + \ell h, 1 + 2\ell h\right)$
$\left(\frac{1}{2} - \ell h, 1\right)$
$(1,1)$
$(0,0)$
$(1,0)$

Figure 1

Table 1

Eigenvalues of M^{-1} C, for three matrices M

dimension 7

$M^{(1)}$	$M^{(2)}$	$M^{(3)}$
1.29279	1.40048	1 .00000
1.04310	1.36048	1 .00000
1.00953	1.29815	0.99999
1.00135	1.21928	0.99968
0.99771	1.13432	0.99736
0.98384	1.04073	0.96727
0.98265	**0.93631**	0.91185

dimension 15

$M^{(1)}$	$M^{(2)}$	$M^{(3)}$
1.66158	1.41079	1.00000
1.15918	1.40058	1.00000
1.05216	1.38385	1.00000
1.01339	1.36098	1.00000
1.01081	1.33257	1.00000
1.00885	1.29930	1.00000
1.00617	1.26220	1 .00000
1.09090	1.22217	1.00000
0.99844	1.18079	1.00000
0.99125	1.13894	0.99995
0.99092	1.09911	0.99971
0.98361	1.06133	0.99731
0.98262	1.02975	0.98958
0.97753	0.96949	0.93837
0.97735	***0.89807***	***0.88376***

Table 2

Values of $(a^{(k)}, \boldsymbol{Mz}^{(k)})$, for three pre-conditioning matrices $\boldsymbol{M}$

dimension 7

k	$\boldsymbol{M}^{(1)}$	$\boldsymbol{M}^{(2)}$	$\boldsymbol{M}^{(3)}$
1	0.4, -1	0.9, -1	0.3, -2
2	0.2, -4	0.7, -3	0.5, -6
3	0.4, -8	0.4, -5	0.2, -11
4	0.3, -12	0.3, -7	
5		0.6, -10	
6		0.2, -12	

dimension 15

k	$\boldsymbol{M}^{(1)}$	$\boldsymbol{M}^{(2)}$	$\boldsymbol{M}^{(3)}$
1	0.3	0.1	0.7, -2
2	0.5, -3	0.1, -2	0.3, -5
3	**0.2, -5**	0.1, -4	0.5, -9
4	0.3, -9	0.1, -6	
5		0.1, -8	
6		0.1, -10	

References .

[1] **Bjørstad,** P. and 0. Widlund, "Solving elliptic problems on regions partitioned into substructures,* in ***Elliptic*** Problem Solvers, G. Birkhoff and A. Schoenstadt, eds., Academic Press, New York, 1984.

[2] **Buzbee,** B. L., **Dorr,** F. W., George, J. A. and Golub, G. H., "The direct solution of the discrete Poisson equation on irregular region," SIAM ***J. Numer. Anal. 8*** **(1971),** pp. 722-736.

[3] **Concus,** P., Golub, G. and O Leary, D., "A generalized conjugate gradient method for the numerical solution of elliptic partial differential equations," in Sparse Matrix Computation, **J.** Bunch and D. Rose, eds., Academic Press, New York, 1976, pp. 309-322.

[4] Dryja, M., "A capacitance matrix method for Dirichlet problems on polygonal region," Num. ***Math.,*** **(1982),** pp. 51-64.

[5] Golub, G., H., Luk, F. and Pagano, M., "A large sparse least squares problem in **photogrammetry,"** ***Proceedings of*** the 12th Annual ***Symposium*** on ***the*** Interface, University of Waterloo, May, 1979.

[6] Golub, G. H. and Plemmons, R., "Sparse least squares problems," ***Proceedings of the IRIA*** Fourth ***International*** Symposium on Computing Methods in Applied Science ***and Engineering,*** Versailles, France, Dec. 1979.

[7] Reid, J., "The use of conjugate gradients for systems of linear equations possessing 'Property A';" ***SIAM J. Numer. Anal. 9*** **(1972),** pp. 325-332.

[8] Varga, Richard ***S., Matrix Iterative*** Analysis, Prentice-Hall, Englewood Cliffs, New **Jersy,** 1962.

[9] Dingh, Q. V., Glowinski, R. and Periaux, J., "On the solution of elliptic problems by domain decomposition methods Applications," ***manuscript.***