UCLA
COMPUTATIONAL AND APPLIED MATHEMATICS

Domain Decomposition Algorithms and Computational Fluid Dynamics

Tony F. Chan

August 1988

CAM Report 88-25

Department of Mathematics
University of California, Los Angeles
Los Angeles, CA. 90024-1555

DOMAIN DECOMPOSITION ALGORITHMS AND COMPUTATIONAL FLUID DYNAMICS*

TONY F. CHAN†

Abstract. In the past several years, domain decomposition has been a very popular topic, partly motivated by the potential of parallelization. While a large body of theory and algorithms have been developed for model elliptic problems, they are only recently starting to be tested on realistic applications. In this paper, after a brief introduction and survey of the literature, we shall investigate the application of some of these methods to two model problems in computational fluid dynamics: two dimensional convection-diffusion problems and the incompressible driven cavity flow problem. Our approach is the construction and analysis of efficient preconditioners for the interface operator to be used in the iterative solution of the interface solution. For the convection-diffusion problems, we shall discuss the effect of the convection term and its discretization on the performance of some of the preconditioners. For the driven cavity problem, we shall discuss the effectiveness of a class of boundary probe preconditioners.

Key Words. Parallel algorithms, domain decomposition, partial differential equations, preconditioned conjugate gradient, computational fluid dynamics, convection-diffusion problems, driven cavity problem.

1. Introduction. In the past several years, domain decomposition methods for solving elliptic partial differential equations have attracted much attention [19,5]. The main impulse for the enormous interest in these methods has come from the arrival of parallel computers. Besides the ease of parallelization, domain decomposition allows one to treat complex geometries or to isolate singular parts of the domain through adaptive mesh refinement. One of the goals of this paper is to give a very brief introduction to the subject.

The idea of domain decomposition has been used for quite some time in several scientific computing areas, such as computational structural mechanics (CSM) and computational fluid dynamics (CFD). However, in most of these applications, the coupling of the subdomains are handled in a rather primitive way. For example, in substructuring algorithms in CSM, the reduced interface equations are formed explicitly and solved by direct methods. Thus for problems where the degree of freedom on the interfaces is large, the solution of this reduced problem can actually dominate the overall solution process. On the other hand, in CFD domain decomposition ideas have been used primarily in adaptive generation of computational grids for complicated geometries. In these applications, the interface coupling is usually quite simple, such as using the most recent boundary values from a neighboring subdomain.

The more recent theoretical works have been mostly concerned with more sophisticated treatment of the coupling of the subdomain solutions. For second order elliptic problems, the theory and algorithms are quite well developed. However, applications of these newly developed algorithms to real physical problems are still rare. Moreover, extensions to more complicated operators, such as the Navier-Stokes equation,

* This version was last modified August 24, 1988. This paper is based on an invited talk presented at the Second International Conference on Parallel and Vector Processing, Tromso, Norway, June, 1988. It will appear in a special issue of Parallel Computing.

† Department of Mathematics, UCLA, Los Angeles, CA 90024. E-mail: chan@math.ucla.edu or na.tchan@na-net.stanford.edu. The research of the author is supported in part by The Department of Energy under contract DE-FG03-87ER25037, by the National Science Foundation under contract NSF-DMS87-14612, by the Army Research Office under contract No. DAAL03-88-K-0085, and by the Research Institute for Advanced Computer Science (RIACS), NASA Ames, under Cooperative Agreement NCC 2-387.

are only beginning to be studied. Much more work in this direction remains to be done. In this paper, we shall take a small step in this direction by presenting some results on the applications of some of these new domain decomposition algorithms to two model problems in computational fluid dynamics (CFD).

The first problem we shall investigate is the class of convection-diffusion problems in two dimensions. Most of the existing domain decomposition techniques have been derived with only the diffusion part of the operator in mind. Here we shall study the effect of the convection term (magnitude and direction) and the form of its discretization (central and upwind) on the effectiveness of the performance of these techniques. As we shall demonstrate, it is benefitial in practice to take into account the particular attributes of the convection term in constructing the domain decomposition algorithms.

The second problem we shall study is the steady two dimensional driven cavity problem. Specifically, we shall use the fourth order stream function formulation [33]. As mentioned earlier, most of the domain decomposition techniques have been developed for second order elliptic problems. It is therefore not immediately obvious how to apply these techniques to fourth order problems. Of course, for the driven cavity problem there are many solution algorithms which at each step require the solution of only second order problems and to which the appropriate domain decomposition algorithms can be applied. But true to the spirit of domain decomposition as a coarse granularity parallel algorithm, it is interesting to study algorithms which treat the original problem (rather than parts of a solution algorithm) by the domain decomposition approach. Such algorithms may be more flexible because the subdomain solves can be handled by any appropriate Navier Stokes solver. In this paper, we shall present some preliminary results on the use of *boundary probing* as a method for treating the coupling of the subdomain problems. The boundary probe technique, which was first introduced in [12] for second order problems, requires only solving the original problem on the subdomains with a few appropriately chosen "probing" boundary conditions. We shall discuss how to generalize these probing techniques to fourth order problems and present some numerical results for the driven cavity problem with Reynold's number 200. For work on domain decomposition algorithms for the Navier-Stokes equations in the velocity-pressure formulation, see [29,30,17].

The outline of the paper is as follows. In Section 2, we shall give a brief introduction to the various approaches of domain decomposition. In Section 3, we shall give a survey of domain decomposition *preconditioners* for the operator on the interface separating the subdomains, which is the main approach in this paper. The convection-diffusion problem will be treated in Section 4 and the driven cavity problem in Section 5.

2. Domain Decomposition Approaches. The main idea of domain decomposition algorithms is to decompose the original domain into smaller subdomains, solve the original problem on the subdomains, and somehow "patch" the subdomain solutions to form the solution to the original problem. In general, the above process has to be repeated through an iterative process until some convergence criteria is satisfied. There are two main approaches, characterized by the way the subdomains are constructed, namely overlapping and nonoverlapping.

The overlapping approach decomposes the original domain into two or more partially overlapping subdomains. A Schwarz alternating procedure [34] or a variant is then applied. Starting with an initial guess in a subdomain, a problem on a neighboring (overlapped) subdomain is solved, using the initial guess as part of the required

boundary conditions. This process is then repeated until a problem on the first subdomain is solved, giving an update on the initial guess. The overall iteration is then repeated until convergence. There are many variants. For example, the subdomains can be ordered in a such a way (e.g. in red/black fashion) that more than one subdomain solve can be performed in parallel. Also, relaxation parameters can be introduced to form weighted averages of the new guess with the old one. The iterates can be accelerated, say by the conjugate gradient method. Finally, the type of boundary conditions (e.g. Neumann versus Dirichlet) in the overlapped region could be chosen appropriately to speed up the rate of convergence. There are also a wealth of theory available for the Schwarz procedure, ranging from the conditions required for convergence (usually that the operator satisfies the maximum principle) to actual estimates of the rate of convergence as a function of the amount of overlap [34,25,27,28,35,22,11].

The nonoverlapping approach decomposes the domain into nonoverlapping subdomains by lower dimensional interfaces. The original problem is then reduced to an equivalent one posed on these interfaces. The reduced interface operator is usually not a local differential operator and is more nonlocal in nature, making it more difficult to solve efficiently by a direct method. Instead, it is most often solved iteratively. At each iteration, the action of the interface operator on an interface solution value has to be calculated, which turns out to require solves on the subdomains. Just like the overlapping approach, this iteration can also be accelerated, for example by the conjugate gradient method. A key factor in the iteration is the construction of effective preconditioners, which is essential to keep the number of iterations small. The technique can be extended to cases in which an exact subdomain solve is either not available or too expensive and only an approximate solution procedure is to be used. Since this is the main approach taken in this paper, we shall postpone a more detailed description to the next section.

It is natural to ask which of the two approaches is to be preferred in a given application. First, let us mention that it has been recently discovered that the two approaches are related; in fact they are identical under certain conditions [6]. Specifically, given a Schwarz overlapped iteration, there corresponds a nonoverlapped iteration, with a particular interface preconditioner, which produces exactly the same iterates on the interface. The appropriate preconditioners are precisely the exact reduced interface operator for the subdomains. For large class of second order elliptic operators (essentially separable ones) on rectangles, such preconditioners can be derived and implemented via Fast Fourier Transforms on the interfaces [14,13].

An issue that has often been raised concerning the efficiency of domain decomposition algorithms in a parallel implementation is whether they are actually more efficient than parallelizing a standard sequential algorithm. Part of the doubt arises because in the Schwarz procedure, a certain overhead is incurred due to the repeated solves on the overlapped regions. In fact, since the rate of convergence usually decreases exponentially when the amount of overlap is reduced, this overhead seems to be unavoidable. However, the nonoverlapped approach has no such overhead. Therefore, for problems for which such preconditioners can be used, the nonoverlapping approach is more efficient. The overlapped approach is, however, more generally applicable and rather robust. It will remain a main tool in this area.

One aspect that has generally been ignored is the gain in sequential computational complexity that domain decomposition can yield as a divide and conquer technique. When the work for solving a problem grows more than linearly with its size, splitting

it up in 2 subproblems of half the size will yield a faster method *provided* that the subsolutions can be efficiently combined to the solution of the original problem. Using the nonoverlapped approach with the boundary probing technique, we have been able to develop parallel domain decomposition algorithms which, in addition to the advantage of ease of parallelization, are actually faster than the corresponding sequential algorithms [7].

3. Interface Preconditioners. In this section, we shall briefly review some preconditioners which have been proposed for use in the nonoverlapped approach. For a more thorough survey, we refer the readers to [14,24].

We formulate this approach for the simplest case of a domain Ω split into two subdomains Ω_1 and Ω_2 sharing the interface Γ. Consider the problem $Lu = f$ on Ω with boundary conditions $u = u_b$ on $\partial\Omega$, where L is a linear second order elliptic operator. If we order the unknows for the internal points of the subdomains first and those on the interface Γ last, then the discrete solution vector $u = (u_1, u_2, u_3)^T$ satisfies the linear system :

$$Au \equiv \begin{pmatrix} A_{11} & & A_{13} \\ & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} \quad , \tag{2.1}$$

where the discrete vector $f = (f_1, f_2, f_3)^T$ contains the contribution of the right hand side of the differential equation and of the Dirichlet boundary condition.

System (2.1) can be solved by block Gaussian elimination which gives the equations for the interface variables u_3 :

$$Su_3 = \hat{f}_3 \quad , \tag{2.2}$$

with

$$S = A_{33} - A_{31}A_{11}^{-1}A_{13} - A_{32}A_{22}^{-1}A_{23}$$

and

$$\hat{f}_3 = f_3 - A_{31}A_{11}^{-1}f_1 - A_{32}A_{22}^{-1}f_2 \quad .$$

The matrix S is the Schur complement of A_{33} in the matrix A. It corresponds to the reduction of the operator L on Ω to an operator on the internal boundary Γ. Constructing the Schur complement would require the solution of n_Γ elliptic problems on each subdomain, where n_Γ is the number of internal points on Γ. Furthermore it is dense, so that factoring would be expensive.

Instead of solving the system (2.2) directly, iterative methods such as preconditioned conjugate gradient (PCG) can be applied in which only matrix vector product Sy are required. This product can be computed by one solve on each subdomain with boundary condition on Γ determined by y. Since each iteration is rather expensive, it is important to precondition this iteration with a good preconditioner in order to keep the number of iterations small.

Several preconditioners have been proposed in the literature. The first preconditioner was derived from the underlying properties of the "trace" of the differential operator on the interface. It is known that for a large class of second order elliptic operators, the reduced interface operator (the trace operator) is spectrally equivalent to the operator $M_D \equiv \sqrt{K}$, where K denotes the Laplace operator defined on the

interface and the square root is taken in the Fourier space [26]. This is true also for the discretized operator. Therefore M_D should make a reasonably good preconditioner for S, as first proposed by Dryja [18]. An improvement was made later by Golub-Mayers [20], who proposed as preconditioner the operator $M_G \equiv \sqrt{K + K^2/4}$. They arrived at this operator by considering the limiting case of the Laplace operator on semi-infinite planes. In numerical experiments in [20], M_G performs consistantly better than M_D in reducing the number of iterations.

A different class of preconditioner, called the Neumann-Dirichlet preconditioners, was proposed by Bjorstad-Widlund [1] following an earlier suggestion by Dryja. The main feature of these preconditioners is that they require alternatively solving problems on the subdomains with Neumann and Dirichlet boundary conditions on the interface. It may be easiest to understand these preconditioners from the point of view of symmetry. For if both the operator and the subdomains are symmetric about the interface, then the original problem can be reduced to one on one of the subdomains with a homogeneous (symmetric) Neumann boundary condition on the interface. Thus, the more symmetry there is about the interface, the better the preconditioner will perform. These preconditioners have also been recently extended to domain decomposed spectral methods [31]. We note that this preconditioner requires two different kinds of solvers on the subdomains, corresponding to Neumann and Dirichlet boundary conditions, which may be natural in some situations (such as finite element methods) but an inconvenience in other situations.

Another class of preconditioners was proposed by Chan [4]. The key idea is the observation that for many elliptic operators (essentially separable ones including all piecewise constant coefficient operators) on rectangular domains decomposed by an interface parallel to one of its sides, the exact interface operator can be derived analytically using Fourier transforms [10]. For nonrectangular domains, one can then use as preconditioner the exact interface operator of the *nearest rectangular approximation sharing the same interface.* Thus this class of preconditioners, which we shall denote by M_C, is based on the idea of geometric approximation and therefore can be shown to be less sensitive to the aspect ratios (i.e. the relative shape) of the adjoining subdomains [4]. In fact, for certain geometrical shapes such as L-shaped and C-shaped domains, it can be proven that using this preconditioner, the condition number of the preconditioned interface operator can be bounded by a small constant (around 2) independent of the grid size and the particular shape of the domain [15,11].

One of the problems with preconditioners derived from general properties of the differential operators, such as M_D and M_G, is that they cannot be expected to perform uniformly well for any particular operator. In particular, they could be sensitive to both the shape of the domain and the variability of the coefficients. A preconditioner that is designed to adapt well to these effects is the class of Boundary Probe Preconditioners [12]. Additional extensive experiments with this preconditioner have been performed in [24], where it was called the Modified Schur Complement method. The main motivation for this approach is the observation that, for many elliptic operators, the magnitude of elements of the matrix S decay rapidly away from the main diagonal [20], reflecting a weak global coupling among the interface unknowns. It is therefore reasonable to consider a k diagonal approximation to S. However, it would not be efficient to calculate the elements of S in order to do this. Instead, as proposed in [12], a $2k+1$ diagonal approximation to S can be constructed by computing the action of S on $2k+1$ "probing" vectors $v_j, j = 1, \ldots, 2k+1$. The idea is motivated by sparse Jacobian evaluation techniques [16]. For the case $k = 0$ and $k = 1$ the probing

vectors are the following :

$$\begin{aligned} k=0: \quad & v_1 = (1,1,1,1,1,1,1,\ldots)^T \\ k=1: \quad & v_1 = (1,0,0,1,0,0,1,\ldots)^T \\ & v_2 = (0,1,0,0,1,0,0,\ldots)^T \\ & v_3 = (0,0,1,0,0,1,0,\ldots)^T. \end{aligned}$$

The case $k = 0$ corresponds to a scaling of each row of the matrix S by the sum of the elements of the row. For $k = 1$, if S were indeed tridiagonal, all of its elements would be recovered in the vectors Sv_j, $j = 1, 2, 3$. The idea can be generalized to cases with $k > 1$ and multiple domains [23,9] and to wider discretization stencils, as for instance in fourth order equations such as the biharmonic equation [3] and the steady Navier Stokes equation (see Section 5). Applications to coupled systems such as reaction-convection-diffusion problems are also possible [23].

Finally, we consider cases where the subdomain solves are too expensive to perform exactly (e.g. a fast direct solver is not available). One approach is to consider the PCG iteration on the Schur complement as a combination of an outer and an inner iteration [21]. Another approach is to combine preconditioners on the subdomains and on the interface to form a preconditioner on the whole domain, and then to iterate on the subdomains and the interface simultanuously. A simple way to achieve this is through the following block factorization of A :

$$A = \begin{pmatrix} A_{11} & & \\ & A_{22} & \\ A_{31} & A_{32} & S \end{pmatrix} \begin{pmatrix} I & & A_{11}^{-1}A_{13} \\ & I & A_{22}^{-1}A_{23} \\ & & I \end{pmatrix} \quad . \tag{2.3}$$

A preconditioner for A can be derived by replacing A_{ii} in (2.3) by approximations B_{ii} and replacing the Schur complement by a preconditioner M. For the latter, we can take any of the preconditioners that were mentioned earlier. We therfore arrive at the following preconditioner for A :

$$\widetilde{M} = \begin{pmatrix} B_{11} & & \\ & B_{22} & \\ A_{31} & A_{32} & M \end{pmatrix} \begin{pmatrix} I & & B_{11}^{-1}A_{13} \\ & I & B_{22}^{-1}A_{23} \\ & & I \end{pmatrix} \quad . \tag{2.4}$$

Preconditioners of this form were first proposed in [2] and were also mentioned in [14], [1].

4. Convection-Diffusion Problems. In this section, we shall consider two dimensional convection-diffusion operators of the form:

$$Lu \equiv \Delta u + \alpha \frac{\partial u}{\partial x} + \beta \frac{\partial u}{\partial y},$$

with Dirichlet boundary conditions. In particular, we are concerned with constructing efficient domain decomposition preconditioners for the reduced interface operator derived from L. As we have seen from the previous section, there are many preconditioners available for the Laplacian operator and in fact two of these can be directly derived from it (M_D and M_G.) In theory, these preconditioners for the Laplacian should work well in the presence of the convection terms as well. Specifically, the first order convection terms do not affect the spectral equivalence properties. In the

discrete case, this means that if the coefficients α and β are kept fixed while we let a mesh size parameter h go to zero, the condition number (in the spectral norm) can be bounded by a constant independent of h. In practical computations, however, h is bounded from below by both memory and time limitations and the relevant parameter to consider are the cell Reynold's numbers $r_x = \alpha h$ and $r_y = \beta h$. It is well known that for central differencing the discrete solution exhibits oscillatory behaviour when the cell Reynold's numbers exceed the critical value of 2. Moreover, the direction of the flow (determined by α and β) relative to the interfaces may have an effect on the choice of preconditioners because the coupling between the subdomains is affected by the amount of information carried by the flow from one subdomain into the other. Related to this is the effect of the form of discretization, in particular central difference versus upwind difference, on the performance of preconditioners.

To test the effect of the first order term, we compute the condition numbers (in the l_2-norm) in the case when we use only the diffusion operator to construct preconditioners. In Figures 1 and 2, we plot the condition numbers versus the coefficient a in the following two model equations:

$$\Delta u + a\frac{\partial u}{\partial y} = 0 \tag{4.1}$$

and

$$\Delta u + a\frac{\partial u}{\partial x} = 0. \tag{4.2}$$

We use $C(a)$ to represent the Schur complement corresponding to equation (4.1) or (4.2) and $C(0)$ represent the Schur complement for the Poisson equation. In Figure 1, we plot the condition number $K(C^{-1}(0)C(a))$ for equation (4.1) using both central and upwind differencing, for the rectangular $(0,1) \times (0,3)$ with an interface joining the points $(0,2)$ and $(1,2)$ and with a grid size of $h = 0.02$. Figure 2 displays the same computation for equation (4.2). Note that for this value of h, the critical cell Reynold's number corresponds to $a = 100$ for the central differencing case.

These calculations show that the condition number can grow appreciably above 1 as a increases, implying that the preconditioner based on only the diffusion operator may give very slow convergence when the cell Reynolds number is of order $O(1)$. By comparing Figures 1 and 2, we see that the growth is more rapid for (4.1) than for (4.2). In other words, it is more crucial to have a good preconditioner when the flow direction is perpendicular to the interface than when it is parallel to it. Intuitively, in the former case there is a stronger coupling of the subdomains due to information carried by the flow from one domain into the other. Moreover, in both cases, the condition number for central differencing grows faster than that for upwind differencing for cell Reynold's number close to the critical value. These results seem to indicate that upwind differencing is less affected by the lack of a good preconditioner than central differencing.

In addition to the condition number, the eigenvalue distribution of the preconditioned system also plays an important role in the effectiveness of the preconditioner. In Figures 3 to 6, these eigenvalues are plotted for several values of a for the same computation described earlier. Note that the corresponding eigenvalues for (4.2) are complex because the corresponding capacitance matrix is nonsymmetric. These plots show how rapidly the spectrum spreads from unity as a increases. It is interesting to note that the plots for upwind differencing in Figures 3, 5 and for central differencing

in Figures 4, 6 are quite different qualitatively; in the former the clustering of the spectrum shifts to the right when a increases whereas in the latter the clustering remains around unity. In all cases, clustering around the value 1 can be seen, but even this effect weakens as a grows.

From the above numerical results, it is clear that for problems with a sizable cell Reynold's number, the information carried in the convection terms should be taken into account when constructing preconditioners for the interface operator. We shall discuss two ways of achieving this.

The first is to generalize the preconditioner M_G to convection-diffusion operators. It turns out that for rectangular domains, the effects of the convection terms can be incorporated *exactly*. In [10], exact eigen-decompositions (which can be inverted efficiently using FFTs) for the interface operator S are derived for general constant coefficient five point discrete elliptic difference equations, including in particular the convection-diffusion operators of the form L, posed on rectangular domains decomposed into two smaller rectangles. For rectangular domains decomposed into mutiple smaller rectangles, the technique can be easily extended to derive fast direct solvers. For non-rectangular domains or variable coefficient problems, they can be adapted to construct efficient preconditioners which incorporate the effects of the convection terms.

Another approach is to use the boundary probe preconditioners, which automatically captures the effects of the convection terms. Some success has been reported in the numerical experiments in [23] for convection-diffusion problems and also in the experiments to be presented later in Section 5 for the driven cavity problem. The key to the success of the boundary probing technique is a weak global coupling of the interfacial unknowns. The numerical experiments seem to indicate that this property holds at least for the moderate cell Reynold's numbers used. Further analysis to determine the effect of the convection term on this property is needed. An important factor could be the direction of the flow relative to the interface.

5. The Driven Cavity Problem. In this section, we consider the two dimensional driven cavity problem of incompressible flow. We shall use the fourth order stream function formulation:

$$L(\phi) \equiv \Delta^2\phi - RG(\phi) = 0,$$

where

$$G(\phi) \equiv \phi_y(\Delta\phi_x) - \phi_x(\Delta\phi_y). \tag{5.1}$$

The classical driven cavity problem is posed on $\Omega = (0,1) \times (0,1)$ with boundary conditions : $\phi = 0$ on $\partial\Omega$, $\phi_x = 0$ if $x = 0, 1$, $\phi_y = 0$ if $y = 0$ and $\phi_y = 1$ if $y = 1$. We shall follow [33] and use a 13-point central difference discretization of (5.1) on a n by n uniform mesh. Consider the use of the nonoverlapping Schur complement approach of Section 3 in a domain decomposition algorithm for solving the discrete problem. For simplicity, we shall consider only two subdomains, separated by an interface at $y = 0.5$. We are interested in the use of the boundary probing technique described in Section 3 for constructing efficient preconditioners for the interface operator for the discrete problem. There are several modifications necessary in order to extend the boundary probing technique developed for second order problems. Some of these techniques were considered for the biharmonic equation in [3]. Here we consider extending them to the Navier-Stokes problem.

First, since the difference stencil is wider than the corresponding 5-point discretization for second order problems, the interface must consists of two rows of grid points, say Γ_1 and Γ_2, in order to completely decouple the subdomain problems. The Schur complement system corresponding to (2.2) can be written as a block 2 by 2 system:

$$S \begin{pmatrix} \phi^1 \\ \phi^2 \end{pmatrix} \equiv \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix} \begin{pmatrix} \phi^1 \\ \phi^2 \end{pmatrix} = \begin{pmatrix} f^1 \\ f^2 \end{pmatrix}, \tag{5.2}$$

where ϕ^1 and ϕ^2 denote the unknowns on Γ_1 and Γ_2 respectively.

We now consider a second modification of the boundary probing technique in order to construct an efficient preconditioner for S. The basic idea is still to capture the strong local coupling and weak global coupling of the intefacial unknowns. To see how this can be achieved, note that the blocks S_{11} and S_{22} account for the coupling of the unknowns on Γ_1 and Γ_2 respectively among themselves and the blocks S_{12} and S_{21} account for the coupling between the unknowns on the two interfaces. Thus, if we number the unknowns on the two interfaces in a spatially consistent manner, we can still expect the individual subblocks of S to exhibit the property that the magnitude of their elements decay rapidly away from the respectively main diagonals. For example, Figure 7 shows a plot of the elements of the matrix S for the case $n = 30$, where the unknowns on both interfaces are ordered from left to right. The decay property can be seen clearly in the figure. Moreover, the elements of the main diagonal blocks S_{11} and S_{22} are negligible except for the 5 main diagonals. Similarly, the off diagonal blocks S_{12} and S_{21} have only 3 non-negligible main diagonals.

A simple way to construct an interface preconditioner is therefore to use the boundary probing technique on the individual blocks of S. For example, suppose we want to compute a preconditioner M_{kl} consisting of k-diagonal approximations for the diagonal blocks S_{11} and S_{22}, and l-diagonal approximations for the off-diagonal blocks S_{12} and S_{21}. Let V_k be a n by k matrix consisting of k probing vectors for either one of the two interfaces as described in Section 3. Then M_{kl} can be obtained by probing S by the columns of the following matrix:

$$\begin{pmatrix} V_k & V_l & 0 & 0 \\ 0 & 0 & V_k & V_l \end{pmatrix}.$$

This requires solving subdomain problems with boundary conditions consisting of probing vectors from V_k or V_l on one grid line and zero on the other grid line. More efficient probing techniques, with fewer probing vectors and hence subdomain solves for given values of k and l, can be constructed but for simplicity we shall not present them here.

Finally, the block matrix M_{kl} can be permuted into a narrow banded matrix by reordering the unknowns to preserve their physical proximity. For example, if we start from the left and alternatively order the unknowns on the two grid lines, then M_{kl} is reordered into a banded matrix with bandwidth $2k - 1$, assuming $l < k$ for simplicity. Therefore, the product $M_{kl}^{-1} w$ for a given interfacial vector w can be computed efficiently by banded Gaussian elimination.

We now present some numerical results for the performance of the above boundary probing techniques on the driven cavity problem for Reynold's number $R = 200$ and $n = 30$. Figures 8 and 9 show the eigenvalue distribution of the unpreconditioned interface matrix S and the preconditioned matrix $M_{53}^{-1} S$ respectively, where S corresponds to the interface operator for the Jacobian matrix A of L at the solution.

Note that the matrices A, S and M are all nonsymmetric and hence the eigenvalues are complex in general. The figures show that the preconditioner M_{53} produces a dramatic improvement in the conditioning of the interface operator. As a sample measure, the real parts of the eigenvalues of S lie in the interval $(.0083, 45)$ while those of the preconditioned system lie in $(0.2, 1.8)$. Moreover, many eigenvalues of the preconditioned system are clustered around the point $(0, 1)$. Figure 10 shows the condition number $M_{kl}^{-1}S$ in the spectral norm as a function of n for several values of k and l. These results show that not only are the condition numbers of the preconditioned matrix much lower than S itself, but also that they grow at a slower rate. The plots also show that M_{53} is in some sense optimal because the more expensive M_{75} produces negligible improvement in the condition numbers. Finally, to show that the improvement in the condition number and the eigenvalue distribution of the preconditioned matrix does improve the performance in an iterative solution of the interfacial unknowns, we solve the interfacial system by the GMRES algorithm [32]. Figure 11 shows the history of an iteration, with the norm of the residual plotted against the iteration step. It is clear that M_{53} produces a much faster convergence rate.

The above are only preliminary evidence that the boundary probing technique can be applied successfully to Navier-Stokes problems. Much further work needs to be done, especially concerning the decay rates of the elements of S and the properties of the preconditioners derived from it.

Acknowledgement. Several topics in this paper represent joint work with Diana Resasco, Tom Hou, Danny Goovaerts and WaiKin Tsui. The author would like to acknowledge their contributions.

REFERENCES

[1] P.E. Bjørstad and Olof Widlund, *Iterative Methods for the Solution of Elliptic Problems on Regions Partitioned into Substructures*, SIAM J. Numer. Anal., 23/6, (1986), pp. 1097-1120.

[2] J.H. Bramble, J.E. Pasciak and A.H. Schatz, *The Construction of Preconditioners for Elliptic Problems by Substructures*, Math. Comp., 46 (1986), pp. 361-369.

[3] Tony F. Chan, *Boundary Probe Preconditioners for Fourth Order Elliptic Problems*, in Proceedings of Second Int'l Symposium on Domain Decomposition Methods, Jan, 1988, Los Angeles, Calif., T.F. Chan et al (editors), SIAM Publ., to appear.

[4] Tony F. Chan, *Analysis of Domain Decomposition Preconditioners*, SIAM J. of Numer. Anal., 24/2 (1987), pp. 382-390.

[5] Tony F. Chan, Roland Glowinski, Jacques Periaux and Olof Widlund (editors), *Domain Decomposition Algorithms*, Proceedings of the 2nd International Symposium on Domain Decomposition Methods, Jan 14-16, 1988, UCLA, SIAM Publ., to appear.

[6] Tony F. Chan and Danny Goovaerts, *Schwarz=Schur: Overlapping Versus Nonoverlapping Domain Decomposition*, UCLA Comp. and App. Math. Report 88-21, August 1988.

[7] Tony F. Chan and Danny Goovaerts, *Domain Decomposition Benefitial Even Sequentially*, UCLA Comp. and App. Math. Report 88-18, June 1988.

[8] Tony F. Chan and Danny Goovaerts, *Schur Complement Domain Decomposition Algorithms for Spectral Methods*, UCLA Comp. and App. Math. Report 88-23, August, 1988.

[9] Tony F. Chan and Danny Goovaerts, *Domain Decomposition Methods with Inexact Subdomain Solves*, UCLA Comp. and App. Math. Report, to appear.

[10] Tony F. Chan and Thomas Y. Hou, *Domain Decomposition Preconditioners for General Second Order Elliptic Problems*, UCLA Comp. and App. Math. Report 88-16, June, 1988.

[11] Tony F. Chan, Thomas Y. Hou and P.L. Lions, *Geometry Independent Convergence Results for Domain Decomposition Algorithms*, UCLA Comp. and App. Math. Report, to appear.

[12] Tony F. Chan and Diana Resasco, *A Survey of Preconditioners for Domain Decomposition*, in Proceedings of the IV Coloquio de Matematicas, TAller de Analisis Numerico y sus Aplicaciones, Taxco, Guerrero, Mexico, August, 1985, to be published by Springer Verlag. Also appeared as Yale Univ. Dept. of Comp. Sci. Report YALEU/DCS/RR-414, July, 1985.

[13] Tony F. Chan and Diana Resasco, *A Domain Decomposed Fast Poisson Solver on a Rectangle*, SIAM J. Sci. Stat. Comp., 8/1 Jan (1987), pp. s14-s26.

[14] Tony F. Chan and Diana Resasco, *A Framework for the Analysis and Construction of Domain Decomposition Preconditioners*, in "Domain Decomposition Methods for Partial Differential Equations", Proceedings of First Int'l Symposium on Domain Decomposition Methods, Jan, 1987, Paris, France, R. Glowinski et al (editors), SIAM Publ., 1988, pp. 217-230.

[15] Tony F. Chan and Diana Resasco, *Analysis of Domain Decomposition Preconditioners on Irregular Regions*, in "Advances in Computer Methods for Partial Differential Equations, VI", R. Vichnevetsky and R.S. Stepleman (Editors), Publ. IMACS, 1987, pp. 317-322.

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

[17] Q.V. Dinh, B. Mantel, J. Periaux and R. Glowinski, *Approximate Solutions of the Navier-Stokes Equations for Incompressible Viscous Fluids. Related Domain Decomposition Methods.*, Lecture Notes in Math., 1005, (1982), pp.46-86, Springer Verlag.

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

[19] Roland Glowinski, G.H. Golub, G.A. Meurant and J. Periaux, *Domain Decomposition Methods for Partial Differential Equations*, Proceedings of First Int'l Symposium on Domain Decomposition Methods, Jan, 1987, Paris, France, R. Glowinski et al (editors), SIAM Publ., 1988.

[20] G.H. Golub and D. Mayers, *The Use of Preconditioning Over Irregular Regions*, in "Computing Methods in Applied Sciences and Engineering VI", R. Glowinski and J.L. Lions (Editors), North Holland (1984), pp. 3-14.

[21] G. H. Golub and M.L. Overton, *The Convergence of Inexact Chebyshev and Richardson Iterative Methods for Solving Linear Systems*, Research Report CMA-R46-86, Centre for Math. Anal., Australian Nat'l Univ., Dec. 1986.

[22] Li-shan Kang, editor, *Parallel Algorithms and Domain Decomposition*, Wuhan Univ. Press, 1987.

[23] D. Keyes and W. Gropp, *Domain Decomposition Techniques for Nonsymmetric Systems of Elliptic Boundary Value Problems : Examples from CFD*, in Proceedings of Second Int'l Symposium on Domain Decomposition Methods, Jan, 1988, Los Angeles, Calif., T.F. Chan et al (editors), SIAM Publ., to appear.

[24] D. Keyes and W. Gropp, *A Comparison of Domain Decomposition Techniques for Elliptic Partial Differential Equations*, SIAM J. Sci. Stat. Comp., 8/2 March (1987), pp. s166-s202.

[25] Keith Miller, *Numerical Analogs to the Schwarz Alternating Procedure*, Numer. Math. 7, pp. 91-103, 1965.

[26] J.L. Lions and E. Magenes, *Non-homogeneous Boundary Value Problems and Applications*, Vol. 1-3, Springer Verlag, (1972, 1973).

[27] P.L. Lions, *On the Schwarz Alternating Method. I*, in Proceedings of First Int'l Symposium on Domain Decomposition Methods, Jan, 1987, Paris, France, R. Glowinski et al (editors), SIAM Publ., 1988, pp. 1-42.

[28] P.L. Lions, *On the Schwarz Alternating Method. II*, in Proceedings of Second Int'l Symposium on Domain Decomposition Methods, Jan, 1988, Los Angeles, Calif., T.F. Chan et al (editors), SIAM Publ., to appear.

[29] J. Pasciak, *Two Domain Decomposition Techniques for the Stokes Problems*, in Proceedings of Second Int'l Symposium on Domain Decomposition Methods, Jan, 1988, Los Angeles, Calif., T.F. Chan et al (editors), SIAM Publ., to appear.

[30] Alfio Quarteroni, *Domain Decomposition Algorithms for the Stokes Equations*, in Proceedings of Second Int'l Symposium on Domain Decomposition Methods, Jan, 1988, Los Angeles, Calif., T.F. Chan et al (editors), SIAM Publ., to appear.

[31] Alfio Quarteroni and G. Sacchi-Landriani, *Domain Decomposition Preconditioners for the Spectral Collocation Method*, ICASE Report No. 88-11, NSA Langley Research Center, Hampton, Virginia 23665, January 1988.

[32] Y. Saad and M. Schultz, *GMRES : A Generalized Minimum Residual Algorithm for Solving Nonsymmetric Linear Systems*, SIAM J. Sci. Stat. Comp., 7 (1986), pp. 856-869.

[33] R. Schreiber and H.B. Keller, *Driven Cavity Flows by Efficient Numerical Techniques*, J. Comp. Phys., pp. 310-333, 1985.

[34] H.A. Schwarz, *Über einige Abbildungsaufgaben*, Ges. Math. Abh., 11 (1869), pp. 65-83.

[35] W.P. Tang, *Schwarz Splitting and Template Operators*, Ph.D. thesis, Stanford Univ., 1987.

Figure 1. K(inv(C(0))C(a)) vs a for Uxx + Uyy + a Uy = 0, h=.02

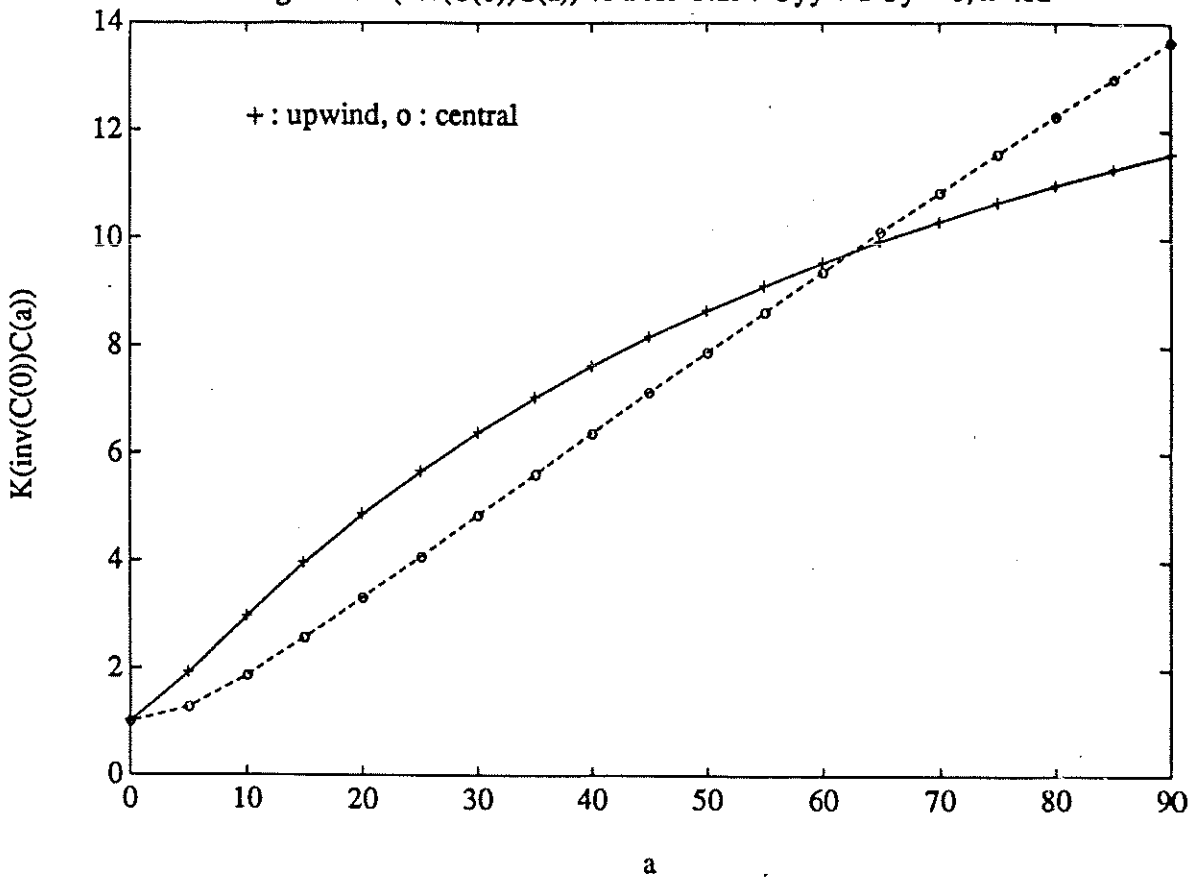

Figure 2. K(inv(C(0))C(a)) vs a for Uxx + Uyy + a Ux = 0, h=.02

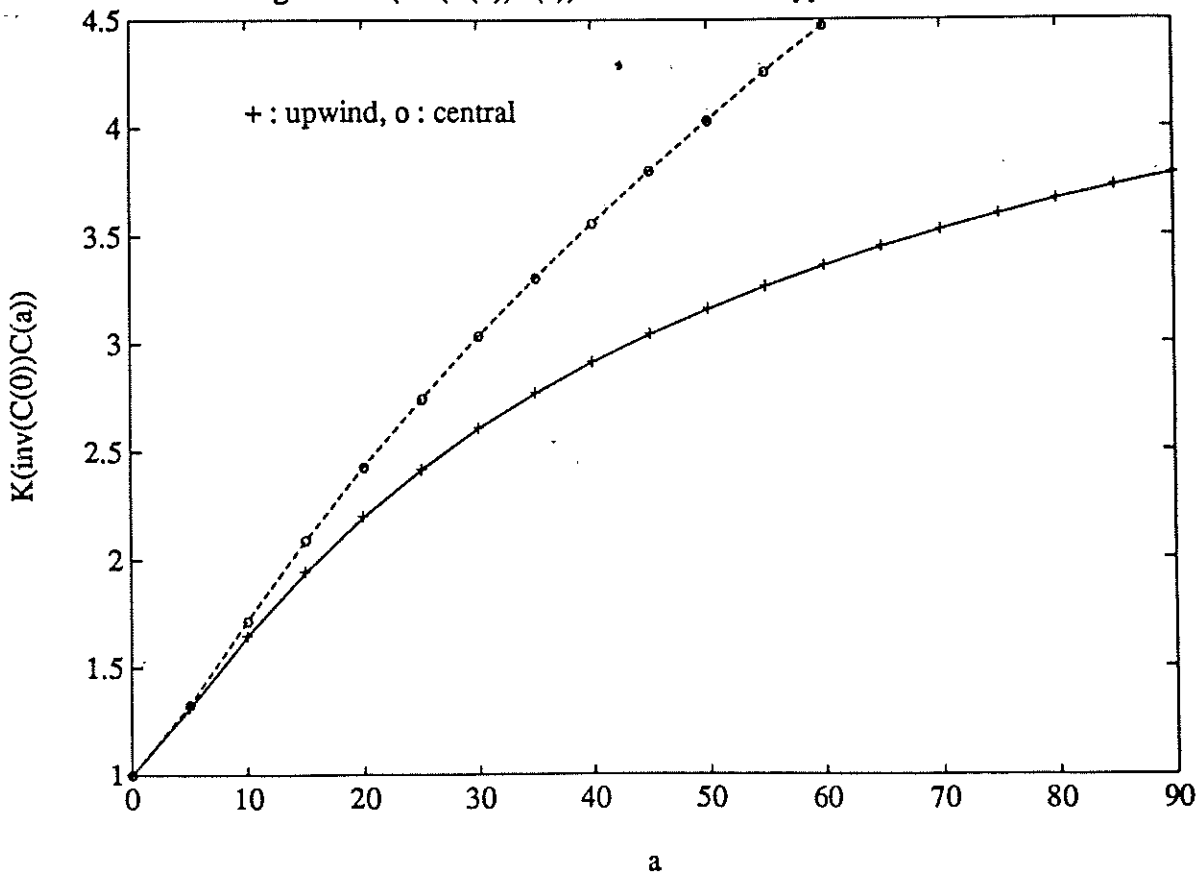

Figure 3. Eigenvalue distributions for upwind difference for Uxx + Uyy + aUx = 0; h=0.02

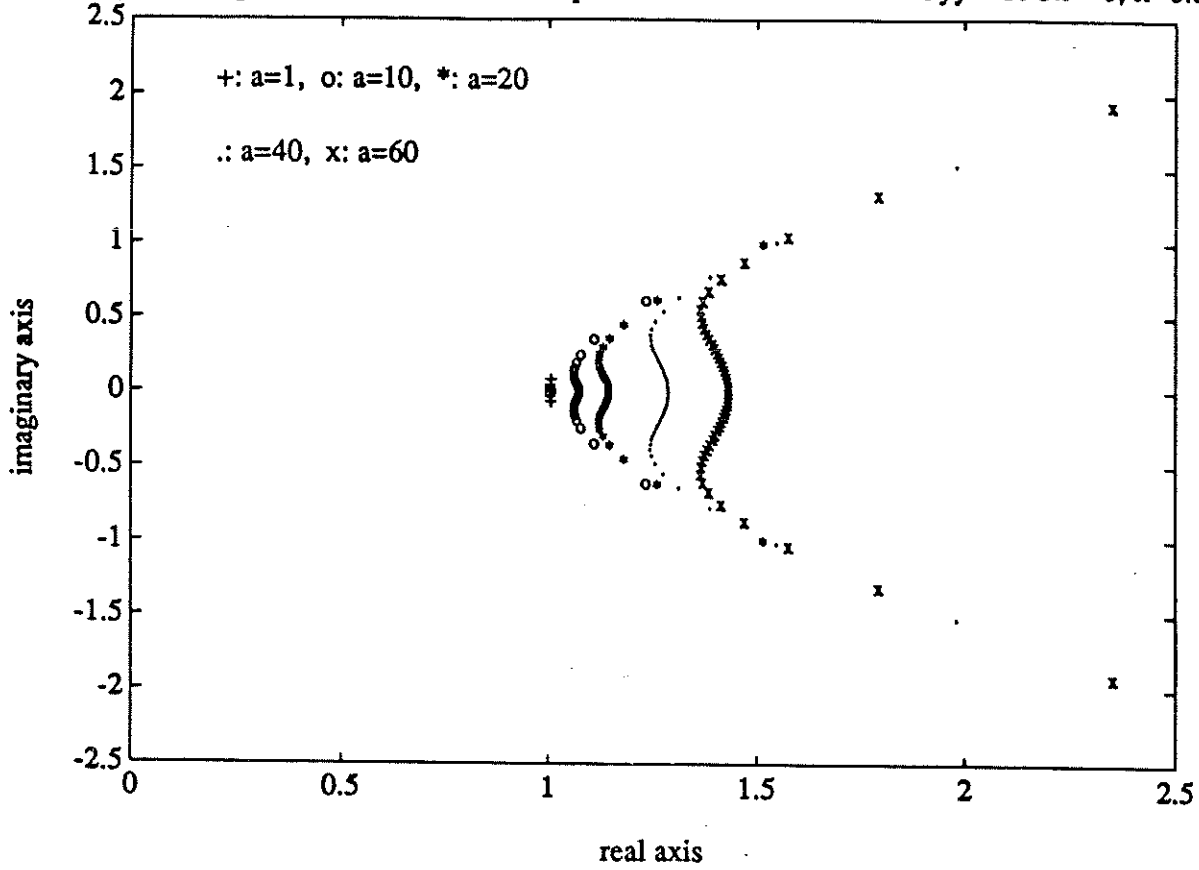

Figure 4. Eigenvalue distributions for central difference for Uxx + Uyy + aUx = 0, h=0.02

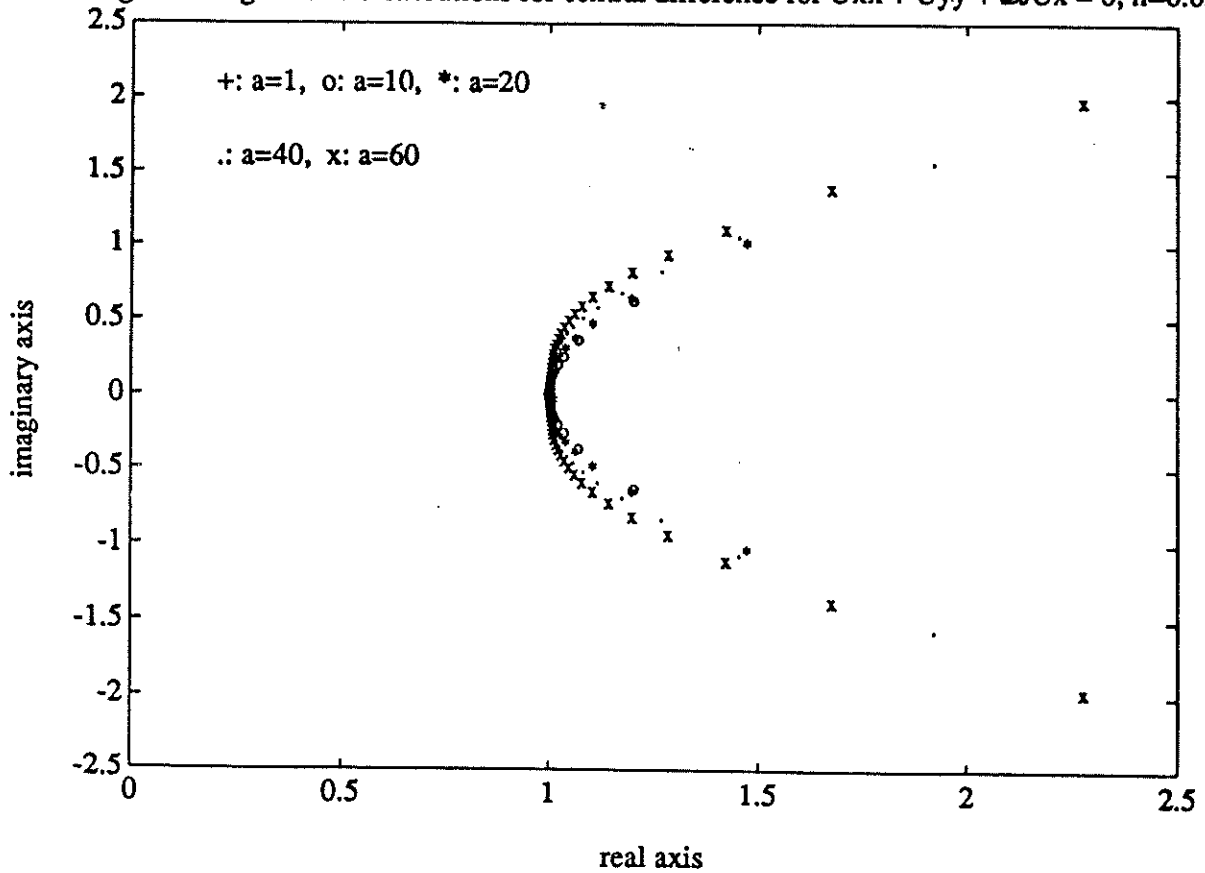

Figure 5. Eigenvalues for upwind diff. for Uxx+Uyy+a Uy=0, h=0.02

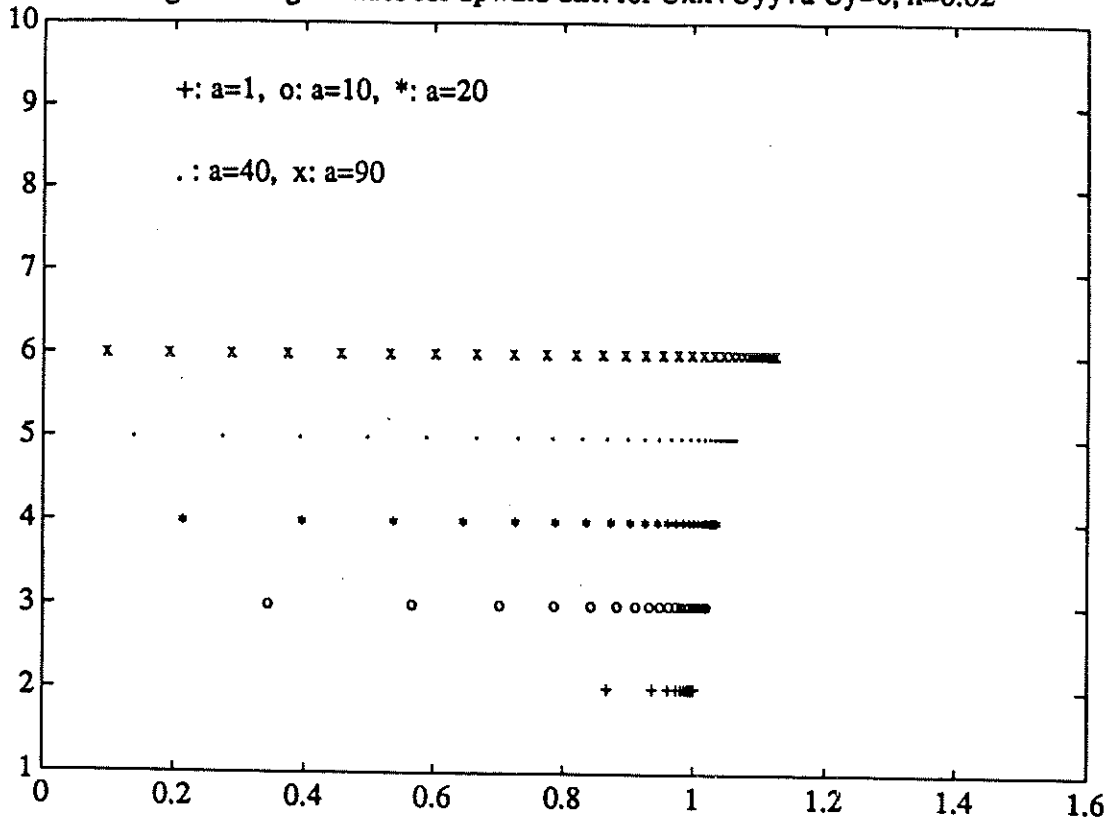

Figure 6. Eigenvalues for central diff. for Uxx+Uyy+a Uy=0, h=0.02

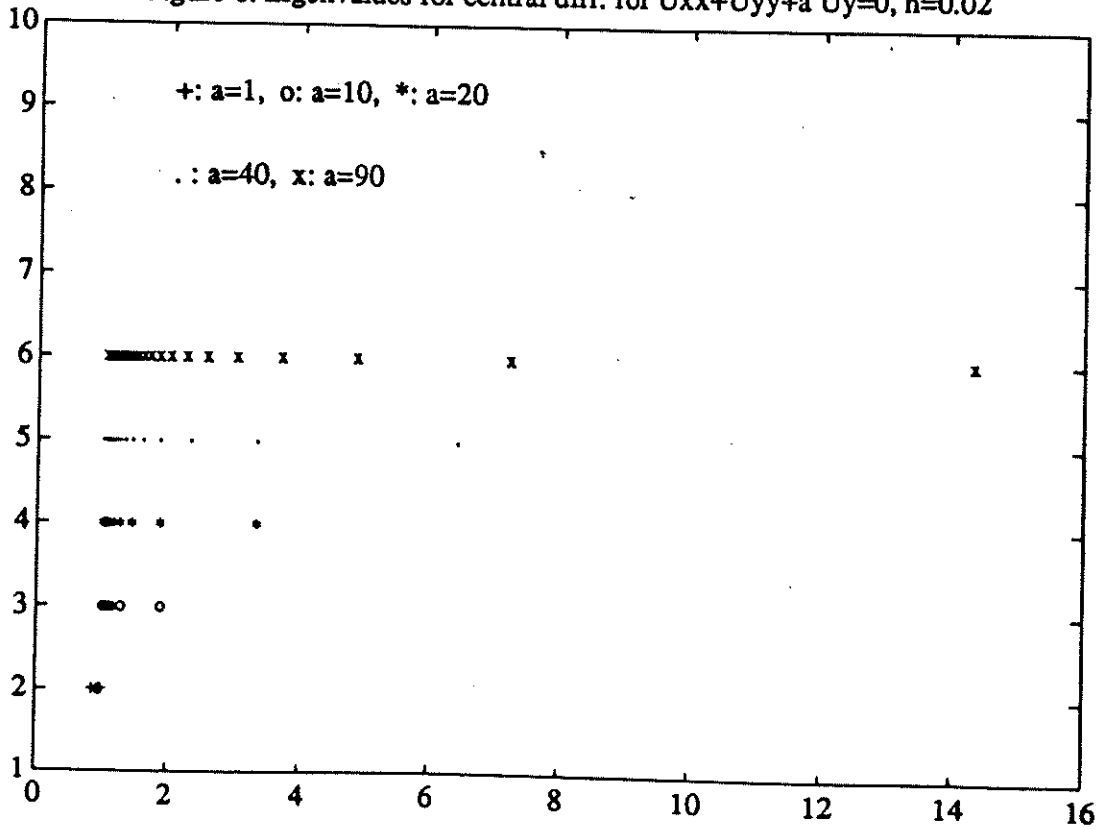

FIGURE 7

Plot of S, n=30, R=200

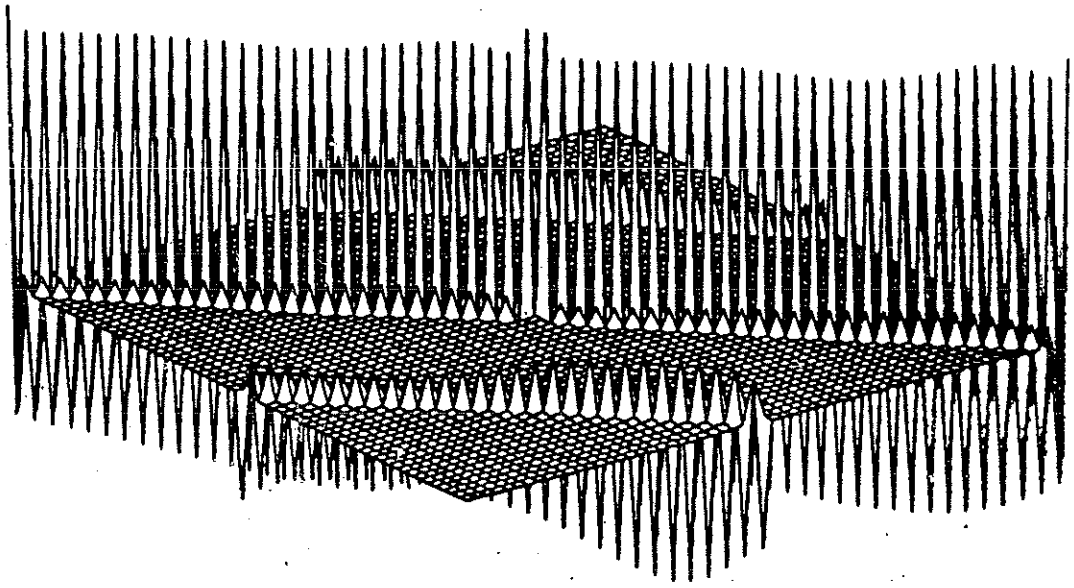

FIGURE 8

Eigenvalue Distribution of S, n=30

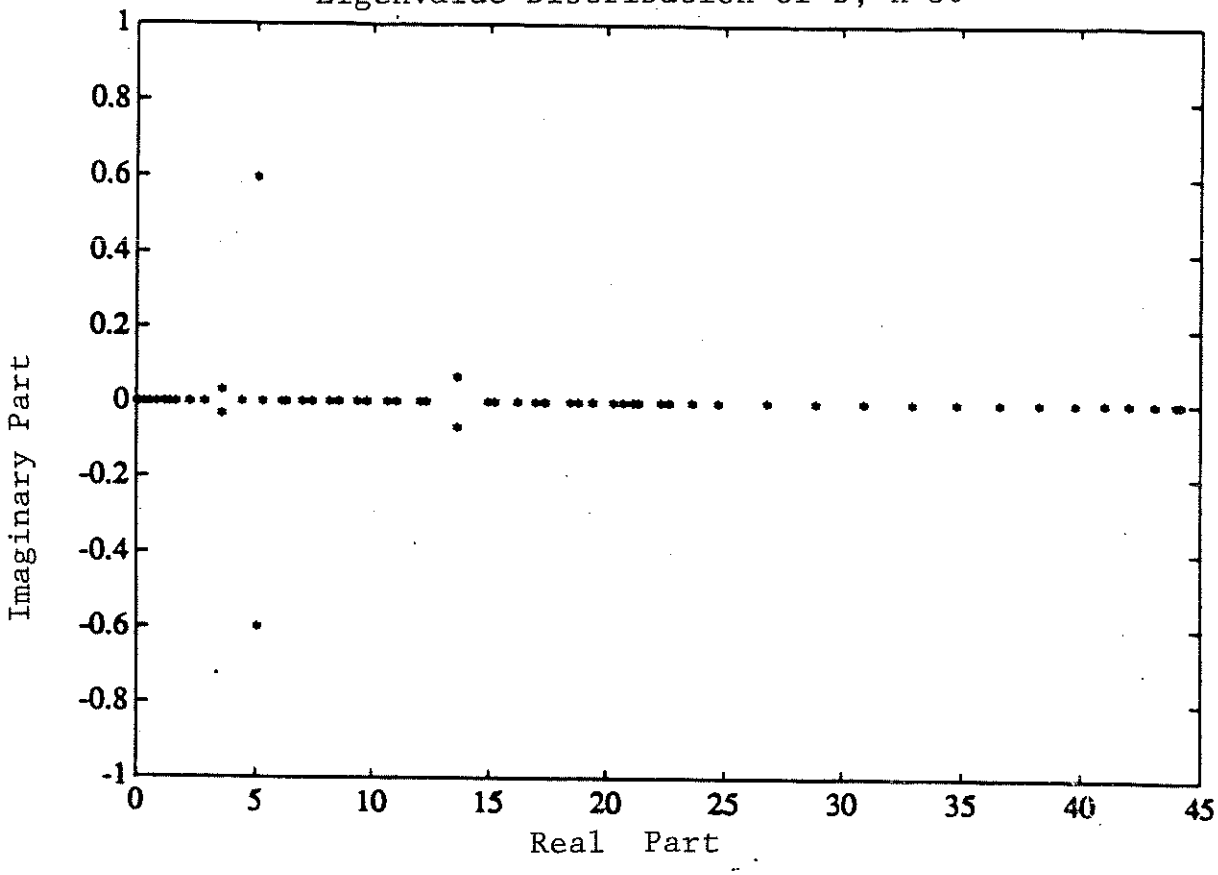

FIGURE 9

Eigenvalue Distribution of $M_{53}^{-1}S$, n=30

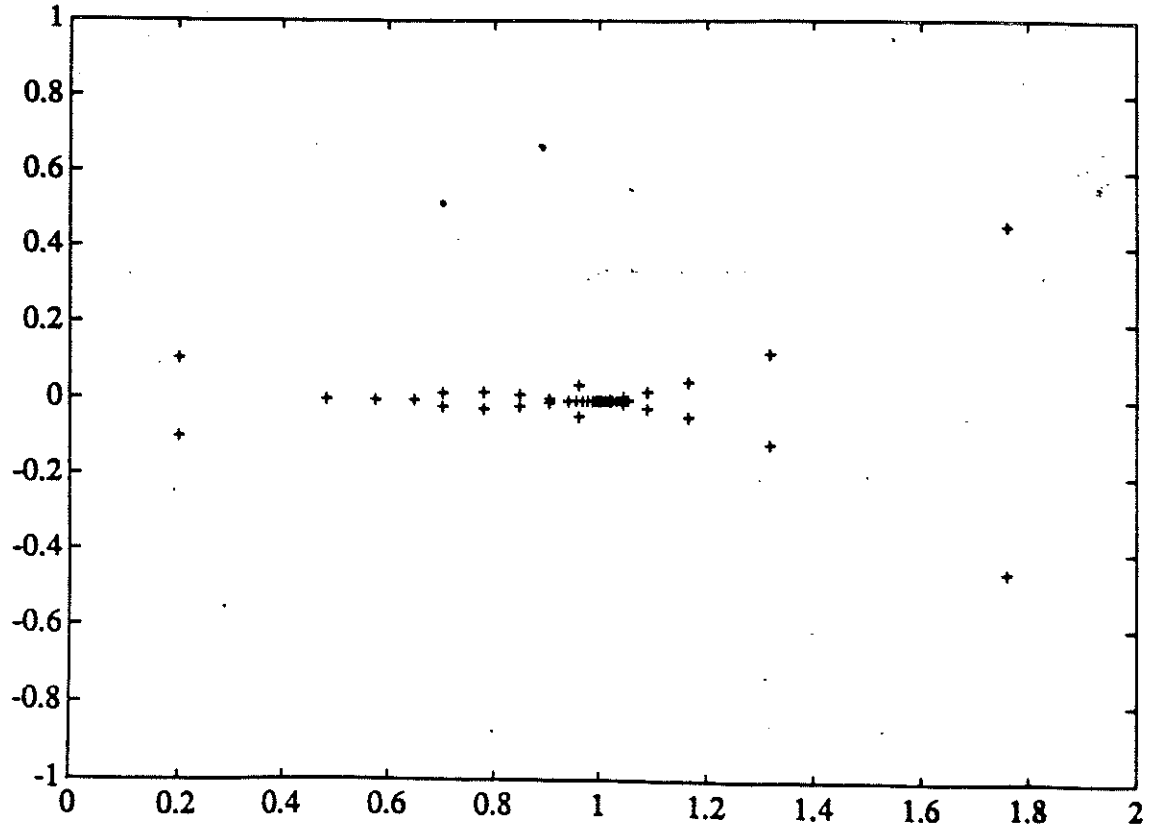

FIGURE 10

Condition Number $K(M^{-1}S)$ vs n, R=200

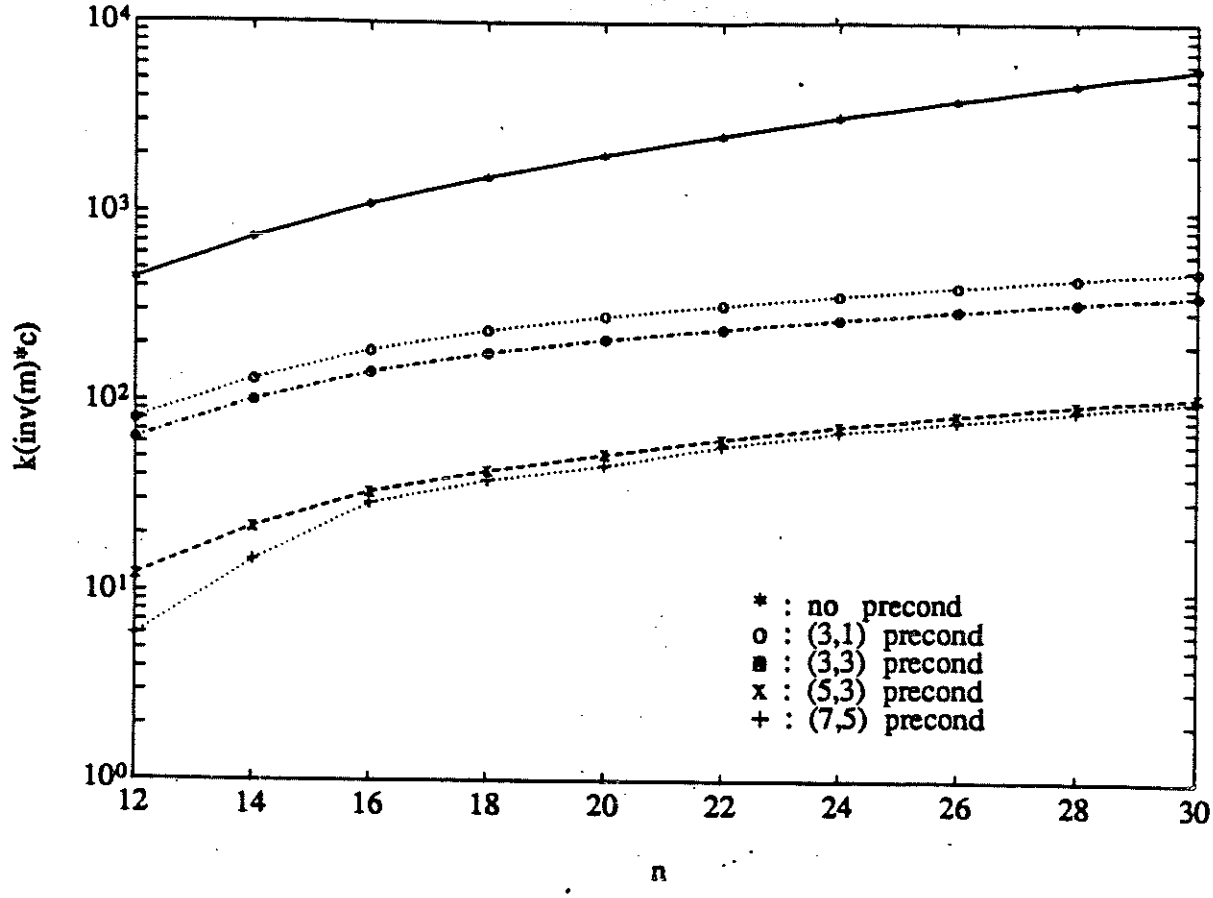

FIGURE 11

GMRES Convergence history,n=30,(5,3)"x",(3,3)"o",no precond"*"