A Framework for the Analysis and Construction of Domain Decomposition Preconditioners*

TONY F. CHAN† AND DIANA C. RESASCO‡

Abstract: Domain Decomposition is a class of techniques for the solution of partial differential equations on a domain by solving smaller problems on subdomains. They are particularly useful for solving problems on irregular domains and on parallel computers. The key ingredient is the system of equations governing the variables on the interfaces between the subdomains, which is often solved by preconditioned iterative methods. Since each iteration involves solving problems on each subdomain, it is essential to keep the number of iterations low by using a good preconditioner. In this paper, we present a framework for analyzing and constructing such efficient preconditioners. We use two approaches. The first is based on spectral analysis and can be used to invert exactly the interface operator for general piecewise constant coefficient elliptic operators on rectangular regions in any dimension. Methods for adapting these techniques to nonconstant coefficient problems and irregular domains will be discussed. The second approach is based on treating the interface operator as a localised pseudo-differential operator on the interface unknowns and is applicable to more general operators than the spectral approach. One of our objectives is to illuminate the relationships among the most common preconditioners in the literature.

*The authors were supported in part by the Department of Energy under contracts DE-FG03-87ER25037 at UCLA and DE-AC02-81ER10996 at Yale. The second author was also supported by a BID-CONICET fellowship from Argentina.
†Dept. of Mathematics, University of California, Los Angeles, CA 90024.
‡Dept. of Computer Science, Yale Univ., New Haven, CT 06520.
1. Introduction

The term Domain Decomposition generally refers to a class of techniques for solving partial differential equations on a given domain by first decomposing the domain into smaller ones and then obtaining the overall solution by solving smaller problems on these subdomains. In this sense the idea is rather old and can be traced to Schwarz's alternating procedure, in which existence of solutions to boundary value problems are proved by an iteration involving solutions on overlapping subdomains. This idea is also widely used in many fields of scientific computing. In structural mechanics, these techniques are known as substructuring or frontal methods and are especially useful when the size of the complete problem is too large for the main memory of the computing machine. In computational fluid dynamics, it is common to decompose the physical domain into different regions and use slightly different forms of the governing equations in each (e.g. the boundary layer equations near a body and potential flow in the far field.)

In the past several years there has been an explosion of activities this research area. The primarily reason is no doubt the advent of parallel computing and the obvious opportunity for parallelism in these methods. Another development has been in the improvement in the efficiency of these methods, primarily through improved handling of the coupling between the subdomain solutions. For example, while Schwarz's procedure is known to converge slowly, acceleration of this method can lead to computationally efficient algorithms [15]. However, we shall not address this class of methods in this paper.

Instead we shall restrict our attention to the class of domain decomposition techniques which use non-overlapping subdomains. The basic idea is to reduce the differential operator on the whole domain to an operator (not necessarily a differential one) on the interfaces between the subdomains. The equations for the interfaces are then solved by iterative methods, such as preconditioned conjugate gradient methods. Typically, each iteration involves the solution of a problem on each of the subdomains and therefore for efficiency reasons, it is essential to keep the number of iterations small by using a good preconditioner. Several such preconditioners have been proposed in the recent literature [3,5,6,10,13,17]. In most aspects, their derivations are mostly unrelated. Our main purpose in this paper is to give a uniform framework in which efficient preconditioners can be derived and their properties analyzed. Moreover, within this framework most of the preconditioners in the literature can be related, compared and generalised.

We use two approaches. Our main approach is based on spectral analysis and can be used to invert exactly the interface operator for general piecewise constant coefficient elliptic operators on rectangular regions in any dimensions. For these operators, our technique leads to domain decomposed fast direct solvers. For more general operators on irregular domains where the exact inverses cannot be derived explicitly, these techniques can easily be adapted to construct efficient preconditioners for the interface operator. Our second approach is based on approximating the interface operator by treating it as a localised pseudo-differential operator. Since this approach does not depend on the special form of the differential operator, it is applicable to more general operators than the spectral approach.

The outline of the paper is as follows. In section 2, we introduce our formulation of the interface system. In section 3, we consider the spectral approach. In particular, for the case of a rectangle decomposed into two smaller ones, we give the exact inverse of the interface operator for a variety of second order elliptic operators and discretisations in 2D and 3D: the Laplace operator (5 point and 9 point discretization), the Helmholtz operator, operators with first order terms with central and upwind differencing, and operators with piecewise constant coefficients in each subdomain. Moreover, we extend these to the multiple subdomain case. In section 5, we consider the use of these exact inverses as preconditioners in the case of irregular domains. The exact inverses allow a comparison of the various preconditioners in the literature to be made. Both numerical and theoretical results are presented in section 6 considering a variety of examples.

2. Formulation

We will first focus with one interface.

with boundary conditions where $L$ is a linear operator and $f$ is a given function.

If we order the interface $\Gamma$ last, the system can be expressed as

The system (2.3) can be written as

Step 1: Compute

The system (2.3) can be written as

Step 1: Compute

The system (2.3) can be written as

Step 1: Compute

The system (2.3) can be written as

Step 1: Compute
Figure 1: The domain $\Omega$ and its partition.

Theoretical results will be given. Section 5 considers extensions to non-separable problems. Finally, section 6 considers briefly the operator approach.

2. Formulation

We will first formulate our approach in the simplest case of a domain split into two subdomains with one interface. Consider the problem:

$$Lu = f \quad \text{on} \quad \Omega$$

with boundary condition

$$u = u_\delta \quad \text{on} \quad \partial \Omega$$

where $L$ is a linear elliptic operator and the domain $\Omega$ is as illustrated in Fig. 1. We will call the interface between $\Omega_1$ and $\Omega_2$, $\Gamma$.

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

$$Au = b,$$  

which can be expressed in block form as:

$$
\begin{pmatrix}
A_{11} & A_{12} & A_{13} \\
A_{21} & A_{22} & A_{23} \\
A_{31}^T & A_{32}^T & A_{33}
\end{pmatrix}
\begin{pmatrix}
u_1 \\
u_2 \\
u_3
\end{pmatrix}
= 
\begin{pmatrix}
b_1 \\
b_2 \\
b_3
\end{pmatrix}.
$$

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

**Step 1:** Compute

$$C = A_{33} - A_{32}^T A_{11}^{-1} A_{12} - A_{32}^T A_{22}^{-1} A_{23},$$  

$$w_1 = A_{11}^{-1} b_1.$$
\[ w_2 = A_{22}^{-1} b_2 \]  
\[ (2.6) \]

and solve
\[ C u_3 = b_3 - A_{13}^T w_1 - A_{23}^T w_2. \]  
\[ (2.7) \]

**Step 2:** Compute
\[ u_1 = w_1 - A_{11}^{-1} A_{13} u_3 \]  
\[ (2.8) \]

and
\[ u_2 = w_2 - A_{22}^{-1} A_{23} u_3. \]  
\[ (2.9) \]

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. The matrix \( C \) (2.4) is the Schur complement of \( A_{12} \) in \( A \) and it is sometimes called the *capacitance matrix* in this context. It corresponds to the reduction of the operator \( L \) on \( \Omega \) to an operator on the boundary \( \Gamma \).

### 3. The Spectral Approach for Separable Operators

The basic idea of the spectral approach is to diagonalize the matrix \( C \) by appropriately chosen eigenvectors [6]. Because of the form of \( C \) in (2.4), it is clear that a vector \( w \) would be an eigenvector of \( C \) if it is an eigenvector for each of the three terms in (2.4). It can be verified that the product of the last two terms with \( w \) corresponds imposing local averages of \( w \) (namely \( A_{13} w \) and \( A_{23} w \)) as Dirichlet boundary conditions on \( \Gamma \), solving for the solutions [say \( v_1 \) and \( v_2 \)] on each subdomain and evaluating local averages of these solutions near \( \Gamma \) (namely \( A_{13}^T v_1 \) and \( A_{23}^T v_2 \)). Thus the issue of finding the eigenvectors of \( C \) is closely related to the separability of the operators \( A_{11}^{-1} \) and \( A_{22}^{-1} \) along the direction of \( \Gamma \). In particular, if \( \Omega \) is rectangular with \( \Gamma \) parallel to one of its edges and the operator \( L \) is also separable in the directions of the two edges of \( \Omega \), then the eigenvectors of \( C \) can often be easily found in terms of the separating eigenfunctions of \( L \). For constant coefficient operators such as the Laplacian with Dirichlet boundary conditions, the eigenvectors are simply the discrete Fourier functions defined on \( \Gamma \). For more complicated operators, such as ones with first order terms, these eigenfunctions are slightly more complicated. For variable coefficient operators, these eigenfunctions may have to be computed numerically. Generally, this technique works for separable operators on rectangular domains, similar to the situation for conventional fast elliptic solvers. Analogously, the spectral approach leads to domain decomposed fast elliptic solvers.

#### 3.1. Laplace Operator

We first consider the case where \( L \) is the Laplacian operator, discretized by the standard second order centered differencing, and \( \Omega \) is a rectangle divided into two or more strips like is shown in Fig. 2. Using the spectral technique explained earlier, the exact eigenvectors and eigenvalues of \( C \) can be derived [2, 6, 7]. The eigenvectors are discrete sine functions.

For the case of two strips, \( C \) has the following eigenvalue decomposition: [6]

\[ W \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{pmatrix} W^T, \]  
\[ (3.1) \]

where \( W \) is the matrix whose columns are

\[ w_j = \sqrt{\frac{2}{n+1}} (\sin j \pi h, \sin 2 j \pi h, \ldots, \sin n j \pi h)^T \]  
\[ (3.2) \]
and

\[
\lambda_j = - \left( \frac{1 + \gamma_j^{m_1+1}}{1 - \gamma_j^{m_1+1}} + \frac{1 + \gamma_j^{m_2+1}}{1 - \gamma_j^{m_2+1}} \right) \sqrt{\sigma_j + \frac{\sigma_j^2}{4}}
\]  \hspace{1cm} (3.3)

\[
\sigma_j = 4 \sin^2 \left( \frac{j \pi h}{2} \right)
\]  \hspace{1cm} (3.4)

\[
\gamma_j = \left( 1 + \frac{\sigma_j}{2} + \sqrt{\sigma_j + \frac{\sigma_j^2}{4}} \right)^2
\]  \hspace{1cm} (3.5)

for \( j = 1, \ldots, n \), where \( h \) is the grid size, and \( m_1 \) and \( m_2 \) are the number of rows of grid points in the \( y \)-direction in \( \Omega_1 \) and \( \Omega_2 \) 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 \( \Gamma \).

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

\[
C = \begin{pmatrix}
C_1 & B_2 & & \\
B_2 & C_2 & & \\
& & \ddots & B_k \\
& & B_k & C_h
\end{pmatrix}
\]  \hspace{1cm} (3.6)

The \( C_i \)'s correspond to the reduce operator on \( \Gamma_i \) and the \( B_i \)'s correspond to the coupling between the interfaces. All blocks \( C_i \) and \( B_i \) have the same matrix of eigenvectors \( W \), i.e. for \( i = 1, \ldots, k \), we have

\[
W^T C_i W = \Lambda_i = \text{diag}(\lambda_{i1}, \ldots, \lambda_{in})
\]  \hspace{1cm} (3.7)
and for \( i = 2, \ldots, k \), we have
\[
W^T B_i W = \text{diag}(\delta_{i1}, \ldots, \delta_{in})
\]
where
\[
\lambda_{ij} = - \left( \frac{1 + \gamma_{ij}^{m+1}}{1 - \gamma_{ij}^{m+1}} + \frac{1 + \gamma_{ij}^{m+1+1}}{1 - \gamma_{ij}^{m+1+1}} \right) \sqrt{\frac{\sigma_i^2}{4} + \frac{\sigma_j^2}{4} - 1},
\]
and
\[
\delta_{ij} = \sqrt{\gamma_{ij}^m} \left( \frac{1 - \gamma_{ij}^m}{1 - \gamma_{ij}^{m+1}} \right).
\]

By first diagonalizing \( C \) with a block diagonal matrix with \( W \) as the diagonal blocks and then rearranging the equations, \( C \) can be reduced to a set 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, the extra work can be saved if care is taken to save some intermediate results from the first solves. We refer the interested reader to \([8],[16]\) where the parallel implementations of these algorithms are also discussed.

3.2. More General Operators and Discretizations

The spectral technique can be extended to more general operators and discretisations and in higher dimensions. We give a few examples here.

The capacitance matrix for the second order centered finite difference discretisation of the operator
\[
u_{xx} + \beta u_{yy},
\]
where the coefficient \( \beta \) takes constant values \( \beta \) on each subdomain \( \Omega_i \), has the same form as \( (3.8) \) except that the eigenvalues of \( C_i \) and \( B_i \) are given by \([17]\)
\[
\lambda_{ij} = - \left( \frac{1 + \gamma_{ij}^{m+1}}{1 - \gamma_{ij}^{m+1}} \right) \sqrt{\frac{\sigma_i^2}{4} + \beta_i \sigma_j} - \left( \frac{1 + \gamma_{ij}^{m+1+1}}{1 - \gamma_{ij}^{m+1+1}} \right) \sqrt{\frac{\sigma_j^2}{4} + \beta_{i+1} \sigma_j},
\]
and
\[
\delta_{ij} = \beta_i \sqrt{\gamma_{ij}^m} \left( \frac{1 - \gamma_{ij}^m}{1 - \gamma_{ij}^{m+1}} \right),
\]
where
\[
\gamma_{ij} = \frac{1}{\beta_i^2} \left( \frac{\sigma_j}{2} + \beta_i - \sqrt{\frac{\sigma_j^2}{4} + \beta_i \sigma_j} \right)^2.
\]

The capacitance matrix for the second order centered finite difference discretisation of the Helmholtz operator
\[
\Delta u + \alpha u
\]
also has the form \((3.6)\), with the eigenvalues of \( C_i \) and \( B_i \) given by \([17]\):
\[
\lambda_{ij} = - \left( \frac{1 + \gamma_{ij}^{m+1}}{1 - \gamma_{ij}^{m+1}} + \frac{1 + \gamma_{ij}^{m+1+1}}{1 - \gamma_{ij}^{m+1+1}} \right) \sqrt{\frac{\mu_j^2}{4} - 1},
\]
and
\[
\delta_{ij} = \beta_i \sqrt{\gamma_{ij}^m} \left( \frac{1 - \gamma_{ij}^m}{1 - \gamma_{ij}^{m+1}} \right),
\]
where \( \alpha \) denotes the frequency of the given problem.

3.3. Non-Self-Adjoint Operators

Our framework allows adding terms, such as \( \kappa \), to the 5-point stencil \( C_0 \) to make it a non-self-adjoint operator, which is very complicated. General results do not exist for the operator \( C \) because the determination of the eigenvalues \( \lambda_j \)'s cannot be used to solve the problem. For the operator \( \Delta u + \alpha u \), analytical results are found analytically, but the results depend on \( \alpha \). A preconditioner \( \nu \) will be presented here that works for the kinds of problems we are talking about.

What we will talk about today needs a preconditioner and
and

\[ \delta_{ij} = \sqrt{\tau_{ij}^2 \left( \frac{1 - \gamma_j}{1 - \gamma_{j+1}} \right)^2}, \]  

(3.16)

where

\[ \mu = -\sigma_j - 2 + a h^2 \]  

(3.17)

and

\[ \gamma_j = \left( -\frac{\mu_j^2}{2} - \sqrt{\frac{\mu_j^2}{4} - 1} \right)^2. \]  

(3.18)

The capacitance matrix for the 3D Laplacian has the same form as the 2D version with \( \sigma_j \) replaced by the eigenvalues of the 2D Laplace operator [16].

Different discretizations can also be treated within this framework. For example, consider the following nine point fourth order discretization of the Laplacian [21]:

\[ \Delta_9 = 1/6h^2 \text{tridiagonal}(S_i, T_i, S_i) \]

where \( S_i = \text{tridiagonal}(1, 4, 1) \) and \( T_i = \text{tridiagonal}(4, -20, 4) \). The capacitance matrix \( C_9 \) associated with \( \Delta_9 \) has the same form as (3.1), with \( \sigma_j \) replaced by

\[ \sigma_j = \frac{4 \sin^2 \left( \frac{i x h}{2} \right)}{1 - \frac{3}{2} \sin^2 \left( \frac{i x h}{2} \right)}. \]  

(3.19)

It can also be easily shown that

\[ \lim_{h \to 0} \mathcal{K}(C_9^{-1}C_9) = \sqrt{\frac{3}{2}}, \]

where \( \mathcal{K} \) denotes the spectral condition number and \( C_9 \) denotes the capacitance matrix corresponding to the 5-point discrete Laplacian. The above result shows that \( C_9 \) is spectrally equivalent to \( C_0 \) and is a reasonably good preconditioner for it.

3.3. Non-Self-Adjoint Operators With First Order Terms

Our framework can also be extended to second order elliptic problems with first order derivative terms, such as the operators \( L_x \equiv \Delta + \alpha u_y \) and \( L_y \equiv \Delta + \alpha u_x \). Since the discretizations of these non-self-adjoint operators lead to nonsymmetric matrices, the spectral approach becomes more complicated. Consider a rectangle split by an interface \( \Gamma \) along the \( x \)-direction, as in Figure 2. For the operator \( L_y \), it is easy to see that the Fourier matrix \( W \) can still be used to diagonalize \( C \) because the \( y \)-derivative does not affect the separating eigen-modes in the \( x \)-direction. The eigenvalues \( \lambda_j \)'s of course depends on the value of \( \alpha \). On the other hand, for the operator \( L_x \), \( W \) cannot be used to diagonalized \( C \) because the Fourier modes are no longer eigenfunctions of the operator \( L_x \) in the \( x \)-direction. It turns out, however, that the eigenfunctions of \( C \) can still be found analytically - they are simply given by \( DW \), where \( D \) is a suitably chosen diagonal matrix that depends on \( \alpha \). The eigenvalues depend on \( \alpha \) as well. These formulas are too complicated to be presented here and the interested reader is referred to a recent report by Chan and Hou [19], where results for both centered and upwind discretizations for the first order terms are presented.

What we would like to show here is the effectiveness of these exact preconditioners when applied to problems where the first derivative terms are not negligible. Consider the situation where one needs a preconditioner \( M \) for the boundary operator \( C(\alpha) \) corresponding to \( L_y \). Without the
knowledge of the exact preconditioners, a natural approach is to use $M = C(0)$, for which we know
the exact diagonalization. Moreover, spectral equivalence results [10] guarantee that for fixed $\alpha$
the condition number of the preconditioned system remains bounded independent of $h$. It turns
out, however, that for fixed $h$ and large values of $\alpha$ the preconditioned system can have a large
condition number. For example, for the case where $h = 0.02, m_1 = 50, m_2 = 100$, and upwind
differencing is used, the values of $K(C^{-1}(0)C(\alpha))$ are approximately 15 and 40 for values of the
"cell-Reynolds-number" $\alpha h/2$ equal to 0.4 and 0.8 respectively.

4. Irregular Domains and Preconditioners

For general irregular domains, the eigenvalues and eigenvectors of the capacitance matrix
cannot be computed analytically via the spectral techniques, and hence one must find alternative
methods for solving the capacitance system (2.7).

Note that the computation of the capacitance matrix $C$ 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 in general.

Instead of solving the system (2.7) directly, iterative methods such as preconditioned conjugate
gradient methods (PCG) can be applied, in which only matrix vector products $Cy$ for arbitrary
$y \in \mathbb{R}^m$ are required. As explained earlier, this product can be computed by one solve on each
subdomain with boundary condition on $\Gamma$ determined 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 $C$. In
this section, we shall survey some preconditioners in the literature, highlighting the relationships
among them. We shall also analyze their performance, with special emphasis on the dependence
on the mesh size $h$ and the departure from regularity of the domain.

4.1. Survey of Preconditioners

We summarize several of the preconditioners which have been proposed in the literature
[3,5,6,10,13]. We summarize them here in our notations to make it easier to compare them.

1. In [10], Dryja proposed the following preconditioner for (2.4):

$$M_D = W \text{diag}(\lambda_D^1, \lambda_D^2, \ldots, \lambda_D^m) W^T,$$

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

$$\lambda_D^j = -2\sqrt{\sigma_j}$$

with $\sigma_j$ given by (3.4). This preconditioner is based on the Sobolev trace theorem [22]. He
proved that $K(M_D^{-1}C)$ is bounded independently of the mesh size $h$.

2. Golub and Mayers [13] proposed the preconditioner:

$$M_G = W \text{diag}(\lambda_G^1, \lambda_G^2, \ldots, \lambda_G^m) W^T,$$

where

$$\lambda_G^j = -2\sqrt{\sigma_j + \frac{\sigma_j^2}{4}}$$

The derivation is motivated by considering the generating function for the solution for the case
where the boundaries of the two domains move away to infinity. Empirical results in [13] show
that $M_G$ performs better than $M_D$.

3. Another interesting preconditioner of Dryja [10] is

$$M_D = W \text{diag}(\lambda_D^1, \lambda_D^2, \ldots, \lambda_D^m) W^T,$$

where

$$\lambda_D^j = -2\sqrt{\sigma_j}$$

The underlying idea is that the diagonal of the matrix $M_D$ can be factorized by $W$ and it is easy to see that the product $M_D^{-1}$ is a block diagonal product of the subdomain preconditioners $L_B^{-1}$.

4. Finally, Chan, et al. [8] proposed a matrix that is close to the regular case, arising from the
FFT computation. It can be proved that $A_f$ is block diagonal.

4.2. Performance

In Fig. 3 we consider a T-shaped region. We consider a uniform number of interior points and
the number of the points on the boundary. From the performance of the

$$\lambda_D^j = -2\sqrt{\sigma_j + \frac{\sigma_j^2}{4}}$$

where

$$\lambda_G^j = -2\sqrt{\sigma_j}$$

The derivation is motivated by considering the generating function for the solution for the case
where the boundaries of the two domains move away to infinity. Empirical results in [13] show
that $M_G$ performs better than $M_D$.

3. Another interesting preconditioner of Dryja [10] is

$$M_D = W \text{diag}(\lambda_D^1, \lambda_D^2, \ldots, \lambda_D^m) W^T,$$

where

$$\lambda_D^j = -2\sqrt{\sigma_j}$$

The underlying idea is that the diagonal of the matrix $M_D$ can be factorized by $W$ and it is easy to see that the product $M_D^{-1}$ is a block diagonal product of the subdomain preconditioners $L_B^{-1}$.

4. Finally, Chan, et al. [8] proposed a matrix that is close to the regular case, arising from the
FFT computation. It can be proved that $A_f$ is block diagonal.

4.2. Performance

In Fig. 3 we consider a T-shaped region. We consider a uniform number of interior points and
the number of the points on the boundary. From the performance of the
3. Another interesting preconditioner was given by Björstad and Widlund [3] (based on a suggestion of Dryja’s) and has the following form:

\[ M_B = A_{33} - 2A_{13}^T A_{11}^{-1} A_{13} \]

It is easy to show that the eigenvalue decomposition of \( M_B \) is

\[ M_B = W \text{diag}(\lambda_1^B, \lambda_2^B, \ldots, \lambda_n^B) W^T, \quad (4.5) \]

where

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

The underlying motivation for this preconditioner is exploiting symmetries in the operator and the domain about the interface. When \( \Omega_1 \) and \( \Omega_2 \) are identical (and hence \( A_{11} = A_{22} \)), it is easy to see that \( M_B \) is an exact preconditioner. Björstad and Widlund showed that the product \( M_B^{-1} Cu \) can be computed by solving a mixed Neumann-Dirichlet problem in one of the subdomains and a Dirichlet problem in the other one. The basic idea is that if both the operator \( L \) and the domain are symmetric about \( \Gamma \) then the solution can be found by solving on only one of the subdomains with a Neumann boundary condition on \( \Gamma \). They also proved that \( K(CM_B^{-1}) \) is uniformly bounded for certain finite element approximation of Dirichlet problems for self-adjoint second order elliptic problems in plane regions. Their method has the advantage that it can be applied to more general operators and domain shapes. However, 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) suggests. They also proved that \( M_B \) is spectrally equivalent to \( O \) [3].

4. Finally, Chan [6] suggested a procedure for extending the exact preconditioner (3.1) for rectangular regions to construct preconditioners for irregular regions. The idea is to use as preconditioner the exact capacitance matrix corresponding to a best rectangular approximation to the irregular domain sharing the same interface. The motivation is to improve Dryja’s and Golub/Mayer’s preconditioners by taking into account the aspect ratio of the subdomains. We will call this preconditioner \( M_G \).

Although \( M_D, M_G, M_B \) and \( M_G \) were derived independently, we have expressed them in the same matrix factorization formats. Since the eigenvectors are the same, to compare them we only need to look at their eigenvalues \( \lambda_j \)'s. On a rectangle, for which \( M_G \) is exact, \( M_D, M_G \) and \( M_B \) can be viewed as progressively better approximations to \( M_G \). The \( \lambda_j^G \)’s are exact for rectangles with infinitely large aspect ratios because the coefficient in front of \( \lambda_j \) in (3.9) tends to \(-2\) in the limit of \( m_1 \) and \( m_2 \) tending to \( \infty \). It can also be easily observed that \( \lambda_j^D \) is a first order approximation to \( \lambda_j^G \) for the small \( \lambda_j \)'s but underestimates the larger \( \lambda_j \)'s. Finally, it is easy to see that the \( \lambda_j^B \)’s are exact only for the case when \( m_1 = m_2 \). For more detailed analysis, the reader is referred to [6].

4.2. Performance of Preconditioners: Numerical Results

In Fig. 3 we compare the preconditioners \( M_D, M_G \) and \( M_G \) for the Poisson equation on a T-shaped region \( \Omega \) as given in Fig. 1, where we vary the aspect ratio of the subdomain \( \Omega_1 \). We consider a uniform grid on \( \Omega \) with \( n = 15 \) grid points on the interface \( \Gamma \). By varying \( m_1 \), the number of interior grid points in the \( y \) direction on the subdomain \( \Omega_1 \), we computed the condition number of the preconditioned capacitance system for different aspect ratios defined as \( \frac{m_1+1}{m_1} \). As we can see from the plots, \( M_G \) performs very well, even when \( \Omega_1 \) becomes very narrow, while the performance of the others deteriorate as the aspect ratio becomes small. The curves for \( M_G, M_G \)
and $M_B$ are indistinguishable for aspect ratios larger than one and they are all better than $M_D$. See [14] for a careful numerical comparison of these and other preconditioners for constant and variable coefficients operators.

4.3. Dependence on Irregularity of Domain: Some Theoretical Results

Besides the empirical evidence of the performance of the various preconditioners, there are also some theoretical results available. The most common results of this kind are spectral equivalence results which asserts that a particular preconditioner is spectrally equivalent to the exact boundary operator as the grid size $h$ tends to zero for a fixed domain $[3,10]$. This essentially guarantees that the number of iterations needed to solve the preconditioned system to a given accuracy is independent of $h$. For domains partitioned with interfaces and cross-points, these spectral results must be relaxed to allow for a slight increase (of the form $\log(h^{-1})$ in the conditioning of the preconditioned operator $[3,5]$.

In a somewhat orthogonal direction, we have recently obtained some theoretical results concerning the performance of preconditioners as the shape of the domain varies. This issue is of obvious practical importance in applying the preconditioners to domains of varying shapes. We prove that $[18]$ on any L-shaped domain, the preconditioned capacitance matrix for the preconditioner $M_C$ is bounded by 2.16, independent of $h$ and the aspect ratios of the subdomains. Moreover, the convergence rate is essentially the same irrespective of how the domain is partitioned (there are two ways of partitioning an L-shape domain into two rectangular domains). Similar results are also obtained for C-shaped regions. This independence of the aspect ratios is a special property of the preconditioner $M_C$ not shared by the other preconditioners in general (see Fig. 3 for example), and can be traced directly to the fact that the aspect ratios of the subdomains are incorporated into the exact preconditioner for the approximating rectangle from which $M_C$ is derived.

![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) preconditioners.](image)

5. Non-Separable Systems

If the operator $A$ is non-separable, there is no vector product interface system.

An alternative is to use a preconditioner $M$ that is diagonal. The former could be obtained from the matrix $A$ by

$$M = C A C$$

where $C$ is the block-diagonal matrix

$$C = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

where $M$ is a block-diagonal matrix.

6. The Operator $M_D$

So far, our discussion has been centered around the preconditioner $M_D$. As a steady state Neumann problem for an elliptic equation $M_D u = f$, the operator $M_D$ is intimately connected with the solution of the original differential operator. An interesting example is the heat equation. Here we shall use this equation to gain insight into the behavior of the preconditioned operator, the theoretical results for more general problems being of course much more difficult.

Here we shall consider the heat equation with a $k$-diagonal preconditioner. The preconditioned operator $M_D$ is defined as

$$M_D = D^{-1} A$$

where $D$ is a diagonal matrix.
5. Non-Separable Problems

If the operator $A$ is non-separable, there usually are no fast solvers available for $A_{11}$ and $A_{22}$. Therefore, in each iteration of an iterative method for solving the capacitance system, the matrix-vector product $Cy$ cannot be evaluated inexpensively, making methods which work only with the interface system ineffective.

An alternative is to solve the system (2.3) on the whole domain instead of just the capacitance system on the interface. We will show that preconditioners for (2.3) can be derived from preconditioners for the capacitance matrix. Let $B_{11}$ and $B_{22}$ be approximations to $A_{11}$ and $A_{22}$. The former could be separable approximations to the latter or they could represent some truncated inner iteration for solving systems with the latter [20]. Based on the following decomposition of the matrix $A$ in (2.3):

$$
A = egin{pmatrix}
A_{11} & A_{12} \\
A_{21} & A_{22}
\end{pmatrix}
\begin{pmatrix}
I & A_{11}^{-1}A_{13} \\
I & A_{22}^{-1}A_{23}
\end{pmatrix}
\begin{pmatrix}
I & 0 \\
0 & I
\end{pmatrix},
$$

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

$$
\tilde{M} = egin{pmatrix}
B_{11} & B_{12} \\
B_{21} & B_{22}
\end{pmatrix}
\begin{pmatrix}
I & B_{11}^{-1}A_{13} \\
I & B_{22}^{-1}A_{23}
\end{pmatrix}
\begin{pmatrix}
I & 0 \\
0 & I
\end{pmatrix},
$$

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 [4,5,8a]. They used both $M_D$ and $M_B$ as the preconditioner $M$ for $C$. 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 $w_0^0$ is equivalent to the PCG algorithm applied to (2.3) with preconditioner given by (5.2) and initial guess $(A_{11}^{-1}(b_1 - A_{13}w_0^0), A_{22}^{-1}(b_2 - A_{23}w_0^0), w_0^0)$. In [14], numerical experiments were performed with these and other preconditioners.

6. The Operator Approach

So far, our approach for deriving preconditioners for $C$ depends on special differential properties of the operator $A$ (except for $M_B$ which only uses symmetry arguments). This raises the question of how effective they will be when applied to other more different and complicated operators, (e.g. the steady state Navier-Stokes operator), without first somehow reducing the problem to one of a second order elliptic problem that we have already treated here. For example, Dryja's preconditioner $M_D$ is intimately tied to the Sobolov Trace Theorem for second order elliptic problems and it cannot be expected to perform well for other types of operators. It is therefore desirable to derive preconditioners in a more general way that depends less critically on the particular form of the differential operator, but more on the other computable quantities of the given operator. An example is the exploitation of symmetry in the style of $M_B$, which can be expected to be applicable for more general class of problems.

Here we investigate another approach which depends on efficiently "probing" the operator $C$ to gain information on its structure. This information can then be used to construct an effective preconditioner. Our main motivation is the empirical observation that, in the case of the Laplace operator, the elements of the matrix $C$ decay rapidly away from the main diagonal [18]. 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 \( C u_i, i = 1, 2, 3 \), where \( u_2 = (1,0,0,1,0,0,\ldots)^T \), \( u_3 = (0,1,0,0,1,0,\ldots)^T \) and \( u_3 = (0,0,1,0,0,0,\ldots)^T \). 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 \( C u_i, i = 1, 2, 3 \). Note that the computation of each product \( C u_i \) involves solving one problem on each subdomain with \( u_i \) as boundary condition on the interface.

Note that in principle this approach can be applied to any operator \( A \) and requires only a solver for the subdomains. However, it can only be expected to be effective for those operators for which the reduced boundary operator \( C \) is predominantly local (corresponding to the rapidly decaying elements away from the diagonal.)

In Figure 4, we plot the eigenvalues of the tridiagonal preconditioner computed by the above method (denoted by \( M_3 \)) together with the eigenvalues of \( M_D \), \( M_G \), and \( C \) for the problem of a Laplacian on a square divided into two strips, with \( n = 15 \) and \( m_1 = m_2 = 7 \). For this problem, the plots for \( C \) and \( M_G \) are indistinguishable. The preconditioner \( M_D \) underestimates the large eigenvalues of \( C \) whereas \( M_3 \) seems to follow the exact eigenvalues more closely.

The generalisation 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 \( C e \), 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.

In general, this approach is well suited to problems involving block Toeplitz matrices, such as those that arise in physical problems where a Newton-Raphson iteration is used many times and the coefficients of \( C \) are modified at each iteration.

---

**Figure 4:** Comparison of Eigenvalues of Preconditioners for Poisson's Equation with \( n = 15, m_1 = m_2 = 7 \).
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 suggested 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.

References


