Electronic Transactions on Numerical Analysis.
Volume 2, pp. 76-91, June 1994.

ISSN 1068-9613.

ETNA
Kent State University
etna@mcs.kent.edu

EFFICIENT ITERATIVE SOLUTION OF LINEAR SYSTEMS FROM DISCRETIZING SINGULAR INTEGRAL EQUATIONS*

KE CHEN †

Abstract. In this paper we study the solution of singular integral equations by iterative methods. We show that discretization of singular integral operators obtained by domain splitting yields a system of algebraic equations that has a structure suitable for iterative solution. Numerical examples of Cauchy type singular integral equations are used to illustrate the proposed approach. This paper establishes a theory for experimental results presented previously.

Key words. singular integral equations, non-compact operators, direct solutions, preconditioning, conjugate gradient iterative methods.

AMS subject classifications. 65F10, 65N38, 45E05.

1. Introduction.

Numerical solution of integral equations is a much studied subject. However, most successes with iterative solutions have been associated with some non-general assumptions of the underlying operator. Consider the integral equation of Fredholm type in the standard form

$$\lambda u(s) - \int_a^b k(s,t)u(t)dt = f(t), \qquad a \le s \le b, \tag{1.1}$$

or in the equivalent operator form (in $C[a,b]$)

$$(\lambda - \mathcal{K})u = f. \tag{1.2}$$

Here equation (1.2) is of the second kind when $\lambda \neq 0$ and of the first kind when $\lambda = 0$. When solving (1.2) numerically, we either seek to determine an approximate solution in a chosen finite dimensional space $\mathcal{F}_n$ by a projection method

$$(\lambda - \mathcal{P}_n\mathcal{K})u_n = \mathcal{P}_n f, \tag{1.3}$$

where $u_n \in \mathcal{F}_n$ and $\mathcal{P}_n : C \to \mathcal{F}_n$ is a projection operator, or we use the Nyström quadrature method

$$(\lambda\mathcal{I} - \mathcal{K}_n)u_n = f, \tag{1.4}$$

where $\mathcal{K}_n$ approximates $\mathcal{K}$ and is obtained by discretization of $\mathcal{K}$ by an n-point quadrature rule; see [4, 8, 13, 28].

Such discretizations of integral equations give rise to dense linear systems of equations. As is known, these systems can be prohibitively expensive to solve as n, the order of the linear system of algebraic equations, increases. Iterative methods are the natural options for efficient solutions. Efficient iterative solvers such as the conjugate gradient and multigrid methods have only been shown to be applicable to the case where the underlying integral operator is compact. In the case of $\mathcal{K}$ in (1.2) being compact, successful applications of iterative methods have been reported; see [5, 2, 21, 25, 30, 32].

* Received January 12, 1994. Accepted for publication June 15, 1994. Communicated by L. Reichel.

† Department of Statistics and Computational Mathematics, Victoria Building, The University of Liverpool, Brownlow Hill, Liverpool L69 3BX, England. *(Email : chen@scmp.scm.liv.ac.uk).*

ETNA
Kent State University
etna@mcs.kent.edu

When $\mathcal{K}$ is not a compact operator, only few results are available on the convergence of iterative methods. In the particular case when $\mathcal{K}$ is not compact because the boundary in a boundary integral equation is not smooth, some results on multigrid methods can be found in [6, 11, 34] and on the conjugate gradient method in [10]. In the notation of (1.1), this kind of non-compactness may be characterized by a fixed point singularity, say, as $t \to a$. The concern of the present paper is the large class of problems where the 'moving' singularity as $t \to s$ causes the operator to be non-compact. These problems include singular integral equations (SIE's) of Cauchy type and of Hadamard finite part type.

We remark that there are regularization methods for the solution of Cauchy type singular integral equations (CSIE's), which effectively transform the original integral equation, such as (1.1), into a new integral equation with compact operators (weakly-singular kernels) involving double integrals; see [17, 22]. Iterative methods for the so obtained integral equations are discussed in [18]. A related approach can be used to solve hypersingular integral equations (HSIE's); see [9, 1]. For the regularized equations obtained, several iterative methods are known to be readily applicable, although this regularization approach is expensive.

It has been widely noted that the direct approach to solving SIE's is more efficient than first regularizing the equation and then solving the regularized equation, because the direct approach requires less numerical integration. Moreover, the solutions obtained from the direct and the regularized approach can be identical in the case of CSIE's; see [22]. Direct solution methods for SIE's are discussed in [16, 26, 27, 1] and the references therein. However, there has been little work reported on direct iterative methods that we are aware of. In [3], we attempted to devise a modification of a two-grid method that obtained good results for the case of HSIE's. But that approach was heuristic.

The present paper considers the direct solution of a non-compact integral operator equation by iterative methods. The singularity occurs at $t \to s$ for all $s \in (a, b)$ in equation (1.1). The results of our work are applicable to the direct solution of both CSIE's and HSIE's, and they also explain why the method in [3] works well.

Our main idea in this paper is based on introducing and identifying suitable splittings of singular integral operators into the most singular (but bounded) part and the compact part. We propose to use the inverse of the bounded operator as a preconditioner for the equation. Numerical discretizations will reveal that the preconditioned equation can be solved efficiently and that iterative methods are applicable. We suggest the use of a simple conjugate gradient method, but other methods such as the multigrid methods may be applied. Examples of solving CSIE's are given to show the efficiency of the proposed method.

2. Operator splitting techniques. We consider a class of singular integral operators of the form

$$(\mathcal{K}u)(s) = \int_a^b k(s,t)u(t)dt, \qquad s \in [a, b], \tag{2.1}$$

whose kernel $k(s, t)$ is singular when $t \to s$, and such that $\mathcal{K}$ is non-compact, but for any positive $\hat{\varepsilon}_1,\ \hat{\varepsilon}_2 > 0$ the operator

$$(\hat{\mathcal{K}}u)(s) = \left(\int_a^{s-\hat{\varepsilon}_1} + \int_{s+\hat{\varepsilon}_2}^b \right) k(s,t)u(t)dt, \qquad s \in [a, b], \tag{2.2}$$

ETNA
Kent State University
etna@mcs.kent.edu

is compact in $C[a,b]$. Assume that $\mathcal{K}$ is bounded and that λ is not an eigenvalue of $\mathcal{K}$. Then equation (1.1) has a unique solution. This kind of kernels include as a special case

$$k(s,t) = \frac{\hat{k}(s,t)}{|s-t|^\alpha}, \qquad \alpha \geq 1, \tag{2.3}$$

where $\alpha = 1$ corresponds to a Cauchy singular operator, and where $\alpha > 1$ corresponds to a hypersingular operator valid in the sense of Hadamard finite part; see [20, 15]. Here we have assumed that $\hat{k}(s,t)$ is continuous for $s,t \in [a,b]$.

Partition $[a,b]$ into m subintervals $I_i = [s_{i-1}, s_i),\ i = 1, \cdots, m$, i.e.,

$$P^m \ : \ a = s_0 < s_1 < \cdots < s_{m-1} < s_m = b. \tag{2.4}$$

We propose to split $\mathcal{K}$ into $\mathcal{K}_1 + \mathcal{K}_2$ based on this partition, where $\mathcal{K}_1$ is bounded and $\mathcal{K}_2$ is compact, and consider two approaches.

2.1. Method A. Define for $i,j = 1, \cdots, m$ operators

$$(\mathcal{K}_{i,j}u)(s) = \int_{s_{j-1}}^{s_j} k(s,t)u(t)dt, \qquad \text{for} \quad s \in I_i, \tag{2.5}$$

which are the restrictions of $\mathcal{K}$ over interval I_i. Then evidently we have for $s \in I_i$ that

$$(\mathcal{K}u)(s) = \sum_{j=1}^{m} (\mathcal{K}_{i,j}u)(s). \tag{2.6}$$

If we denote by u_i the restriction of any function $u \in C[a,b]$ to I_i accordingly, then, for any $s \in [a,b]$,

$$\mathcal{K}u = \begin{pmatrix} \mathcal{K}_{1,1} & \mathcal{K}_{1,2} & \cdots & \mathcal{K}_{1,m} \\ \mathcal{K}_{2,1} & \mathcal{K}_{2,2} & \cdots & \mathcal{K}_{2,m} \\ \vdots & \vdots & \cdots & \vdots \\ \mathcal{K}_{m,1} & \mathcal{K}_{m,2} & \cdots & \mathcal{K}_{m,m} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_m \end{pmatrix} \tag{2.7}$$

We propose the splitting

$$\mathcal{K} = \mathcal{D}_1 + \mathcal{C}_1, \tag{2.8}$$

where

$$\mathcal{D}_1 = \begin{pmatrix} \mathcal{K}_{1,1} & \mathcal{K}_{1,2} & & \\ \mathcal{K}_{2,1} & \mathcal{K}_{2,2} & \ddots & \\ & \ddots & \ddots & \mathcal{K}_{m-1,m} \\ & & \mathcal{K}_{m,m-1} & \mathcal{K}_{m,m} \end{pmatrix} \qquad \text{and} \qquad \mathcal{C}_1 = \mathcal{K} - \mathcal{D}_1.$$

It can be seen that, for the partition P^m, operator $\mathcal{C}_1$ is compact as its kernel is continuous, $\mathcal{D}_1$ contains all singularities of $\mathcal{K}$ and is bounded since $\mathcal{K}$ is bounded.

ETNA
Kent State University
etna@mcs.kent.edu

2.2. Method B. In order to further isolate the singularities of $\mathcal{D}_1$ as well as those of $\mathcal{K}$, we now present a second approach. We modify the partition in (2.4) to

$$(2.9)\ P^m_{\underline{\varepsilon}} \ : \quad \begin{array}{r} a = s_0 < s_0 + \varepsilon_0 < s_1 - \varepsilon_1 < s_1 < s_1 + \varepsilon_1 < \cdots \\ \cdots < s_{m-1} < s_{m-1} + \varepsilon_{m-1} < s_m - \varepsilon_m < s_m = b, \end{array}$$

where $\underline{\varepsilon} = \{\varepsilon_0, \varepsilon_1, \cdots, \varepsilon_m\}$ with $\varepsilon_0 = \varepsilon_m = 0$. As before, let $\bar{I}_i = [s_{i-1} + \varepsilon_{i-1}, s_i - \varepsilon_i)$, for $i = 1, \cdots, m$, and $\bar{I}_{\varepsilon_j} = [s_j - \varepsilon_j, s_j + \varepsilon_j)$, for $j = 2, \cdots, m-1$, be subintervals of $[a, b]$. Denote the restrictions of operator $\mathcal{K}$ by $\{\mathcal{K}_{i,j}\}$, $\{\mathcal{K}_{i,\varepsilon_j}\}$, $\{\mathcal{K}_{\varepsilon_i,j}\}$, and $\{\mathcal{K}_{\varepsilon_i,\varepsilon_j}\}$, i.e.,

$$\begin{array}{rcll}
(\mathcal{K}_{i,j}u)(s) & = & \displaystyle\int_{\bar{I}_j} k(s,t)u(t)dt, & for\ s \in \bar{I}_i, \\
(\mathcal{K}_{i,\varepsilon_j}u)(s) & = & \displaystyle\int_{\bar{I}_{\varepsilon_j}} k(s,t)u(t)dt, & for\ s \in \bar{I}_i, \\
(\mathcal{K}_{\varepsilon_i,j}u)(s) & = & \displaystyle\int_{\bar{I}_j} k(s,t)u(t)dt, & for\ s \in \bar{I}_{\varepsilon_i}, \\
(\mathcal{K}_{\varepsilon_i,\varepsilon_j}u)(s) & = & \displaystyle\int_{\bar{I}_{\varepsilon_j}} k(s,t)u(t)dt, & for\ s \in \bar{I}_{\varepsilon_i}.
\end{array}$$

Then operator $\mathcal{K}$ may be written as

$$(2.10)\ \ \mathcal{K} = \begin{pmatrix}
\mathcal{K}_{1,1} & \mathcal{K}_{1,\varepsilon_1} & \mathcal{K}_{1,2} & \mathcal{K}_{1,\varepsilon_2} & \cdots & \mathcal{K}_{1,\varepsilon_{m-1}} & \mathcal{K}_{1,m} \\
\mathcal{K}_{\varepsilon_1,1} & \mathcal{K}_{\varepsilon_1,\varepsilon_1} & \mathcal{K}_{\varepsilon_1,2} & \mathcal{K}_{\varepsilon_1,\varepsilon_2} & \cdots & \mathcal{K}_{\varepsilon_1,\varepsilon_{m-1}} & \mathcal{K}_{\varepsilon_1,m} \\
\mathcal{K}_{2,1} & \mathcal{K}_{2,\varepsilon_1} & \mathcal{K}_{2,2} & \mathcal{K}_{2,\varepsilon_2} & \cdots & \mathcal{K}_{2,\varepsilon_{m-1}} & \mathcal{K}_{2,m} \\
\vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
\mathcal{K}_{m,1} & \mathcal{K}_{m,\varepsilon_1} & \mathcal{K}_{m,2} & \mathcal{K}_{m,\varepsilon_2} & \cdots & \mathcal{K}_{m,\varepsilon_{m-1}} & \mathcal{K}_{m,m}
\end{pmatrix} = \mathcal{D}_2 + \mathcal{C}_2,$$

where

$$\mathcal{D}_2 = \begin{pmatrix}
\mathcal{K}_{1,1} & \mathcal{K}_{1,\varepsilon_1} & & \\
\mathcal{K}_{\varepsilon_1,1} & \mathcal{K}_{\varepsilon_1,\varepsilon_1} & \ddots & \\
 & \ddots & \ddots & \mathcal{K}_{\varepsilon_{m-1},m} \\
 & & \mathcal{K}_{m,\varepsilon_{m-1}} & \mathcal{K}_{m,m}
\end{pmatrix}.$$

Note that the definition for $\mathcal{K}_{i,j}$ in (2.10) is different from that of (2.8) because the partition has been modified.

Remark 2.1. *When the integral in (2.1) is obtained by simplifying a boundary integral over a closed contour, it is possible that both ε_0 and ε_m in Method B must be chosen non-zero and in this case both methods require a slight modification with regard to the correct splitting in order to make $\mathcal{C}_1$ and $\mathcal{C}_2$ compact. This point is discussed further in § 7.*

3. Discretization of singular integral operators. We now consider the discretization of the singular integral operator (2.1) and show how the same discretization for singular operators $\mathcal{D}_1$ and $\mathcal{D}_2$ gives rise to block diagonal matrices. This is true for a class of commonly used numerical methods including the classical collocation and Nyström methods.

3.1. Collocation methods using piecewise polynomials. For simplicity and without essential loss of generality, we describe all subsequent discretization methods by taking the number of divisions of $[a, b]$ to be m and by partitioning $[a, b]$ according to P^m of (2.4). More generally, this idea can be applied to any independent partition of $[a, b]$ different from that of the operator splitting partition (2.4), and similar results may be derived.

Now let us assume that we use the partition P^m of (2.4). Suppose in each interval I_i $(i = 1, \cdots, m)$ an r-th order piecewise interpolating polynomial is sought to approximate the function

$$u_m(s) = \sum_{\ell=1}^{r} u_{i,\ell}\phi_{i,\ell}(s) \tag{3.1}$$

at r interpolation points $\{s_{i,\ell}\}$

$$s_{i-1} \le s_{i,1} < s_{i,2} < \cdots < s_{i,r} \le s_i, \tag{3.2}$$

where $\phi_{i,\ell}$ represents the ℓ-th Lagrange basis polynomial. Then the approximation for $\mathcal{K}$ is (for $s \in I_i$)

$$(\mathcal{K}u_m)(s) = \sum_{i=1}^{m} \int_{s_{i-1}}^{s_i} \sum_{j=1}^{r} u_{i,j}\phi_{i,j}(t)k(s,t)dt = \sum_{i=1}^{m} (\mathcal{K}_{i,j}u_m)(s), \tag{3.3}$$

where $(\mathcal{K}_{i,j}u_m)(s)$ approximates $(\mathcal{K}_{i,j}u)(s)$ for $i, j = 1, \cdots, m$.

Define the residual function, for $s \in [a, b]$,

$$r(s) = \lambda u_m(s) - (\mathcal{K}_{i,j}u_m)(s) - f(s). \tag{3.4}$$

On formally collocating at $\{s_{i,j}\}$, we obtain for Method A of §2.1 the equation

$$\lambda \underline{u}_m - D_1\underline{u}_m - C_1\underline{u}_m = \underline{f}_m, \tag{3.5}$$

where

$$\begin{cases} \underline{u}_m = (u_{1,1}, \cdots, u_{1,r}, u_{2,1}, \cdots, u_{2,r}, \cdots, u_{m,1}, \cdots, u_{m,r})^T \\ \underline{f}_m = (f_{1,1}, \cdots, f_{1,r}, f_{2,1}, \cdots, f_{2,r}, \cdots, f_{m,1}, \cdots, f_{m,r})^T \end{cases}$$

and $u_{i,j} = u_m(s_{i,j})$. Here $K = (K_{i,j})_{mr\times mr}$ is a block matrix of size $m \times m$, with each $K_{i,j}$ block of size $r \times r$, and $C_1 = K - D_1$, where

$$D_1 = \begin{pmatrix} K_{1,1} & K_{1,2} & & \\ K_{2,1} & K_{2,2} & \ddots & \\ & \ddots & \ddots & K_{m-1,m} \\ & & K_{m,m-1} & K_{m,m} \end{pmatrix} \tag{3.6}$$

is a block tridiagonal matrix of size $m \times m$.

3.2. Nyström quadrature methods. Suppose the following quadrature rule is used over interval I_i for $i = 1, \cdots, m$ (note that $\bigcap I_i = \{0\}$ and $\bigcup I_i = [a, b]$)

$$\int_{s_{i-1}}^{s_i} g(t)dt \approx \sum_{j=1}^{r} w_j g(s_{i,j}), \tag{3.7}$$

ETNA
Kent State University
etna@mcs.kent.edu

where $\{w_j\}$ are the weights and $\{s_{i,j}\}$ are the r quadrature nodal points. Then $(\mathcal{K}u)(s)$ is approximated by

$$(\mathcal{K}_m u_m)(s) = \sum_{i=1}^{m}\sum_{j=1}^{r} w_j k(s, s_{i,j}) u_m(s_{i,j}) = \sum_{i=1}^{m} (\mathcal{K}_{i,j}^{(m)} u_m)(s), \tag{3.8}$$

where $(\mathcal{K}_{i,j}^{(m)} u_m)(s)$ approximates $(\mathcal{K}_{i,j})u(s)$ for $i, j = 1, \cdots, m$. Denote the residual function by

$$r(s) = \lambda u_m(s) - (\mathcal{K}_{i,j} u_m)(s) - f(s), \qquad \text{for } s \in [a, b]. \tag{3.9}$$

Then collocating at all nodal points $\{s_{i,j}\}$ using Method A of §2.1 gives rise to the linear system

$$\lambda \underline{u}_m - D_1 \underline{u}_m - C_1 \underline{u}_m = \underline{f}_m, \tag{3.10}$$

where we keep the same notation in (3.5) because D_1 and C_1 are the same as before in structure, and $\underline{u}_m$ and $\underline{f}_m$ are also the same except that now $s_{i,j}$ represent nodes rather than interpolation points. The main point to note is that D_1 is again a block tridiagonal matrix.

3.3. Numerical methods for the splitting Method B. For partition $P_{\underline{\varepsilon}}^m$ of (2.9) for operators, we now investigate the structure of discretized systems when the underlying approximation is again based on partition P^m of (2.4). The relevance of $\underline{\varepsilon}$ is considered in relation to nodes $\{s_{i,j}\}$. Essentially, we want to choose $\underline{\varepsilon}$ small, and in particular if all nodes $\{s_{i,j}\}$ are outside the set $\bigcup_{j=1}^{m-1} \bar{I}_{\varepsilon_j}$, the singular operators give rise to block diagonal matrices.

3.3.1. The general case. Suppose r_{ε_j} denotes the number of nodes falling inside interval $\bar{I}_{\varepsilon_j}$ for $j = 1, \cdots m-1$ and r_j denotes the number of nodes inside $\bar{I}_j$ for $j = 1, \cdots, m$. Then the total number of nodes is given by

$$n = \sum_{i=1}^{m} r_i + \sum_{j=1}^{m-1} r_{\varepsilon_j} = \begin{cases} mr, & \text{open}, \\ m(r-1)+1, & \text{closed}, \end{cases} \tag{3.11}$$

where 'open' and 'closed' refer to the type of quadrature methods if the underlying method is as in §3.2. If the method is as in §3.1, then 'open' and 'closed' refer to piecewise and continuous approximations respectively.

Using either method of §3.1 or of §3.2, the operator equation (1.2) based on (2.10) is discretized, and we obtain

$$\lambda \underline{u}_m - D_2 \underline{u}_m - C_2 \underline{u}_m = \underline{f}_m, \tag{3.12}$$

where $\underline{u}_m$ and $\underline{f}_m$ are as understood in (3.5) and (3.10). Here D_2 is now a block tridiagonal matrix of size $(2m-1) \times (2m-1)$, with the i-th diagonal block of size $r_i \times r_i$ (or $r_{\varepsilon_i} \times r_{\varepsilon_i}$) and off-diagonal blocks of rectangular matrices. Again $K = D_2 + C_2$ is a block matrix.

Since our intention is to invert matrix $(\lambda I - D_2)$ (or rather to solve the linear system $(\lambda I - D_2)\underline{x} = \underline{y}$; see the next section), we should generally choose each ε_i such that r_{ε_i} is small compared to r_i so as to have an inexpensive inversion process. For example, for $m = 3$ and quartic approximations with $r_1 = r_2 = r_3 = 3$ and

ETNA
Kent State University
etna@mcs.kent.edu

$r_{\varepsilon_1} = r_{\varepsilon_2} = 1$ are used, the structure of matrix D_2 is of the following form (a 5×5 block tridiagonal matrix)

$$
(3.13)\quad D_2 = \begin{pmatrix}
\boxed{\begin{matrix}\times & \times & \times \\ \times & \times & \times \\ \times & \times & \times\end{matrix}} & \boxed{\begin{matrix}\times \\ \times \\ \times\end{matrix}} & & & \\
\boxed{\begin{matrix}\times & \times & \times\end{matrix}} & \boxed{\times} & \boxed{\begin{matrix}\times & \times & \times\end{matrix}} & & \\
& \boxed{\begin{matrix}\times \\ \times \\ \times\end{matrix}} & \boxed{\begin{matrix}\times & \times & \times \\ \times & \times & \times \\ \times & \times & \times\end{matrix}} & \boxed{\begin{matrix}\times \\ \times \\ \times\end{matrix}} & \\
& & \boxed{\begin{matrix}\times & \times & \times\end{matrix}} & \boxed{\times} & \boxed{\begin{matrix}\times & \times & \times\end{matrix}} \\
& & & \boxed{\begin{matrix}\times \\ \times \\ \times\end{matrix}} & \boxed{\begin{matrix}\times & \times & \times \\ \times & \times & \times \\ \times & \times & \times\end{matrix}}
\end{pmatrix}.
$$

3.3.2. The special case. The case when all nodes are outside $\bigcup_{j=1}^{m-1} \bar{I}_{\varepsilon_j}$ is of interest because further simplifications occur. For piecewise polynomial approximations, this means that the approximation is globally discontinuous or piecewise continuous in general. For quadrature methods, this case corresponds to the use of open type quadrature formula, see (3.6). Gaussian quadrature formulas are of this type; [19]. Here $r_{\varepsilon_1} = \cdots = r_{\varepsilon_{m-1}} = 0$. Then in (3.12), the structure of D_2 reduces to the simpler form

$$
D_2 = \begin{pmatrix} K_{11} & & & \\ & K_{22} & & \\ & & \ddots & \\ & & & K_{m,m} \end{pmatrix}. \tag{3.14}
$$

where each diagonal block is a $r \times r$ matrix; cf. (3.6).

REMARK 3.1. *As long as the underlying numerical method is of 'open' type as implied in (3.11), we may always choose $\underline{\varepsilon}$ to be so small that the above conditions hold ($r_{\varepsilon_i} = 0$ for all i), and we still have the structure of D_2 as given in (3.14) when all ε_i are positive.*

3.3.3. Computational complexity. With our new splitting method, the work required to invert D_2 or to solve $D_2\underline{x} = \underline{y}$ (for the preconditioner in §4) is determined by the size of $\underline{\varepsilon}$. There may be two approaches one can adopt for choosing $\underline{\varepsilon}$ to ensure an $O(n^2)$ efficiency for the subsequent iterative method; see §5.

(1) Fix $r_{\varepsilon_j} \equiv k_1$ (for some integer $k_1 \geq 1$, independent of $n = mr$), and $\varepsilon_j < \min\{(s_j - s_{j-1}), (s_{j+1} - s_j)\}$. Then the solution of $D_2\underline{x} = \underline{y}$ requires $O\left(m(r + k_1)^3\right) = O(n)$ operations.
(2) Choose $1 \leq r_{\varepsilon_j} \leq k_2 \min\{r, n^{1/3}\}$ for some fixed integer $k_2 \geq 1$, and again $\varepsilon_j < \min\{(s_j - s_{j-1}), (s_{j+1} - s_j)\}$. Then the solution of $D_2\underline{x} = \underline{y}$ requires $O(m(r + \max_j r_{\varepsilon_j})^3)$, i.e., at most $O(n^2)$ arithmetic operations.

Here we have shown how to choose $\underline{\varepsilon}$, given a discretization (mesh), to ensure the $O(n^2)$ efficiency (see §§4-5). But conversely if we use different discretizations with an increasing number of mesh points to solve the same integral equation, given a prescribed $\underline{\varepsilon}$, then r_{ε_j}'s will be free parameters depending on n and thus $D_2\underline{x} = \underline{y}$ will be asymptotically expensive to solve as $n \to \infty$, requiring $O(n^3)$ operations in theory, though in practice we may still be able to observe some gains in efficiency from using such a preconditioner; see [6, 10, 33] for related work.

4. Reformulated integral equations and preconditioning. We have considered numerical methods for discretizing the original (or singular) integral equation i.e. $(\lambda u - \mathcal{K}u) = f$. We now first reformulate the above integral equation and then discretize subsequently. It turns out that the discretized equations of the singular integral equation (SIE) and the reformulated integral equation (RIE) only differ by a matrix multiplied to the latter. This matrix serves as a natural preconditioner, which is a key to understand the vast difference in eigenvalue spectrum of the two coefficient matrices. Note that the RIE has a compact operator plus a multiple of an identity operator. Consequently, iterative methods are only efficient for the discretized RIE's (see §5).

As the subsequent discussion will apply to both splitting methods (A and B of §2), we simply denote the original SIE as ($\mathcal{K} = \mathcal{D} + \mathcal{C}$, refer to (1.1) and (2.1)))

$$(\lambda - \mathcal{D} - \mathcal{C})u = f \tag{4.1}$$

and the corresponding RIE as

$$(\mathcal{I} - (\lambda - \mathcal{D})^{-1}\mathcal{C})u = (\lambda - \mathcal{D})^{-1} f. \tag{4.2}$$

The useful observation is that $\mathcal{K}$ is non-compact while $(\lambda - \mathcal{D})^{-1}\mathcal{C}$ is compact as $(\lambda - \mathcal{D})^{-1}$ is bounded and $\mathcal{C}$ is compact.

Adopting the numerical methods of §3, equations (4.1)-(4.2) may be discretized to give

$$(\lambda - D - C)\underline{u}_m = f \tag{4.3}$$

and

$$(I - (\lambda I - D)^{-1}C)\underline{u}_m = (\lambda I - D)^{-1}\underline{f}_m, \tag{4.4}$$

where we have shown in the previous section that D and also $(\lambda I - D)$ are at most block tridiagonal matrices. We can claim that the eigenvalues of the coefficient matrix of (4.4) cluster at the point 1 as those of the operator in (4.2) do. This is a desirable property for the iterative method in the next section. The remaining question is whether it is feasible to calculate $\underline{x} = (\lambda I - D)^{-1}\underline{y}$ which would be required by most iterative methods. This can be achieved by solving $(\lambda I - D)\underline{x} = \underline{y}$ which is not expensive because D is of simple forms.

<u>Remark</u> 4.1. *In general, $\mathcal{K}$ as an operator may be split into the form in (4.1) for any strongly elliptic operator; see [37]. As long as $(\lambda - \mathcal{D})$ is invertible, the arguments for preconditioning as in (4.4) would follow. However, if the linear system $(\lambda I - D)\underline{x} = \underline{y}$ is as difficult to solve as the unpreconditioned system $(\lambda I - D - C)\underline{x} = \underline{y}$, there will be no gains from solving the preconditioned equations.*

5. The conjugate gradient method for compact operator equations. We now proceed to seek the solution of the discretized non-compact equation (4.3) by solving the preconditioned equation (4.4). For this kind of compact operator equations, the conjugate gradient (CG) method can be seen to be efficient from these theoretical results :

- For a linear system $A\underline{x} = \underline{b}$ with a symmetric positive definite (SPD) matrix A of size $n \times n$, the conjugate gradient method starts with an initial guess $\underline{x}_0$ and generates a sequence of approximations $\underline{x}_1, \underline{x}_2, \cdots$. Then there exists a correspondence between generating the i-th approximation and fitting a i-th degree polynomial at all eigenvalues of A with the minimal error in the weighted and discrete L_2-norm; see [24, 7]. Therefore if most eigenvalues of A are clustered at one point, only a few CG iterations will be needed to obtain a required numerical solution.
- Following the above result, let $[\alpha, \beta]$ be a small interval containing most of the eigenvalues of A and t be the small number of eigenvalues which are outside $[\alpha, \beta]$. Then it can be shown that after $i + t^*$ CG iterations (for some $t^* \geq t$), the energy norm of the solution is given by

$$E(\underline{x}_{i+t^*}) \leq 4 \left(\frac{\sqrt{C} - 1}{\sqrt{C} + 1} \right)^{2i} E(\underline{x}_0), \qquad \text{for} \quad i = 1, 2, \cdots, \tag{5.1}$$

  where $C = \beta/\alpha$ is the pseudo-condition number in 2-norm and $E(\underline{x}) = (\underline{x} - \underline{x}^*)^T A(\underline{x} - \underline{x}^*)$ with $\underline{x}^*$ denoting the true solution $A^{-1}\underline{b}$; see [2].
- When applied directly to an operator equation $\mathcal{A}x = b$ in the Hilbert space where $\mathcal{A} = \lambda\mathcal{I} + \Lambda$ is positive definite and Λ is a compact operator, the CG method shows super-linear convergence. For the CG sequence, define the error by $e_i = x - x_i$. Then we have the following super-linear convergence result, see [38],

$$\|e_i\| \leq (c_i)^i \|e_0\| \qquad \text{and} \qquad \lim_{i\to\infty} c_i = 0. \tag{5.2}$$

Equation (4.4) is in general unsymmetric. We shall apply the CG method to the associated normal equations, i.e.,

$$AA^*\underline{y} = \underline{b}, \tag{5.3}$$

where '$*$' denotes the conjugate transpose (or transpose in the real case), $A = (I - (\lambda I - D)^{-1}C)$ and $\underline{b} = (\lambda I - D)^{-1}\underline{f}_m$. Then the solution to (4.4) is given by $\underline{u}_m = A^*\underline{y}$. A detailed description of the algorithm may be found in [10]. We shall use the diagonal matrix in (5.3) to scale the matrix before applying the CG method.

<u>Remark</u> 5.1. *The use of normal equations is generally not recommended for solving unsymmetric systems but for systems from integral equations, this approach has been found to to be feasible; see [2]. The reason is that for systems from these compact integral operator equations, the condition number (and hence that of the normal equation) has been found to be generally small. For a discussion of a bi-conjugate gradient type method for iterative solution of nonsymmetric linear systems of equations; see [35] and the references therein.*

6. Numerical examples. We describe the application of the splitting methods to three examples. All examples are singular integral equations of Cauchy type. An example of solving the hypersingular type integral equation may be found in [3].

The numerical solution of the system of singular integral equations (SIE's) of the Cauchy type

$$\frac{1}{\pi}\int_{-1}^{1}\sum_{j=1}^{M} a_{i,j}\phi_j(t)\frac{dt}{t-x}+\int_{-1}^{1}\sum_{j=1}^{M}k_{i,j}\phi_j(t)dt=g_i(t), \tag{6.1}$$

where $x \in (-1,1)$ and $i = 1,\cdots,M$ has been considered by a number of authors; see [16, 22, 18, 36] among others. As is known, the solution $\phi_j(t)$ of (6.1) is singular at $t=\pm1$, and the singular behaviour is characterized by its fundamental solutions

$$R_j(t)=(1+t)^{\alpha-\frac{1}{2}}(1-t)^{\beta-\frac{1}{2}}, \tag{6.2}$$

where $\alpha,\beta = 0,\pm1$; $\alpha-\frac{1}{2},\beta-\frac{1}{2}\in(-1,1)$, and the *index* of the SIE is $\kappa=-(\alpha+\beta)=\pm1$. For $\kappa=1$, the solution of (6.1) is only unique if the following conditions are satisfied

$$\int_{-1}^{1}\phi_j(t)dt=C_j, \qquad j=1,\cdots,M, \tag{6.3}$$

for fixed constants $C_1,\cdots,C_M$; see [29] for a general theory.

Let us consider the following SIE of case $\kappa=1$,

$$\begin{cases} \frac{1}{\pi}\int_{-1}^{1}\frac{w(t)\phi(t)}{t-x}dt+\int_{-1}^{1}w(t)k(t,x)\phi(t)dt=f(x), & x\in(-1,1),\\ \frac{1}{\pi}\int_{-1}^{1}w(t)\phi(t)dt=C, \end{cases} \tag{6.4}$$

where $w(t)=(1-t^2)^{-1/2}$. For the Nyström method, we use the Gauss-Chebyshev quadrature for a function g, defined by

$$\frac{1}{\pi}\int_{-1}^{1}w(t)g(t)dt=\frac{1}{m}\sum_{i=1}^{m}g(t_i) \quad \text{where} \quad t_i=\cos\frac{2i-1}{2m}\pi. \tag{6.5}$$

When applying (6.5) to the Cauchy singular integral in (6.4), we have

$$\frac{1}{\pi}\int_{-1}^{1}\frac{\phi(t)}{t-x}dt=\frac{1}{m}\sum_{i=1}^{m}\frac{\phi(t_i)}{t_i-x}+\frac{U_{m-1}(x)}{T_m(x)}, \tag{6.6}$$

where $x\neq t_i$, $i=1,\cdots,m$, and U_{m-1} is the $(m-1)$-th order Chebyshev polynomial of the second kind, while T_m is the m-th order of the first kind; see [23]. Now choose as collocation points the roots of U_{m-1}, i.e., $x=u_j=\cos\frac{j}{m}\pi$ for $j=1,\cdots,m-1$. Then (6.4) may be discretized as

$$\begin{cases} \frac{1}{m}\sum_{i=1}^{m}\frac{\phi(t_i)}{t_i-u_j}dt+\frac{\pi}{m}\sum_{i=1}^{m}k(t_i,u_j)\phi(t_i)dt=f(u_j), \text{ for } j=1,\cdots,m-1,\\ \frac{1}{m}\sum_{i=1}^{m}\phi(t_i)dt=C, \end{cases} \tag{6.7}$$

see also [16, 22]. We solve the above system using the methods of §4-5.

To first make use of the results of §3, we define the partition P^m of $[-1,1]$ based on $-1=u_0<u_1<\cdots<u_{m-1}<u_m=1$ for $u_j=\cos\frac{j}{m}\pi$ $(j=0,1,\cdots,m)$. We then

ETNA
Kent State University
etna@mcs.kent.edu

modify the partition P^m to produce $P^m_{\underline{\varepsilon}}$ by introducing $\varepsilon_j = \min\left(\frac{u_j - t_j}{100}, \frac{t_{j+1}-u_j}{100}\right)$ for $j = 1, \cdots, m-1$ so that we have $r_{\varepsilon_1} = \cdots = r_{\varepsilon_{m-1}} = 0$ while $r = 1$. Thus, with $\{t_j\}$ as nodal points, and since $u_{j-1} < t_j < u_j$, the block matrix in (3.14) becomes a simple diagonal matrix of size $m \times m$ and is given by

$$D_2 = \frac{1}{m}\begin{pmatrix} \frac{1}{t_1-u_1} & & & & \\ & \frac{1}{t_1-u_1} & & & \\ & & \ddots & & \\ & & & \frac{1}{t_1-u_1} & \\ & & & & 1 \end{pmatrix}. \tag{6.8}$$

We now demonstrate the effectiveness of the preconditioner D_2 for three examples.

Example 1.

$$\begin{cases} \frac{1}{\pi}\int_{-1}^{1}\frac{w(t)\phi(t)}{t-x}dt = 0, & x \in (-1,1), \\ \frac{1}{\pi}\int_{-1}^{1} w(t)\phi(t)dt = 1, \end{cases} \tag{6.9}$$

which has the exact solution $\phi(x) = 1$; see [36].

Example 2.

$$\begin{cases} \frac{1}{\pi}\int_{-1}^{1}\frac{w(t)\phi(t)}{t-x}dt = U_7(x), & x \in (-1,1), \\ \frac{1}{\pi}\int_{-1}^{1} w(t)\phi(t)dt = 0, \end{cases} \tag{6.10}$$

which has the exact solution $\phi(x) = T_8(x)$ with T_8 and U_7 being Chebyshev polynomials of orders 8 and 7 respectively; see [36].

Example 3.

$$\begin{cases} \frac{1}{\pi}\int_{-1}^{1}\frac{w(t)\phi(t)}{t-x}dt + \int_{-1}^{1}\frac{(t^2-x^2)^2}{t^2+x^2}w(t)\phi(t)dt = f(x), & x \in (-1,1), \\ \frac{1}{\pi}\int_{-1}^{1} w(t)\phi(t)dt = 0, \end{cases} \tag{6.11}$$

which has the exact solution $\phi(x) = x|x|$; see [12].

Tables 6.1-6.3 show the numerical results. We compare the performance of the preconditioned conjugate gradient method (PCGM) with the unpreconditioned version (CGM). The tests are carried on a SUN-10 and CPU times are shown, together with iteration steps required to reduce the residual error to below 10^{-10} and the numerical solution error. This error refers to the root mean square (RMS) error of the computed solution against the exact solution at all nodal points. Let E_n be such an error obtained for the numerical solution by applying Gaussian elimination to (6.7). Then E_n may be considered as a measure of the discretization error.

Further we have re-computed all cases by iterating until the residual error is of the same magnitude as the discretization error. In Tables 6.1-6.3, the preconditioned conjugate gradient method and the unpreconditioned version with such a stopping criterion are respectively denoted as PCGM* and CGM*; see [31] for an earlier use of this kind of stopping criteria.

From Tables 6.1-6.3 with all three examples, we can clearly see the dramatic improvements achieved with the PCGM in terms of fast convergence. In the first two examples, the analytical solutions are simple, and we do not need to use large values

TABLE 6.1
Example 1 — Convergence results

Numerical Method	CGM Steps	CGM* Steps	PCGM Steps	PCGM* Steps
m = 20	10	10	3	2
m = 40	16	20	3	2
m = 80	29	38	3	2
m =160	49	72	3	2
m =320	90	144	3	3
m =640	175	299	3	3

TABLE 6.2
Example 2 — Convergence results

Numerical Method	CGM Steps	CGM* Steps	PCGM Steps	PCGM* Steps
m = 20	13	12	3	3
m = 40	28	23	3	3
m = 80	61	43	3	3
m =160	119	72	3	3
m =320	224	128	3	3
m =640	441	239	3	3

of m in order to get very accurate results. The measured errors are small for all cases in these two examples.

REMARK 6.1. *In separate experiments, we have tried to use the simple Gauss-Seidel iterations to solve the linear system of (6.7) but observed no convergence in the cases tested. To present the problem from an algebraic point of view, we show below a typical coefficient matrix for Example 3 (with $m = 6$)*

$$\begin{pmatrix} 10.0 & -6.3 & -1.6 & -0.9 & -0.6 & -0.5 \\ 2.1 & 4.8 & -4.1 & -1.3 & -0.8 & -0.7 \\ 1.0 & 1.4 & 3.9 & -3.9 & -1.4 & -1.0 \\ 0.7 & 0.8 & 1.3 & 4.1 & -4.8 & -2.1 \\ 0.5 & 0.6 & 0.9 & 1.6 & 6.3 & -10.0 \\ 1.0 & 1.0 & 1.0 & 1.0 & 1.0 & 1.0 \end{pmatrix},$$

which can be seen not to be diagonally dominant. The eigenvalues of the preconditioned coefficient matrix cluster at 1.

7. Singular boundary integral equations. When the singular integral equation of the form (1.1) is obtained from boundary integral equation (BIE), the two operators $\mathcal{K}_{1,m}$ and $\mathcal{K}_{m,1}$ are singular and not compact because $u(a) = u(b)$ and the kernel function $k(s,t)$ for both these operators is singular; see §2 and (2.1). Therefore, the operator splitting methods of §2 must be modified accordingly to (for

ETNA
Kent State University
etna@mcs.kent.edu

TABLE 6.3
Example 3 — Convergence results

Numerical Method	CGM			CGM*	PCGM			PCGM*
	Steps	CPU	Error	Steps	Steps	CPU	Error	Steps
m = 20	12	4.0E-2	1.5E-4	16	3	2.0E-2	1.5E-5	4
m = 40	29	4.1E-1	2.6E-5	33	3	4.0E-2	2.6E-5	4
m = 80	53	3.2E+0	4.7E-6	72	3	1.8E-1	4.7E-6	4
m = 160	101	2.6E+1	8.2E-7	141	3	7.8E-1	8.2E-7	4
m = 320	186	1.7E+2	1.5E-7	267	3	3.5E+0	1.5E-7	4
m = 640	356	1.3E+3	3.1E-8	501	3	1.4E+1	3.1E-8	4

$\mathcal{K} = \bar{\mathcal{D}}_1 + \bar{\mathcal{C}}_1 = \bar{\mathcal{D}}_2 + \bar{\mathcal{C}}_2)$

$$(7.1) \quad \bar{\mathcal{D}}_1 = \begin{pmatrix} \mathcal{K}_{1,1} & \mathcal{K}_{1,2} & & \mathcal{K}_{1,m} \\ \mathcal{K}_{2,1} & \mathcal{K}_{2,2} & \ddots & \\ & \ddots & \ddots & \mathcal{K}_{m-1,m} \\ \mathcal{K}_{m,1} & & \mathcal{K}_{m,m-1} & \mathcal{K}_{m,m} \end{pmatrix} \qquad \textbf{Method A}$$

and

$$(7.2) \quad \bar{\mathcal{D}}_2 = \begin{pmatrix} \mathcal{K}_{\varepsilon_0,\varepsilon_0} & \mathcal{K}_{\varepsilon_0,1} & & \mathcal{K}_{\varepsilon_0,\varepsilon_m} \\ \mathcal{K}_{1,\varepsilon_0} & \mathcal{K}_{1,1} & \ddots & \\ & \ddots & \ddots & \mathcal{K}_{m,\varepsilon_m} \\ \mathcal{K}_{\varepsilon_m,\varepsilon_0} & & \mathcal{K}_{\varepsilon_m,m} & \mathcal{K}_{\varepsilon_m,\varepsilon_m} \end{pmatrix} \qquad \textbf{Method B}$$

where for the latter method

$$(7.3)\, \mathcal{K} = \begin{pmatrix} \mathcal{K}_{\varepsilon_0,\varepsilon_0} & \mathcal{K}_{\varepsilon_0,1} & \mathcal{K}_{\varepsilon_0,\varepsilon_1} & \mathcal{K}_{\varepsilon_0,2} & \cdots & \mathcal{K}_{\varepsilon_0,\varepsilon_{m-1}} & \mathcal{K}_{\varepsilon_0,m} & \mathcal{K}_{\varepsilon_0,\varepsilon_m} \\ \mathcal{K}_{1,\varepsilon_0} & \mathcal{K}_{1,1} & \mathcal{K}_{1,\varepsilon_1} & \mathcal{K}_{1,2} & \cdots & \mathcal{K}_{1,\varepsilon_{m-1}} & \mathcal{K}_{1,m} & \mathcal{K}_{1,\varepsilon_m} \\ \mathcal{K}_{\varepsilon_1,\varepsilon_0} & \mathcal{K}_{\varepsilon_1,1} & \mathcal{K}_{\varepsilon_1,\varepsilon_1} & \mathcal{K}_{\varepsilon_1,2} & \cdots & \mathcal{K}_{\varepsilon_1,\varepsilon_{m-1}} & \mathcal{K}_{\varepsilon_1,m} & \mathcal{K}_{\varepsilon_1,\varepsilon_m} \\ \mathcal{K}_{2,\varepsilon_0} & \mathcal{K}_{2,1} & \mathcal{K}_{2,\varepsilon_1} & \mathcal{K}_{2,2} & \cdots & \mathcal{K}_{2,\varepsilon_{m-1}} & \mathcal{K}_{2,m} & \mathcal{K}_{2,\varepsilon_m} \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ \mathcal{K}_{m,\varepsilon_0} & \mathcal{K}_{m,1} & \mathcal{K}_{m,\varepsilon_1} & \mathcal{K}_{m,2} & \cdots & \mathcal{K}_{m,\varepsilon_{m-1}} & \mathcal{K}_{m,m} & \mathcal{K}_{m,\varepsilon_m} \\ \mathcal{K}_{\varepsilon_m,\varepsilon_0} & \mathcal{K}_{\varepsilon_m,1} & \mathcal{K}_{\varepsilon_m,\varepsilon_1} & \mathcal{K}_{\varepsilon_m,2} & \cdots & \mathcal{K}_{\varepsilon_m,\varepsilon_{m-1}} & \mathcal{K}_{\varepsilon_m,m} & \mathcal{K}_{\varepsilon_m,\varepsilon_m} \end{pmatrix}.$$

With either splitting, the discretization using numerical methods of §3 yields a linear system with coefficient matrices of quasi-triangular structure similar to (7.1) and (7.2). Our results relating to preconditioning carry over to the present problem, but there remains the important and technical question of how to solve such linear systems with quasi-triangular structures by direct methods; see [14]. We are presently studying this issue. To illustrate the problem, let us consider the model problem of

ETNA
Kent State University
etna@mcs.kent.edu

solving $A\underline{x} = \underline{b}$, where A is is a 8×8 quasi-triangular matrix given by

$$A = \begin{pmatrix} \times & \times & & & & & & \times \\ \times & \times & \times & & & & & \\ & \times & \times & \times & & & & \\ & & \times & \times & \times & & & \\ & & & \times & \times & \times & & \\ & & & & \times & \times & \times & \\ & & & & & \times & \times & \times \\ \times & & & & & & \times & \times \end{pmatrix}.$$

The undirected graph for characterizing the sparsity of A is a closed loop which gives the natural ordering (1 2 3 4 5 6 7 8) for the 8 nodes. Employing the symmetric reverse Cuthill-McKee (RCM) gives the new ordering (1 8 2 7 3 6 4 5) for the nodes by which we can permute matrix A into $\bar{A}$ given by

$$\bar{A} = \begin{pmatrix} \times & \times & \times & & & & & \\ \times & \times & & \times & & & & \\ \times & & \times & & \times & & & \\ & \times & & \times & & \times & & \\ & & \times & & \times & & \times & \\ & & & \times & & \times & & \times \\ & & & & \times & & \times & \times \\ & & & & & \times & \times & \times \end{pmatrix}.$$

Since $\bar{A}$ is now a block triangular matrix, it is not difficult to carry out the direct solution of $\bar{A}\underline{\bar{x}} = \underline{\bar{b}}$.

<u>Remark</u> 7.1. *For the case of boundary integral equations with a fixed point singularity, i.e., equation (1.1) is singular at $t = a$ and b only, a related approach to Method A has been discussed in [33]. In terms of our notation, we first write (compare to (2.6)-(2.7)), for $s \in I_i$,*

$$(\mathcal{K}u)(s) = \int_{s_0}^{s_1} k(s,t)u(t)dt + \int_{s_1}^{s_{m-1}} k(s,t)u(t)dt + \int_{s_{m-1}}^{s_m} k(s,t)u(t)dt, \tag{7.4}$$

or, for any $s \in [a, b]$,

$$\mathcal{K}u = \begin{pmatrix} \mathcal{K}_{1,1} & \mathcal{K}_{1,2} & \mathcal{K}_{1,m} \\ \mathcal{K}_{2,1} & \mathcal{K}_{2,2} & \mathcal{K}_{2,m} \\ \mathcal{K}_{m,1} & \mathcal{K}_{m,2} & \mathcal{K}_{m,m} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_m \end{pmatrix}, \tag{7.5}$$

where u_1, u_2 and u_m are restrictions of $u \in C[a, b]$ in $I_1 = [s_0, s_1), I_2 = [s_1, s_{m-1})$, and $I_m = [s_{m-1}, s_m]$ respectively. Then the proposed splitting is simply

$$\mathcal{K} = \mathcal{D}_0 + \mathcal{C}_0, \tag{7.6}$$

where

$$\mathcal{D}_0 = \begin{pmatrix} \mathcal{K}_{1,1} & 0 & \mathcal{K}_{1,m} \\ \mathcal{K}_{2,1} & 0 & \mathcal{K}_{2,m} \\ \mathcal{K}_{m,1} & 0 & \mathcal{K}_{m,m} \end{pmatrix} \quad \textit{and} \quad \mathcal{C}_0 = \mathcal{K} - \mathcal{D}_0,$$

because in (7.5) only $\mathcal{K}_{1,2}$, $\mathcal{K}_{2,2}$ and $\mathcal{K}_{m,2}$ are compact operators. If the problem has additionally a 'moving' singularity at $t = s$ as in (2.3), then a suitable splitting is given by (7.1).

8. Conclusions and future work. We have considered the numerical solution of singular integral equations and demonstrated that iterative methods can be made efficient by an appropriate use of preconditioning. The essential idea is to first split the singular integral operator into a bounded part and a compact part, and then reformulate the integral equation so that, essentially, it is a compact perturbation of the identity. The preconditioning matrix is formed from contributions of the bounded part after discretizations and is shown to be at most a block tridiagonal matrix. Numerical experiments of solving Cauchy singular integral equations show that the proposed method is very effective.

Our method is not readily applicable to 'global' methods such as the Galerkin and spectral type methods. The problem is that preconditioners will be too expensive to work with in such cases. For singular boundary integral equations, more experiments are needed to test the viability of the proposed method. Further generalizations and experiments of the proposed approach are in progress.

9. Acknowledgement. The author wishes to thank the referee for making several constructive comments.

REFERENCES

[1] S. Amini, P. J. Harris and D. Wilton, *Coupled boundary and finite element methods for the solution of the dynamic fluid-structure interaction problem*, Lecture Notes in Engineering, vol. 77, Springer-Verlag, 1992.

[2] S. Amini and K. Chen, *Conjugate gradient method for second kind integral equations - applications to the exterior acoustic problem*, Engng. Anal. Boundary Elements, 6 (1989), pp. 72-77.

[3] S. Amini, K. Chen and P. J. Harris, *Iterative solution of the boundary element equations for the exterior acoustic problem*, ASME J. Vib. Acous., 112 (1990), pp. 257-262.

[4] K. E. Atkinson, *A Survey of Numerical Methods for the Solution of Fredholm Integral Equations of the Second Kind*, SIAM, Philadelphia, 1976.

[5] K. E. Atkinson, *A survey of boundary integral equation methods for the numerical solution of Laplace's equation in three dimensions*, in Numerical Solution of Integral Equations, M. Goldberg, ed., Plenum Press, 1990.

[6] K. E. Atkinson and I. G. Graham, *Iterative solution of linear systems arising from the boundary integral methods*, SIAM J. Sci. Stat. Comp., 13 (1992), pp. 694-722.

[7] O. Axelsson and V. A. Barker, Finite Element Solution of Boundary Value Problems, Academic Press, Orlando, 1984.

[8] C. T. H. Baker, The Numerical Treatment of Integral Equations, Clarendon Press, Oxford, 1977.

[9] A. J. Burton, *Numerical solution of acoustic radiation problems*, NPL contract Report OC5/535, Teddington, Middlesex, England, 1976.

[10] K. Chen, *Conjugate gradient methods for the solution of boundary integral equations on a piecewise smooth boundary*, J. Comput. Phys., 97 (1991), pp. 127-143.

[11] K. Chen and S. Amini, *Numerical analysis of boundary integral solution of the Helmholtz equation in domains with non-smooth boundaries*, IMA J. Numer. Anal., 13 (1993), pp. 43-66.

[12] J. A. Cuminato, *On the uniform convergence of a collocation method for a class of singular integral equations*, BIT, 27 (1987), pp. 190-202.

[13] L. M. Delves and J. L. Mohammed, Computational *Methods for Integral Equations*, Cambridge University Press, 1985.

[14] I. S. Duff, A. M. Erisman and J. Reid, *Direct Methods for Sparse Matrices*, Clarendon Press, Oxford, 1986.

[15] D. Elliott, *An asymptotic analysis of two algorithms for certain Hadamard finite-part integrals*, IMA J. Numer. Anal., 13 (1993), pp 445-462.

[16] F. Erdogan and G. D. Gupta, *On the numerical solution of singular integral equations*, Q. Appl. Math., 30 (1972), pp. 525-534.

[17] F. D. Gakov, *Boundary Value Problems*, Pergamon Press, Oxford, 1966.

[18] A. Gerasoulis, *Nyström iterative variant methods for the solution of Cauchy singular integral-equations*, SIAM J. Numer. Anal., 26 (1989), pp. 430-441.
[19] C. A. Chandler and I. G. Graham, *Product integration collocation methods for non-compact integral operator equations*, Math. Comp., 50 (1988), pp. 125-138.
[20] J. Hadamard, *Lectures on Cauchy's Problem in Linear Partial Differential Equations*, Yale University Press, New Haven, 1923.
[21] P. W. Hemker and H. Schippers, *Multigrid Methods for the Solution of Fredholm Integral Equations of the Second Kind*, Math. Comp., 36 (1982), pp. 215-232.
[22] N. I. Ioakimidis and P. S. Theocaris, *A comparison between the direct and the classical numerical methods for the solution of Cauchy type singular integral equations*, SIAM J. Numer. Anal., 17 (1980), pp. 115-118.
[23] N. I. Ioakimidis, *An iterative algorithm for the numerical-solution of singular integral-equations*, J. Comput. Phys.,43 (1981), pp. 164-176.
[24] A. Jennings, *Influence of the eigenvalue spectrum on the convergence rate of the conjugate gradient method*, J. Inst. Math. Applics., 20 (1977), pp. 61-72.
[25] R. E. Kleinman and G. F. Roach, *Iterative solutions of boundary integral-equations in acoustics*, Proc. Roy. Soc. London, Ser. A, 417 (1988), pp. 45-57.
[26] A. C. Kaya and F. Erdogan, *On the solution of integral equations with strong singularities*, in Numerical Solution of Singular Integral Equations, A. Gerasoulis and R. Vichnevetsky, eds., IMACS Publication, 1984, pp. 54-56.
[27] A. C. Kaya and F. Erdogan, *On the solution of integral-equations with a generalized Cauchy kernel*, Quart. Appl. Math., 45 (1987), pp. 455-469.
[28] R. Kress, *Linear Integral Equations*, Springer-Verlag, New York, 1989.
[29] N. I. Muskhelishvili, Singular Integral Equations, P. Noordhoff, Groningen, 1953.
[30] L. Reichel, *A method for preconditioning matrices arising from linear integral-equations for elliptic boundary-value problems*, Computing, 37 (1986), pp. 125-136.
[31] L. Reichel, *Parallel iterative methods for the solution of Fredholm integral equations of the second kind*, in Hypercube Multiprocessors 1987, M. T. Heath, ed., SIAM, Philadelphia, 1987, pp. 520-529.
[32] L. Reichel, *Fast solution methods for Fredholm integral-equations of the 2nd kind*, Numer. Math., 57 (1990), pp. 719-736.
[33] H. Schippers, *Multigrid methods for boundary integral equations*, Numer. Math., 46 (1985), pp. 351-363.
[34] H. Schippers, *Theoretical and practical aspects of multigrid methods in boundary element calculations*, in Topics in Boundary Element Research, vol. 3 - Computational Aspects, C. A. Brebbia, ed., Computational Mechanics Publications, Southampton, 1987.
[35] G. L. G. Sleijpen and D. R. Fokkema, *BICGSTAB(L) for linear equations involving unsymmetric matrices with complex spectrum*, Electron. Trans. Numer. Anal., 1 (1993), pp. 11-32.
[36] E. Venturino, *On solving singular integral-equations via a hyperbolic tangent quadrature rule*, Math. Comp., 47 (1986), pp. 159-167.
[37] W. L. Wendland, *On Galerkin collocation methods for integral equations of elliptic boundary value problems*, in Numerical Treatment of Integral Equations, eds J. Albrecht and L. Collatz, Internat. Ser. Numer. Math, vol. 53, Birkhäuser, Basel, 1980, pp. 244-275.
[38] R. Winther, *Some superlinear convergence results for the conjugate gradient method*, SIAM J. Numer. Anal., 17, (1980), pp. 14-17.