AN ANALYSIS OF SPARSE APPROXIMATE INVERSE PRECONDITIONERS FOR BOUNDARY INTEGRAL EQUATIONS

KE CHEN *

Abstract. Preconditioning techniques for dense linear systems arising from singular boundary integral equations are described and analyzed. A particular class of approximate inverse based preconditioners related to the mesh neighbour methods is known to be efficient. This paper shows that it is an operator splitting preconditioner and clusters eigenvalues for the normal equation matrix thus ensuring a fast convergence of the conjugate gradient normal method (CGN). Clustering of the eigenvalues of the preconditioned matrix and fast convergence of the generalized minimal residual method (GMRES) are also observed. For the type of problems considered, we demonstrate a crucial connection between two essential features of eigenvalue clustering for a sparse preconditioner — approximate inversion for a small cluster radius and operator splitting for a small cluster size. Experimental results from several boundary integral equations are presented.

AMS subject classifications. 65F10, 65R20

Key words. Preconditioning, Operator splitting, Approximate inversion, Singular boundary elements, Least squares solution, Conjugate gradients, CGN, GMRES

1. Introduction. In this paper we consider the efficient solution of dense linear systems $A\underline{x} = \underline{b}$ by preconditioned iterative methods, where A is a $n \times n$ unsymmetric matrix. We are concerned with those systems arising from numerical solution of singular boundary integral equations (BIE's), where preconditioning is essential for convergence of iterative methods. Special attention is given to a justification of sparse approximate inverse related preconditioners.

Most BIE's possess singularities. When the underlying integral operator is smooth or only weakly singular, it is compact and iterative methods have been proved to be efficient even without preconditioning; see [1, 5, 6, 27, 29, 38, 40]. In the special case when the singularities are due to geometric non-smoothness (e.g. corners), the operator can be non-compact but the singularities occur at fixed points. Preconditioners based on separating these fixed singularities have been studied and iterative methods for the preconditioned systems have been shown to be effective; see [8, 12, 15]. Here we consider the general case of singular BIE's e.g. an integral reformulation of the Helmholtz equations with Neumann's boundary conditions. To iteratively solve the dense linear system from discretization of singular BIE's, we consider three types of sparse preconditioners: the operator splitting preconditioner (OSP), the least squares approximate inverse preconditioner (LSAI) and the diagonal block approximate inverse preconditioner (DBAI).

All these sparse preconditioners have been used in the literature. The OSP's [13, 14, 30, 46], although admitting a number of isolated eigenvalues (see Section 2.2), can cluster eigenvalues of both the original matrix and its normal matrix but further improvements are difficult. The LSAI is a general technique and its effectiveness lies in a knowledge of a dominant sparsity pattern of the true inverse (otherwise the technique can be expensive); see [10, 18, 3, 16, 17]. The LSAI's can also cluster eigenvalues collectively but with less tight clusters unless the approximate inverse approaches the true inverse. The DBAI's, under different names and contexts, have been used by many researchers. These include the mesh neighbour preconditioner (MN) of [43], the local least-squares inverse approximation preconditioner of [42], the truncated Green's function preconditioner of [26] and the nearest neighbour preconditioner of [44] among others. The effectiveness of DBAI's has been noted in the work of [3, 4, 7, 14, 17].

* Department of Mathematical Sciences, The University of Liverpool, M & O Building, Peach Street, Liverpool L69 7ZL, UK. *(Email : k.chen@liv.ac.uk, Web : http://www.liv.ac.uk/~cmchenke).*

However despite many successful experiments, there was no analysis done to indicate why DBAI's should be effective. Here we show that the preconditioner implements a useful operator splitting, possessing the advantages of both LSAI and OSP. Thus both the preconditioned matrix and its normal matrix have eigenvalue clustering patterns. Further this is also true for the generalised case with more mesh neighbours and in the three dimensional (3D) case.

The rest of the paper is presented as follows. Section 2 provides some background information on solving BIE's and iterative solvers with preconditioning. Section 3 discusses a general OSP that can be used to establish eigenvalue clustering. Section 4 discusses the LSAI in order to introduce DBAI. Section 5 shows that the DBAI is a version of OSP and a brief discussion is followed in Section 6 generalising the results to the 3D case. Finally Section 7 gives some experimental results for three examples solved by 2 preconditioned conjugate gradient methods.

Define a linear system by $A\underline{u} = \underline{f}$ and write the preconditioned system as $M^{-1}A\underline{u} = M^{-1}\underline{f}$ (left preconditioned) or $AM^{-1}\underline{y} = \underline{f}$ with $\underline{y} = M\underline{u}$ (right preconditioned), with a preconditioner M^{-1}. As is known, there are two somewhat self-conflicting requirements of the preconditioner. The first one follows from convergence estimates of iterative methods, namely, we require M^{-1} be close to A^{-1} (or M to A). Similarly (but not always equivalently), the eigenspectrum and singular value spectrum of the preconditioned matrix (or its normal matrix if CGN is considered) should be clustered, provided all eigenvalues are not sensitive to small perturbations. The second requirement is efficiency, that is, the extra work required to find M^{-1} (or solve $M\underline{x} = \underline{z}$) should be at a minimum. Practically this means that either M or M^{-1} must be sparse or a product of sparse matrices implicitly or explicitly.

We remark that the three classes of preconditioning methods discussed in this paper all satisfy the second requirement but satisfy the first requirement in different ways — this is the key to a successful preconditioner. The first class, OSP, aims to achieve eigenvalue clustering for most eigenvalues of singular BIE's but does not approximate the inverse well. The second class, LSAI, aims to approximate A^{-1} (as the name suggests) and cluster all eigenvalues but may not cluster eigenvalues as tightly as OSP for singular equations. The third class, DBAI, appears as a special case of LSAI but is shown here to be an OSP. Therefore, DBAI can be viewed as an ideal combination of LSAI and OSP producing good approximate inversion (clustering for all eigenvalues) and operator splitting (tight clustering for most eigenvalues) for our class of singular BIE's.

2. Dense linear systems. Let $\Omega \in R^2$ denote[1] a closed domain that may be interior and bounded, or exterior and unbounded, and $\Gamma = \partial\Omega$ be its (finite part) boundary that can be parameterised by $p = (x, y) = (x(s), y(s)), a \le s \le b$. Then a boundary integral equation that usually arises from reformulating a partial differential equation in Ω can be written as

$$\alpha u(p) - \int_\Gamma \overline{k}(p,q)u(q)d\Gamma = f(p), \qquad p \in \Gamma, \tag{1}$$

or

$$\alpha u(s) - \int_a^b k(s,t)u(t)dt = f(s), \qquad s \in [a,b], \tag{2}$$

i.e. simply

$$(\alpha I - \mathcal{K})u = f. \tag{3}$$

Here u may be a density function; see [2, 7]. We do not assume that $\alpha \neq 0$ so our methods will work for both first and second kind boundary integral equations of the

[1]The three dimensional case can be described similarly; see [2], Sections 6 and 7.

Fredholm type. For the latter type, $\alpha = 1/2$ at smooth points on Γ. We assume that $\mathcal{K}$ is the full operator (not just a principal part); for the Helmholtz equation this refers to the unique formulation which is valid for all wavenumbers (see [2] and §7).

To solve the above equation numerically, we divide the boundary Γ (interval $[a,b]$) into m boundary elements (non-intersecting subintervals $I_i = (s_{i-1}, s_i)$). On each interval I_i, we may either approximate the unknown u by an interpolating polynomial of order τ that leads to a collocation method, or apply a quadrature method of τ nodes and weights w_i, that gives rise to the Nyström method. Both discretization methods approximate equation (3) by

$$(\alpha I - \mathcal{K}_n) u_n = f, \tag{4}$$

where we can write

$$\mathcal{K}_n u = \mathcal{K}_n u_n = \sum_{j=1}^{m} \left[\sum_{i=1}^{\tau} w_i k(s, t_{ji}) u_{ji} \right], \quad u_n(t_{ji}) = u(t_{ji}) = u_{ji}, \quad \text{and} \quad n = m\tau.$$

We use the vector $\underline{u}$ to denote u_{ji}'s at all nodes. By a collocation step in equation (4), we obtain a linear system of equations

$$(\alpha I - K)\underline{u} = \underline{f}, \qquad \text{or} \qquad A\underline{u} = \underline{f}, \tag{5}$$

where matrices K and A are dense and unsymmetric (in general). The conditioning of A depends on the smoothness of kernel function $k(s,t)$. A strong singularity (as $t \to s$) leads to non-compactness of operator $\mathcal{K}$ and consequently the iterative solution of equation (5) requires preconditioning.

2.1. Iterative methods. For a general unsymmetric linear system, there exist many useful iterative solvers of the conjugate gradient type. See [22, 37, 41]. Here we select the following two methods for testing our preconditioners – the conjugate gradients for the normal equation (CGN) and the generalized minimum residuals (GMRES); see [37]. We use the usual notation: $\lambda(A)$ denotes the eigenvalue spectrum of A, $\sigma(A) = \sqrt{\lambda(A^*A)}$ the singular value spectrum of A, and $\mathcal{P}_j$ the space of all polynomials of degree up to j. Similarly $\lambda^\epsilon(A)$ denotes the ϵ-pseudospectrum of A; see [37]. For $\lambda(A)$, as in [9], we assume that its members λ_j are ordered such that $|\lambda_{j+1} - \ell| \le |\lambda_j - \ell|$ for some cluster center ℓ (usually $\ell = 1$). We now briefly review the convergence estimates of these two methods to motivate the need of preconditioning.

CGN. Recall that the error of CGN at the jth iteration is determined by

$$E_j = \inf_{\Psi \in \mathcal{P}_j : \Psi(0)=1} \max_{\lambda_i \in \lambda(A^*A)} |\Psi(\lambda_i)|.$$

GMRES. The relative residual error of the GMRES method, at step j, is generally determined by

$$F_j^\epsilon = \inf_{\Psi \in \mathcal{P}_j : \Psi(0)=1} \sup_{\lambda_i \in \lambda^\epsilon(A)} |\Psi(\lambda_i)|.$$

For the case where A is diagonalizable, this residual error is bounded by the alternative and simpler quantity

$$F_j = \inf_{\Psi \in \mathcal{P}_j : \Psi(0)=1} \max_{\lambda_i \in \lambda(A)} |\Psi(\lambda_i)|,$$

provided the condition number of the matrix of A's eigenvectors is not large.

Clearly smaller values of E_j and F_j^ϵ lead to faster convergence. A good preconditioner should force these quantities to be small. For matrices whose eigenvalues are insensitive to small perturbations, one aims to design preconditioners for $\lambda(A)$ and $\lambda(A^*A)$ to cluster and thus produce small values of E_j and F_j.

By 'cluster' or 'clustering' we mean that there are a large number of eigenvalues that are inside a small interval [35] or close to a fixed point [9]. If we define, for any $\mu_1 \le \mu_2$, a complex row vector set by
$\sum_{[\ell,\mu_2]}^{[n_1,\mu_1]} =$
$\left\{ \mathbf{a}^\top \mid \mathbf{a} = (a_1, \cdots, a_n)^\top \in C^n,\ |a_j - \ell| \le \mu_2 \text{ for } j \ge 1 \text{ and } |a_k - \ell| \le \mu_1 \text{ for } k \ge n_1 \right\},$
then a more precise statement can be made as follows.

Definition 1. *Given a square matrix $A_{n\times n}$, if $\lambda(A) \in \sum_{[\ell,\mu_2]}^{[n_1,\mu_1]}$ for some relatively small n_1 (with respect to n), we say $\lambda(A)$ is clustered at point ℓ with a cluster size μ_1 and cluster radius μ_2.*
Here μ_2 is the radius of a disk, centering at ℓ, containing all the eigenvalues and μ_1 is the radius of a smaller disk that contains most of the eigenvalues (i.e. all eigenvalues except the first $n_1 - 1$).

Remark. As far as convergence of conjugate gradients methods is concerned, point clusterings imply that at step n_1 the underlying approximation in $\mathcal{P}_{n_1}$ is almost as accurate as in $\mathcal{P}_n$. In this sense, both condition number estimates (popular in the literature) and interval clusterings are not as effective as point clusterings (Definition 1) in measuring convergence.

2.2. Compactness, eigenvalue clustering and preconditioning. We recall that for a compact operator, its eigenvalues cluster at zero. For compact $\mathcal{K}$, the adjoint $\mathcal{K}^*$ and product $\mathcal{K}^*\mathcal{K}$ are also compact. Therefore in this case for A in (5), $\lambda(A) \in \sum_{[\alpha,\mu_2]}^{[n_1,\mu_1]}$ and $\lambda(A^*A) \in \sum_{[|\alpha|^2,\mu_2]}^{[n_1,\mu_1]}$ i.e. clustered at α and $|\alpha|^2$ respectively with μ_1 arbitrarily small for some suitable and fixed n_1. Further the superlinear convergence of both CGN and GMRES has been established in [45] and [36] respectively.

When $\mathcal{K}$ is not compact, we wish to use preconditioning to achieve compactness and thus eigenvalue clustering and superlinear convergence. To this end, we shall consider on one hand preconditioning a dense matrix for the general case and on the other hand preconditioning the operator. If the preconditioned operator is compact (plus an identity), we expect the preconditioned matrix and its normal to have clustered eigenvalues at 1, that in turn imply fast convergence of iterative solvers.

Before we proceed we show in Figs. 1-2 two typical examples of eigenvalue distributions $\lambda(A)$ and singular value distributions $\sigma(A)$ of an original matrix A, compared to preconditioned counterparts $C = M^{-1}A$ for the three main types of preconditioners (OSP, DBAI, LSAI). (Note the different scalings in the figures). Clearly the preconditioned cases show better clustering patterns, although of different cluster sizes and radii, than the unpreconditioned case. We have also computed the ϵ-pseudospectra for all cases and found that the preconditioned eigenvalues are not sensitive to small perturbations. This implies that all preconditioned systems can be solved efficiently by either CGN or other solvers such as GMRES because eigenvalues can be used as convergence indicators [20, 37].

Moreover, DBAI has the best singular value clustering pattern among all cases in terms of distances from 1 (both cluster size and radius). The following sections explain the different reasons why such clustering patterns occur.

3. Operator splitting preconditioners. Operator splitting is a useful technique in solving singular BIE's; see [13, 14, 30, 46] and the references therein. There are two main approaches. Both make use of the following elementary lemma.

Lemma 1. *Let linear operators $\mathcal{A}$ and $\mathcal{C}$ be defined in a normed space with $\mathcal{A}$ bounded and $\mathcal{C}$ compact. Then*

1. *operator $\mathcal{A}^{-1}\mathcal{C}$ is also compact;*
2. *operator $\mathcal{B} = \mathcal{A} - \mathcal{C}$ has a bounded inverse if $\mathcal{B}$ is injective.*

The first splitting approach is based on expanding the singular kernel into a principal term of simple forms and a smooth part of remaining terms, giving rise to two splitting operators. The latter part gives rise to a compact operator while the former to a bounded operator. Further because of the simple forms in the kernel, fast algorithms (e.g. the FFT; see [34] and [46]) are used to invert the former operator which serves as a preconditioner.

Here we apply the second idea of operator splitting, previously used in [13]. This is based on domain decomposition rather than kernel decomposition. Use the partition of §2, $[a,b] = \bigcup_{i=1}^{m} I_i$. Accordingly we can partition the variable u and vector $\underline{u}$ as follows: $u = (u_1, u_2, \cdots, u_m)^T$ and $\underline{u} = (\underline{u}_1, \underline{u}_2, \cdots, \underline{u}_m)^T$. Similarly writing the operator $\mathcal{A}$ in matrix form, we obtain the splitting $\mathcal{A} = \alpha\mathcal{I} - \mathcal{K} = \mathcal{B} - \mathcal{C}$ with $\mathcal{B} = \mathcal{I} + \overline{\mathcal{K}}$) and

$$\overline{\mathcal{K}} = \begin{pmatrix} \mathcal{K}_{1,1} & \mathcal{K}_{1,2} & & & \mathcal{K}_{1,m} \\ \mathcal{K}_{2,1} & \mathcal{K}_{2,2} & \mathcal{K}_{2,3} & & \\ & \mathcal{K}_{3,2} & \ddots & \ddots & \\ & & \ddots & \ddots & \mathcal{K}_{m-1,m} \\ \mathcal{K}_{m,1} & & & \mathcal{K}_{m,m-1} & \mathcal{K}_{m,m} \end{pmatrix}.$$

Observe that all singularities of $\mathcal{A}$ are contained in the above operator and so the smooth operator $\mathcal{C}$ is compact. Note also that, after discretisation, the corresponding matrix out of K is

$$\overline{K} = \begin{pmatrix} K_{1,1} & K_{1,2} & & & K_{1,m} \\ K_{2,1} & K_{2,2} & K_{2,3} & & \\ & K_{3,2} & \ddots & \ddots & \\ & & \ddots & \ddots & K_{m-1,m} \\ K_{m,1} & & & K_{m,m-1} & K_{m,m} \end{pmatrix}.$$

Also define matrix $B = \alpha I - \overline{K}$ and $C = K - \overline{K}$. Then from the above Lemma, it can be shown that the operator $\mathcal{B}$ is bounded. Since the operator $\mathcal{B}^{-1}\mathcal{C}$ is also compact, we can use $\mathcal{M} = \mathcal{B}$ as an operator preconditioner and $M = B$ as a matrix preconditioner. This is what we call the operator splitting preconditioner (OSP).

Thus the solution of $A\underline{u} = \underline{f}$ is reduced to that of $M^{-1}A\underline{u} = M^{-1}\underline{f}$ i.e. $[I - M^{-1}C]\underline{u} = M^{-1}\underline{f}$. Here B is in general a block quasi-tridiagonal matrix and the solution of $B\underline{x} = \underline{y}$ is via $B = LU$, where L, U are of the same sparsity structure as B apart from the last row of L and the last column of U; see [3, 14].

We remark that our techniques of constructing M, for singular operators, may be viewed as efficient regularization methods. Therefore from properties of compact operators, the preconditioned matrix $(I - M^{-1}C)$ and its normal matrix should have most of its eigenvalues clustered as demonstrated in Figures 1-2. Most sparse preconditioners (e.g. approximate inverses [31]) do not possess the latter property of the normal matrix having clustered eigenvalues because of the unsymmetric nature of the matrix (where $\lambda(M^{-1}A)$ and $\sigma(M^{-1}A)$ are not related).

Note that the underlying preconditioner $M^{-1} = B^{-1}$, although not sparse, does not approximate A^{-1} well because $\|M^{-1}A - I\|$ is not small. Solving $B\underline{x} = \underline{y}$ takes $O(kn)$ flops at each step of iterative methods where k $(= 3$ here) denotes the total band-width of B. Using Definition 1, we may write

$$\lambda(M^{-1}A) \in \sum\nolimits_{[1,\mu_2]}^{[n_1,\mu_{\rm OSP}]}$$

for a small $\mu_{\rm OSP}$ and some relatively small n_1 (with respect to n) but the poor approximation of A by M means that we cannot estimate μ_2. It has been found that, to further improve on the above preconditioner i.e. reduce $n_1, \mu_{\rm OSP}$, μ_2, increasing the band-width k alone is not efficient (or sufficient) as the improvements are only marginal unless $k \approx n$. We now consider a different class of preconditioners (i.e. LSAI) before establishing the connection of mesh neighbour preconditioners with OSP and LSAI.

4. Approximate inversion techniques. Sparse approximate inverse preconditioners are widely used for preconditioning sparse linear systems, since the earlier work of [10] and [33]. See also [18, 25, 31, 41, 16, 17] and the references therein for more details. Here we review the method briefly in order to introduce the DBAI and consider its applications to iterative solution of dense linear systems $A\underline{u} = \underline{f}$ from §2.

4.1. The LSAI approach. The problem is to construct approximations, M^{-1}, to the inverse of matrix A for which $\|AM^{-1} - I\|$ is small in some norm. Once such an M^{-1} has been constructed, we solve the preconditioned linear system $AM^{-1}\underline{w} = \underline{f}$, $\underline{u} = M^{-1}\underline{w}$.

To specify the matrix M^{-1}, denote $\mathcal{N} = \{1, 2, \cdots, n\}$, let $\mathcal{S}$ be a given set of (i,j)'s with $(i,j) \in \mathcal{N}$, and $\mathcal{G}_\mathcal{S}$ be the space of all $n \times n$ matrices that have entries in positions indexed by $\mathcal{S}$. For each column index $j = 1, \cdots, n$, define its row indices $\mathcal{S}_j = \{i : (i,j) \in \mathcal{S}\}$ and let its vector space $\mathcal{G}_{\mathcal{S}_j}$ contain all vectors that have entries in positions indexed by $\mathcal{S}_j$. Then the approximate inverse M^{-1} is calculated by solving the least squares problem

$$\min_{M^{-1} \in \mathcal{G}_\mathcal{S}} \|AM^{-1} - I\|_F^2 = \sum_{j=1}^{n} \min_{m_j \in \mathcal{G}_{\mathcal{S}_j}} \|Am_j - e_j\|_2^2, \tag{6}$$

where $M^{-1} = [m_1, \cdots, m_n]$ and $I = [e_1, \cdots, e_n]$. In theory, such a problem can be posed in any norm but the Frobenius norm leads to an easier solution. The full problem of finding M^{-1} is reduced to n standard least squares problems. The above equation (6) has been used by various researchers. The resulting preconditioner M^{-1} is obviously the right preconditioner since A is multiplied on its right.

But we can equally attempt to search for the left preconditioner M^{-1} by trying to minimize $\|M^{-1}A - I\|$:

$$\min_{M^{-1} \in \mathcal{G}_\mathcal{S}} \|M^{-1}A - I\|_F^2 = \min_{M^{-1} \in \mathcal{G}_\mathcal{S}} \|A^T M^{-T} - I\|_F^2 = \sum_{j=1}^{n} \min_{m_j \in \mathcal{G}_{\mathcal{V}_j}} \|A^T m_j - e_j\|_2^2, \tag{7}$$

where $M^{-T} = [m_1, \cdots, m_n]$, $I = [e_1, \cdots, e_n]$, $\mathcal{V}_j = \{i : (j,i) \in \mathcal{S}\}$ and $\mathcal{G}_{\mathcal{V}_j}$ contain all vectors that have entries in positions indexed by $\mathcal{V}_j$. We find it slightly convenient to describe and implement the right preconditioner.

The suitable specification of $\mathcal{S}$ will then become essential. For PDE's solved by domain type discretization methods (mainly finite element and finite difference methods), $\mathcal{S}$ is usually chosen so that M^{-1} is of some sparse structure e.g. a band matrix. Its choice may also be coupled with the solution process of least squares problems so we could start from an initial sparse specification (say a diagonal matrix) and increase the number of nonzeros adaptively in order to control the residual errors of each least squares solution under some tolerance. See [25, 31, 41].

4.2. Application to singular integral equations. For singular BIE's, we have shown in §3 that for a suitable preconditioner $M^{-1} = B^{-1}$ (left or right), its inverse $M = B$ can be of the specific sparsity structure

$$\begin{pmatrix}
\times & \times & & & & & & & \times \\
\times & \times & \times & & & & & & \\
 & \times & \times & \times & & & & & \\
 & & \ddots & \ddots & \ddots & & & & \\
 & & & \ddots & \ddots & \ddots & & & \\
 & & & & \times & \times & \times & & \\
 & & & & & \times & \times & \times & \\
 & & & & & & \times & \times & \times \\
\times & & & & & & & \times & \times
\end{pmatrix}. \tag{8}$$

As the diagonal entries of B are large, numerical evidence suggests that the structure of the largest entries in B^{-1} is similar to the structure of B. In fact, if B is strictly diagonal dominant, B^{-1} can be shown to be exponentially decaying from the diagonal. In practice, this strong condition may not be strictly satisfied. For 3D problems, see Section 6 for further discussions.

However it appears reasonable to seek a preconditioner M^{-1} of sparsity structure (8) that can be used as an approximate inverse of A. Thus $\mathcal{S}$ will represent all nonzero positions in (8).

4.3. Solution of the least-squares problem. We now consider the solution of the least-squares problem for finding the right preconditioner; the left preconditioner can be found similarly. Since matrix $M^{-1} = [m_1, \, m_2, \, \cdots, \, m_n]$ is consisted of column vectors, for each column j, the least-squares problem is to solve

$$\min_{m_j \in \mathcal{G}_{\mathcal{S}_j}} \|Am_j - e_j\|_2^2 = \min_{m_j \in \mathcal{G}_{\mathcal{S}_j}} \|\hat{A}_j m_j - e_j\|_2^2,$$

or

$$\begin{pmatrix} A_{1j_1} & A_{1j_2} & A_{1j_3} \\ \vdots & \vdots & \vdots \\ A_{j_1j_1} & A_{j_1j_2} & A_{j_1j_3} \\ A_{j_2j_1} & A_{j_2j_2} & A_{j_2j_3} \\ A_{j_3j_1} & A_{j_3j_2} & A_{j_3j_3} \\ \vdots & \vdots & \vdots \\ A_{nj_1} & A_{nj_2} & A_{nj_3} \end{pmatrix} \begin{pmatrix} M_{j_1j} \\ M_{j_2j} \\ M_{j_3j} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \tag{9}$$

or simply

$$\hat{A}_j \hat{m}_j = e_j$$

where $j_2 = j$, $\hat{m}_j = [M_{j_1j} \;\; M_{j_2j} \;\; M_{j_3j}]^T$, $m_j = [0^T \;\; \hat{m}_j \;\; 0^T]^T$, $j_1 = j-1$, $j_3 = j+1$ for $j = 2, \cdots, n$, $j_1 = n, j_3 = 2$ for $j = 1$ and $j_1 = n-1$, $j_3 = 1$ for $j = n$ due to the choice of $\mathcal{S}$ and the wrap around nature of M^{-1}.

The least squares problem (9) may be solved by the QR method [11, 24]. For the approximation using this specific pattern $\mathcal{S}$, we have the following theorem

THEOREM 1. *For the least squares problem (9) with $\hat{A}_j$ of size $n \times 3$,*

1. *The residual for the solution $\hat{m}_j$ satisfies $\|r_j\|_2 \le 1$ because the right hand side of (9) is a unit vector;*
2. *Problem (9) is equivalent in the least squares sense to the following problem with B_j of a smaller size (i.e. 4×3),*

$$B_j \hat{m}_j = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ & b_{32} & b_{33} \\ & & b_{43} \end{pmatrix} \begin{pmatrix} M_{j_1j} \\ M_{j_2j} \\ M_{j_3j} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

Further, the residual for the solution $\hat{m}_j$ can be written more specifically as

$$r_j = [0 \;\; \bar{r}_j]^T \qquad and \qquad \bar{r}_j = -\sin\theta_1 \sin\theta_2 \sin\theta_3$$

for some θ_i's (so $\|r_j\|_2 < 1$ if $A_{1j_1} \neq 0$).

Therefore the matrix residual for the approximate inverse M^{-1} will be $E = I - AM^{-1}$ and its F-norm satisfies $\|E\|_F^2 = \sum_{j=1}^n \|r_j\|_2^2 < n$ or $\|E\|_F < \sqrt{n}$.

Proof. See the appendix. □

Remark. This theorem illustrates the accuracy of inverse approximation using structure (8). More general results of this type and on eigenvalue bounds can be found in [18, 31] among others. In particular, note that the residual error $\|E\|$ is directly linked to the eigenspectrum $\lambda(M^{-1}A)$. Using Definition 1, we may write

$$\lambda(AM^{-1}) \in \sum\nolimits_{[1,\mu_{\mathrm{LSAI}}]}^{[1,\mu_{\mathrm{LSAI}}]}$$

where μ_{LSAI} is generally small depending on the approximation accuracy. This behaviour of LSAI having the same (maybe small) cluster radius as the cluster size is different from OSP having a very small cluster size but not necessarily small cluster radius. We shall show that DBAI is an interesting method that have both small cluster size and small cluster radius.

5. The DBAI and an analysis. In equation (9), we expect three rows (j_1, j_2, j_3) of $\hat{A}_j$ play a dominate role due to the singular nature of the original operator. Therefore we may approximately reduce (9) to a 3×3 system

$$\overline{A}_j \hat{m}_j = \begin{pmatrix} A_{j_1 j_1} & A_{j_1 j_2} & A_{j_1 j_3} \\ A_{j_2 j_1} & A_{j_2 j_2} & A_{j_2 j_3} \\ A_{j_3 j_1} & A_{j_3 j_2} & A_{j_3 j_3} \end{pmatrix} \begin{pmatrix} M_{j_1 j} \\ M_{j_2 j} \\ M_{j_3 j} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \tag{10}$$

which of course makes sense from a computational point of view. This modified preconditioner M^{-1}, of form (8), is a DBAI preconditioner. This is the so-called method of mesh neighbours in [43]. The same idea was used in the local least-squares inverse approximation preconditioner of [42], the truncated Green's function preconditioner of [26] and the nearest neighbour preconditioner of [44] among others.

While heuristically reasonable, computationally simple and experimentally successful, the DBAI method has not been justified in theory. Here we present results on an analysis for the method before discussing the generalized version using more mesh neighbours.

To simplify the presentation, we first give two definitions and then a simple lemma.

DEFINITION 2 ($\mathbf{Band}^+(d_L, d_U, b_L, b_U)$).
A band matrix $A_{n\times n}$ with wrap-round boundaries is called $\mathbf{Band}^+(d_L, d_U, b_L, b_U)$ if its lower and upper bandwidths are b_L and b_U respectively, and if furthermore the first d_L bands below the main diagonal are all zeros and the first d_U bands above the main diagonal are also all zeros.

DEFINITION 3 ($\mathbf{Band}^-(d_L, d_U, b_L, b_U)$).
A simple band matrix $A_{n\times n}$ (without wrap-round boundaries) is called $\mathbf{Band}^-(d_L, d_U, b_L, b_U)$ if its lower and upper bandwidths are b_L and b_U respectively, and if furthermore the first d_L bands below the main diagonal are all zeros and the first d_U bands above the main diagonal are also all zeros.

Note that the first definition here is for matrices with wrap-round boundaries while the second is for simple band matrices without wrap-rounds. In both definitions, the parameters are nonnegative integers, not exceeding $(n-1)$. Here if $d_L d_U \neq 0$, both $\mathbf{Band}^+(d_L, d_U, b_L, b_U)$ and $\mathbf{Band}^-(d_L, d_U, b_L, b_U)$ matrices have a zero diagonal. But if $d_L d_U = 0$ the diagonal information will be stated in the context. With $n = 6$ we may illustrate $\mathbf{Band}^+(0,1,2,1)$ and $\mathbf{Band}^-(0,1,2,1)$ respectively by

$$\begin{bmatrix} 0 & 0 & \times & 0 & \times & \times \\ \times & 0 & 0 & \times & 0 & \times \\ \times & \times & 0 & 0 & \times & 0 \\ 0 & \times & \times & 0 & 0 & \times \\ \times & 0 & \times & \times & 0 & 0 \\ 0 & \times & 0 & \times & \times & 0 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 0 & 0 & \times & 0 & 0 & 0 \\ \times & 0 & 0 & \times & 0 & 0 \\ \times & \times & 0 & 0 & \times & 0 \\ 0 & \times & \times & 0 & 0 & \times \\ 0 & 0 & \times & \times & 0 & 0 \\ 0 & 0 & 0 & \times & \times & 0 \end{bmatrix}.$$

Therefore, for matrices from §3, B and $\overline{K}$ are $\mathbf{Band}^+(0,0,1,1)$, and C is $\mathbf{Band}^-(1,1,n-3,n-3)$. One may verify that, for example,

$\mathbf{Band}^+(0,0,2,2) = \mathbf{Band}^+(0,0,1,1) + \mathbf{Band}^+(1,1,1,1)$, and
$\mathbf{Band}^+(d_L, d_U, b_L{+}3, b_U{+}4) = \mathbf{Band}^+(d_L, d_U, 3, 4){+}\mathbf{Band}^+(d_L{+}3, d_U{+}4, b_L, b_U)$.

LEMMA 2 (Multiplication of band matrices).
If matrix $A_{n\times n}$ is $\mathbf{Band}^+(0,0,b_{L_1}, b_{U_1})$, $B_{n\times n}$ is $\mathbf{Band}^+(0,0,b_{L_2}, b_{U_2})$ and $C_{n\times n}$ is $\mathbf{Band}^-(d_{L_3}, d_{U_3}, b_{L_3}, b_{U_3})$, then

1. *AB is $\mathbf{Band}^+(0,0,b_{L_4}, b_{U_4})$, with $b_{L_4} = b_{L_1} + b_{L_2}$ and $b_{U_4} = b_{U_1} + b_{U_2}$;*
2. *AC is $\mathbf{Band}^-(d_{L_5}, d_{U_5}, b_{L_5}, b_{U_5})$, with $d_{L_5} = \max(0, d_{L_3} - b_{L_1})$, $d_{U_5} = \max(0, d_{U_3} - b_{U_1})$, $b_{L_5} = b_{L_1} + b_{U_1} + b_{L_3}$ and $b_{U_5} = b_{L_1} + b_{U_1} + b_{U_3}$.*

Proof. By simple inductions.□

We are now in a position to study the singularity separation property of the preconditioned matrix AM^{-1}.

THEOREM 2. *The DBAI preconditioner admits a diagonal operator splitting. Therefore for singular BIE's, the preconditioned matrix and its normal matrix have clustered eigenvalues.*

Proof. Partition matrix A as follows (as illustrated in Fig. 3)

$$A = D + B_2 + C_2$$

where D is the diagonal matrix of A, B_2 is $\mathbf{Band}^+(0,0,2,2)$ (with a zero diagonal) and C_2 is $\mathbf{Band}^-(2,2,n-5,n-5)$. That is,

$$B_2 = \begin{pmatrix}
 & A_{12} & A_{13} & & & & A_{1n-1} & A_{1n} \\
A_{21} & & A_{23} & A_{24} & & & & A_{2n} \\
A_{31} & A_{32} & & A_{34} & A_{35} & & & \\
 & A_{42} & A_{43} & & A_{45} & \ddots & & \\
 & & \ddots & \ddots & & \ddots & \ddots & \\
 & & & \ddots & \ddots & & \ddots & A_{n-2n} \\
A_{n-11} & & & & \ddots & \ddots & & A_{n-1n} \\
A_{n1} & A_{n2} & & & & A_{nn-2} & A_{nn-1} &
\end{pmatrix}.$$

Firstly, from a similar matrix splitting of the operator $\mathcal{A}$, we can show that the off diagonal operators are compact due to smooth kernels. Therefore assuming the original operator $\mathcal{A}$ is bounded, using Lemma 1, we see that $\overline{A}_j$ has a bounded inverse and so M^{-1} is bounded.

Secondly to work out an explicit formula for $AM^{-1} - I$ in terms of D, B_2, C_2, we have

$$AM^{-1} = DM^{-1} + B_2M^{-1} + C_2M^{-1}$$

where DM^{-1} is $\mathbf{Band}^+(0,0,1,1)$ as with M^{-1}. From Lemma 2, B_2M^{-1} is $\mathbf{Band}^+(0,0,3,3)$ with a nonzero diagonal and C_2M^{-1} is $\mathbf{Band}^-(1,1,n-3,n-3)$. Now do a simple splitting $B_2M^{-1} = B_2^{(1)} + B_2^{(2)}$ with $B_2^{(1)}$ as a $\mathbf{Band}^+(0,0,1,1)$ matrix and $B_2^{(2)}$ as $\mathbf{Band}^+(1,1,2,2)$. So defining $C_3 = B_2^{(2)} + C_2M^{-1}$ gives

$$AM^{-1} = DM^{-1} + B_2^{(1)} + C_3. \tag{11}$$

From the construction of M^{-1}, we see that

$$DM^{-1} + B_2^{(1)} = I.$$

Therefore the matrix D is implicitly inverted because equation (11) becomes

$$AM^{-1} = I + C_3. \tag{12}$$

In this formula, notice that C_3 is solely determined by terms B_2 and C_2 which correspond to compact operators. Thus matrix C_3 can be viewed as from a discretization of a compact operator and its eigenvalues and those of its normal matrix are thus clustered at 1. $\square$

The present DBAI is specified by the pattern in (8), that is, using the nearest neighbours. As is known from [42, 26, 44, 16], one may use more than one level of neighbours. In our notation, this means that we use the new pattern of a $\mathbf{Band}^{+}(0,0,s,s)$ matrix or a band $k = 2s+1$ matrix instead of a $\mathbf{Band}^{+}(0,0,1,1)$ matrix or a band 3 matrix. For brevity, we name such a preconditioner DBAI(k). Thus $s = 1$ (or $k = 3$) gives the same DBAI as before. We now consider $s > 1$ (or odd $k \geq 5$).

Then to solve for the j-th column of M^{-1}, we solve a new $k \times k$ system

$$\overline{A}_j \hat{m}_j = \begin{pmatrix} A_{j_1 j_1} & \cdots & A_{j_1 j_{s+1}} & \cdots & A_{j_1 j_k} \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ A_{j_{s+1} j_1} & \cdots & A_{j_{s+1} j_{s+1}} & \cdots & A_{j_{s+1} j_k} \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ A_{j_k j_1} & \cdots & A_{j_k j_{s+1}} & \cdots & A_{j_k j_k} \end{pmatrix} \begin{pmatrix} M_{j_1 j} \\ \vdots \\ M_{j_{s+1} j} \\ \vdots \\ M_{j_k j} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{13}$$

where $j_{s+1} = j$ always, $j_\ell = j + \ell - s - 1$ for $\ell = 1, \cdots, k$ and j_ℓ's take wrap round index values outside the range of $[1, n]$ as with (9). Compare to (10).

It remains to identify the operator splitting implied in this DBAI(k). We can prove the following.

THEOREM 3. *The DBAI(k) admits the same diagonal operator splitting as DBAI. Therefore for singular BIE's, the preconditioned matrix and its normal have clustered eigenvalues.*

Proof. Follow the similar lines of proving Theorem 2, partition matrix A as follows

$$A = D + B_{2s} + C_{2s},$$

where B_{2s} is $\mathbf{Band}^{+}(0,0,2s,2s)$ and C_{2s} is $\mathbf{Band}^{-}(2s,2s,n-2s-1,n-2s-1)$, to complete the proof. $\square$

We have thus shown that DBAI is an operator splitting preconditioner (having a small cluster size), although it appears more like a LSAI method (having a small cluster radius). So DBAI possesses advantages of both methods: inverse approximation (of LSAI) and operator splitting (of OSP). Using Definition 1, we may write for DBAI

$$\lambda(AM^{-1}) \in \textstyle\sum_{[1,\mu_{\text{LSAI}}]}^{[n_1,\mu_{\text{OSP}}]}$$

where the cluster radius μ_{LSAI} is related to the approximation error (that can be made smaller by increasing k) and μ_{OSP} is small due to operator splitting.

It remains to specify what k should be used. Since working out the preconditioner M^{-1} takes $\mathrm{O}(k^3 n)$ operations, to ensure that this work does not exceed n^2 operations (1 step of matrix vector multiplication), we suggest to choose k as an odd integer satisfying $3 \leq k \leq cn^{1/3}$ for some fixed constant c (say $c = 1$). This will be used in the experiments below.

6. Analysis of the 3D case. The analysis presented so far is mainly for 2D problems. However for 3D problems, a similar analysis can be done. For LSAI and DBAI, the essential difference is that the sparsity pattern $\mathcal{S}$ due to mesh neighbours, depending on the geometry of the surface and ordering, is more irregular and complex than that from (8). This is because the mesh neighbours are not always related to neighbouring entries in matrix A. In the 3D example of Section 7, the number of mesh neighbours varies from element to element (say one case with 4 neighbours and another with at least 9 neighbours).

However it is not difficult to understand why the analysis presented for DBAI can be generalized to this case in a similar way since all we need to do is to replace band matrices by pattern matrices. Let $\mathcal{S}$ denote the sparsity pattern of a mesh neighbouring strategy (see Section 7 for both edge and edge/vertex based strategies). This includes the case of introducing levels of neighbours as in the 2D case.

DEFINITION 4. *For any matrix B, given the sparsity pattern $\mathcal{S}$, define the pattern $\mathcal{S}$ splitting of B as*

$$B = Patt_{\mathcal{S}}(B) + Pato_{\mathcal{S}}(B)$$

where $Patt_{\mathcal{S}}(B)$ is the sparse matrix taking elements of B at location $\mathcal{S}$ and zeros elsewhere and $Pato_{\mathcal{S}}(B)$ is the complement matrix for B.
If we use M^{-1} to denote the DBAI preconditioner based on $\mathcal{S}$, then $M^{-1} = \mathrm{Patt}_{\mathcal{S}}(M^{-1})$. $\blacksquare$

We can now establish that the DBAI preconditioner admits a diagonal splitting. As in the proof of Theorem 2, partition matrix A as follows

$$A = D + C,$$

where $D = diag(A)$. Then

$$\begin{aligned} AM^{-1} &= DM^{-1} + CM^{-1} \\ &= \mathrm{Patt}_{\mathcal{S}}(DM^{-1} + CM^{-1}) + \mathrm{Pato}_{\mathcal{S}}(CM^{-1}) \\ &= I + \mathrm{Pato}_{\mathcal{S}}(CM^{-1}) \\ &= I + C_3, \end{aligned}$$

because $\mathrm{Patt}_{\mathcal{S}}$ is not affected by diagonal scaling and it also has the simple summation property. As with (12), matrix C_3 is solely determined by matrix C, which corresponds to a compact operator. Thus DBAI admits a diagonal operator splitting. Therefore the DBAI preconditioned matrix and its normal matrix have clustered eigenvalues at 1 with a small cluster size. Also from the approximation inversion property of DBAI, we know that the eigenvalues have a small cluster radius.

Remark. For both OSP and LSAI type methods, as is known, one may improve the eigenvalue clustering (in particular the cluster size for OSP and cluster radius for LSAI). However as our analysis shows, the DBAI using a more complex sparsity pattern $\mathcal{S}$ does not imply a similar operator splitting beyond the diagonal splitting (i.e. one cannot observe much change in the cluster size) although the cluster radius will be reduced. It is straightforward to establish that a block matrix version of DBAI admits a block diagonal splitting. More work is needed to find a DBAI like method admitting more than the block diagonal splitting (say tridiagonal in 2D).

7. Numerical experiments. We describe the application of three types of preconditioners OSP, LSAI and DBAI(k) discussed in the paper to three examples. For simplicity, we shall use the following abbreviations

Code name	Preconditioning Method	Set up work (flops)
OSP	OSP preconditioning A ($M_1^{-1}A$, §3)	$O(n)$
LSAI	LSAI preconditioning A (AM_2^{-1}, §4)	$3n^2$
DBAI(k)	DBAI preconditioning A (AM_3^{-1}, §5)	n^2
No	Unpreconditioned case	0

On the above list, column three indicates the amount of work needed to compute the preconditioner M^{-1} (or to prepare for solving $M\underline{x} = \underline{z}$) and this set up is only done once. The amount of work needed (by all three methods) at each iterative step is only $O(n)$. We remark that DBAI(k), as a preconditioner, has been used successfully to solve other examples. See [43, 26, 44] for solving the 3D Laplace's equation in potential theory and [42] for solving differential equations with multigrid methods.

Example 1 - Cauchy SIE. Singular integral equations (SIE) of Cauchy type are important in fracture mechanics. Consider a Cauchy SIE

$$(14)\quad \begin{cases} \dfrac{1}{\pi}\displaystyle\int_{-1}^{1} \frac{w(t)\phi(t)}{t-x}dt + \int_{-1}^{1} \frac{(t^2-x^2)^2}{t^2+x^2} w(t)\phi(t)dt = f(x), & x \in (-1,1), \\ \dfrac{1}{\pi}\displaystyle\int_{-1}^{1} w(t)\phi(t)dt = 0, \end{cases}$$

with the exact solution $\phi(x) = x|x|$, where $w(t) = (1-t^2)^{-1/2}$; see [19, 21, 24, 32] for full details. Here

$$f(x) = \frac{2}{\pi}\left((1 + x^2\omega \log\left|\frac{(1-x)\omega + 1}{(x-1)\omega + 1}\right|\right) \qquad \text{with} \qquad \omega = \frac{1}{\sqrt{1-x^2}}.$$

We choose this example because the integral equation resembles a singular BIE in the sense that the operator has a singular principal part and a smooth part.

Example 2 - Singular BIE for 2D Helmholtz equation. The exterior Helmholtz equation (see [2])

$$(\nabla^2 + \mathtt{k}^2)\phi(p) = 0$$

is of importance in acoustic scattering problems. The interior boundary Γ is the ellipse $(x/0.5)^2 + (y/2)^2 = 1$. For Neumann's boundary conditions, a unique BIE formulation due to Burton and Miller is the following:

$$(-\frac{1}{2}\mathcal{I} + \mathcal{M}_{\mathtt{k}} + i\eta\mathcal{N}_{\mathtt{k}})\phi = [\mathcal{L}_{\mathtt{k}} + i\eta(\frac{1}{2}\mathcal{I} + \mathcal{M}_{\mathtt{k}}^T)]\frac{\partial\phi}{\partial n}.$$

Here $\mathcal{L}_{\mathtt{k}}$ and $\mathcal{M}_{\mathtt{k}}$ are the usual single and double layer potential operators respectively,

$$(\mathcal{L}_{\mathtt{k}}\phi)(p) = \int_\Gamma G_{\mathtt{k}}(p,q)\phi(q)dS, \qquad (\mathcal{M}_{\mathtt{k}}\phi)(p) = \int_\Gamma \frac{\partial G_{\mathtt{k}}}{\partial n_q}\phi(q)dS,$$

and $\mathcal{M}_{\mathtt{k}}^T$ is the adjoint of $\mathcal{M}_{\mathtt{k}}$ and $\mathcal{N}_{\mathtt{k}}$ is the hypersingular operator. Recall that the 2D Green function is $G_{\mathtt{k}}(p,q) = \frac{i}{4}H_0^{(1)}(\mathtt{k}|p-q|)$ with $H_0^{(1)}$ the Hankel function. Refer to [2, 3, 4, 28]. Here we have tested the case of wavenumbers $\mathtt{k} = 1, 5, 10$ using $\eta = 1/\mathtt{k}$ and the collocation method.

Example 3 - Singular BIE for 3D Helmholtz equation. Following the last example, we solve the exterior Helmholtz equation with Neumann's boundary conditions in a 3D domain of a cylinder of radius 0.6 and height 1.8. Again the Burton and Miller formulation is used. Here the free space Green function is $G_k = e^{ik|p-q|}/(4\pi|p-q|)$. The 3D surface is approximated by quadratic functions over a triangular mesh; see [2, 6, 28]. In this 3D case, for DBAI(k), k can be a fixed integer depending on n as discussed above and used in [26]. However we have found that making k variable for the number of columns of M^{-1} and dependent on the actual number of mesh neighbours is a better strategy. This does not complicate the algorithm. Specifically, we have implemented two cases: type I — near neighbours share a common edge ($k = 4$) and type II — near neighbours share either a common edge or a common vertex ($k \geq 9$). Correspondingly the OSP and DBAI preconditioners are given the code names: OSP(I), OSP(II) and DBAI(I), DBAI(II). As the

TABLE 1
Iterative solution of Example 1

CGN method								
Size N	No	[cpu]	OSP	[cpu]	LSAI	[cpu]	DBAI(k)	[cpu]
256	129	5.76	97	4.40	138	6.55	61	2.85
512	274	80.2	132	38.8	163	50.1	72	20.0
1024	582	714	167	208	185	226	84	105
2048	1546	8154	215	1149	206	1075	100	530
GMRES(5) method								
Size N	No	[cpu]	OSP	[cpu]	LSAI	[cpu]	DBAI(k)	[cpu]
256	385	48.86	28	3.52	*	*	22	2.85
512	*	*	37	41.7	*	*	29	33.1
1024	*	*	51	251	*	*	31	140
2048	*	*	58	1275	*	*	43	974

unpreconditioned case and the LSAI method are not competitive, we omit the details but show the cpu time from using a direct solver (Gaussian elimination with partial pivoting) instead.

Tables 1-3 show both the number of iteration steps and the corresponding cpu seconds required to reduce the residual error to be of the same magnitude as the discretization error respectively for the above three examples; see [39] for an earlier use of this kind of stopping criteria. This error refers to the root mean square (RMS) error of the computed solution against the exact solution at all nodal points. An '*' entry indicates that the method cannot converge within N iteration steps or diverges. The tests were carried on a SGI IP30 using double precision (Fortran). For DBAI(k), k varies with N as discussed. The performance of the CGN and GMRES are similar, although CGN is slightly better. As the preconditioning step takes O(n) operations (negligible), cpu times for each case are proportional to the number of iterations.

The results clearly demonstrate that for dense linear systems arising from singular BIE's, all preconditioners are effective. In particular, as our theory predicted, DBAI(k) type preconditioners perform better than OSP and LSAI, and OSP is better than LSAI. Because LSAI has more '*' entries, it is the least robust method. Note that although our choice of k for DBAI(k) appears to be adequate, it may be possible that an optimal (and better) choice exists. For the re-started GMRES(m), we have used a fixed number $m = 5$ and it may be possible to find a better value; this is beyond the scope of the paper. For Helmholtz equations, all preconditioners show some dependence on the wavenumbers (`k`) and CGN is less sensitive to `k` than GMRES. Although for practical applications the size N should depend on `k` (or frequency) to achieve a certain accuracy, it is of interest to find a reason for such behaviours.

8. Conclusions. We have discussed three types of sparse approximate inverse preconditioners suitable for dense linear systems arising from singular integral equations: the operator splitting preconditioner, the least squares approximate inverse preconditioner and the diagonal block approximate inverse preconditioner. Both the operator splitting preconditioner and the least squares approximate inverse preconditioner can cluster eigenvalues — the former gives a small cluster size but not necessarily a small cluster radius, and the latter produces a small cluster radius but not necessarily a very small cluster size. We have shown that DBAI, appearing to be a LSAI preconditioner, is an operator splitting preconditioner (OSP). Therefore it can cluster eigenvalues of the preconditioned matrix giving a small cluster size as well as a small cluster radius and it can also cluster eigenvalues of the normal matrix.

Numerical experiments show that, for the type of problems considered here, the preconditioner DBAI(k) is indeed more robust than LSAI and OSP. For other types

TABLE 2
Iterative solution of Example 2

Size N	No	[cpu]	OSP	[cpu]	LSAI	[cpu]	DBAI(k)	[cpu]
CGN method								**(wavenumber k=1)**
256	128	14.4	28	3.44	*	*	16	1.73
512	257	118	58	28.4	*	*	26	15.2
1024	513	954	118	225	*	*	41	91.2
2048	1027	7120	258	1957	*	*	68	538
CGN method								**(wavenumber k=5)**
256	91	9.63	17	2.09	*	*	12	1.91
512	199	83.8	42	18.9	*	*	17	10.2
1024	421	717	86	151	*	*	30	65.4
2048	925	6413	183	1394	*	*	54	441
CGN method								**(wavenumber k=10)**
256	90	9.53	23	2.72	*	*	25	3.14
512	198	83.4	41	18.5	*	*	27	14.4
1024	419	714	87	153	*	*	31	67.1
2048	898	6226	181	1275	*	*	53	434
GMRES(5) method								**(wavenumber k=1)**
N = 256	70	34.8	12	6.37	89	44.6	10	5.72
N = 512	132	271	16	34.5	142	293	13	30.2
N =1024	252	2049	26	218	252	1957	18	162
N =2048	526	15459	33	995	410	12080	24	777
GMRES(5) method								**(wavenumber k=5)**
N = 256	62	28.5	25	11.8	95	43.9	18	8.94
N = 512	140	255	28	52.5	145	266	27	52.4
N =1024	255	1852	34	253	269	1960	36	275
N =2048	559	16428	43	1289	455	13401	43	1335
GMRES(5) method								**(wavenumber k=10)**
N = 256	57	26.2	34	15.9	121	55.8	17	8.48
N = 512	198	83.4	41	18.5	173	317	29	56.1
N =1024	305	2214	50	369	319	2322	47	356
N =2048	640	18806	62	1847	542	15956	67	2040

of problems where diagonal operator splitting is not appropriate, we expect LSAI may be more useful. Experiments involving further generalizations are in progress.

Acknowledgments. The author wishes to thank the anonymous referees who gave critical and useful comments to help clarify and improve the paper in a substantial way. He is also grateful to Peter Appleby for reading a draft version of the manuscript and making comments.

REFERENCES

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

[2] Amini S., Harris P. J. and Wilton D. T., *Coupled boundary and finite element methods for the solution of the dynamic fluid-structure interaction problem*, Springer-Verlag, 1992.

[3] Amini S. and Maines N., *Preconditioned Krylov subspace methods for boundary element solution of the Helmholtz equation*, Int. J. Numer. Meth. Engng., 41 (1998), pp.875-898.

[4] Amini S. and Maines N., *Qualitative properties of boundary integral operators and their discretizations*, Mathematics Technical report MCS-95-12, University of Salford, UK, 1995.

[5] Atkinson K. E., *Iterative variants of the Nystrom method for the numerical solution of integral equations*, Numer. Math., 22 (1973), pp.17-31.

TABLE 3
Iterative solution of Example 3

Size N	Direct cpu	OSP (I) Steps	cpu	OSP (II) Steps	cpu	DBAI (I) Steps	cpu	DBAI (II) Steps	cpu
CGN method									**(wavenumber k=1)**
576	102	31	20.2	15	24.4	41	19.5	25	15.0
2304	7188	63	641	31	943	83	605	51	430
CGN method									**(wavenumber k=5)**
576	102	15	13.5	14	24.0	16	9.1	17	11
2304	7188	24	377	15	835	31	253	19	213
CGN method									**(wavenumber k=10)**
576	102	17	14.3	15	24.4	19	10.3	18	21.1
2304	7188	18	337	17	848	18	165	19	213
GMRES(5) method									**(wavenumber k=1)**
576	102	14	33.9	7	31.6	17	34.7	12	27.5
2304	7188	26	1011	15	1195	36	1143	21	728
GMRES(5) method									**(wavenumber k=5)**
576	102	6	18.8	4	26.0	7	15.9	5	14.3
2304	7188	11	554	7	951	14	474	9	363
GMRES(5) method									**(wavenumber k=10)**
576	102	4	15.1	3	24.1	4	10.2	3	10.6
2304	7188	7	433	4	860	9	322	6	272

[6] Atkinson K. E., *Two-grid iteration methods for linear integral equations of the second kind on piecewise smooth surfaces in R^3*, SIAM. J. Sci. Comp., 15 (1994), pp.1083-1104.

[7] Atkinson K. E., *The numerical solution of Fredholm integral equations of the second kind*, Cambridge University Press, UK, 1996.

[8] Atkinson K. E. and Graham I. G., *Iterative variants of the Nystrom method for second kind boundary integral equations*, SIAM. J. Sci. Comp., 13 (1992), pp.694-722.

[9] Axelsson O., *Iterative solution methods*, Cambridge University Press, UK, 1994.

[10] Benson M. W. and Frederickson P. O., *Iterative solution of large sparse linear systems arising in certain multidimensional approximation problems*, Utilita Math., 22 (1982), pp.127-140.

[11] Bjorck A., *Numerical methods for least squares methods*, SIAM Publications, 1996.

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

[13] Chen K., *Efficient iterative solution of linear systems from discretizing singular integral equations*. Elec. Tran. Numer. Anal., 2 (1994), pp.76-91.

[14] Chen K., *On a class of preconditioning methods for dense linear systems from boundary elements*, SIAM J. Sci. Comput., 20 (1998), pp. 684-698.

[15] Chen K. and Amini S., *Numerical analysis of boundary integral solution of the Helmholtz equation in domains with non-smooth boundaries*, IMA J. Numer. Anal., 13 (1994), pp.43-66.

[16] Chow E., *A priori sparsity patterns for parallel sparse inverse preconditioners*, SIAM J. Sci. Compt., 21 (2000), pp.1804-1822.

[17] Chow E. and Saad Y., *Parallel Approximate Inverse Preconditioners*, Proc. 8th SIAM Conference on Parallel Processing for Scientific Computing, Minneapolis, MN, March 1997, pp.14-17 (http://www.llnl.gov/CASC/people/chow/pubs/history.ps).

[18] Cosgrove J. D. F., Diaz J. C. and Griewank A., *Approximate inverse preconditioners for sparse linear systems*, Int. J. Compt. Math., 44 (1992), pp.91-110.

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

[20] Embree M., *How Descriptive are GMRES Convergence Bounds?*, University of Oxford Computing Lab., NA Report 99/08 (1999);
http://web.comlab.ox.ac.uk/oucl/work/mark.embree/estimates.ps.gz

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

[22] Freund R. W., Golub G. H. and Nachtigal N. M., *Iterative solution of linear systems*, Acta Numerica, Cambridge University Press, UK, pp. 57-100, 1992.

[23] Gerasoulis A., *Nyström iterative variant methods for the solution of Cauchy singular integral-equations*, SIAM J. Numer. Anal., 26 (1989), pp. 430-441.

[24] Golub G. H. and Van Loan C., *Matrix Computation*, 2nd ed, John Hopkins University Press, USA, 1989.
[25] Gould, N. I. M. and Scott, J. A., *Sparse Approximate-Inverse Preconditioners Using Norm-Minimization Techniques*, SIAM J. Sci. Comput., 19 (1998), pp.605-625.
[26] Grama A., Kuma V. and Sameh A., *Parallel hierarchical solvers and preconditioners for boundary element methods*, SIAM J. Sci. Compt., 20 (1999), pp.337-358.
[27] Hackbusch W., *Multigrid methods and applications*, Springer-Verlag, 1985.
[28] Harris P. J., *A boundary element method for the Helmholtz equation using finite part integration*, Comp. Meth. Appl. Mech. Engng., 95 (1992), pp. 331-342.
[29] Hemker P. W. and Schippers H., *Multigrid methods for the solution of Fredholm integral equations of the second kind*, Math. Comp., 36 (1981), pp. 215–232.
[30] Holm H., Maischak M. and Stephan E. P., *The hp-version of the boundary element method for Helmholtz Screen problems*, Institute For Applied Mathematics report, IFAM No.7, University of Hannover, Germany (ftp.ifam.uni-hannover.de/pub/preprints), 1995.
[31] Huckle T. and Grote M., *Parallel Preconditioning with Sparse Approximate Inverses*, SIAM J. Sci. Comput., 18 (1998), pp.838-853.
[32] Ioakimidis N. I. and Theocaris P. S., *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.
[33] Kolotilina L. Y. and Yeremin A. Y., *On a family of two-level preconditionings of the incomplete block factorization type*, Soviet J. Numer. Anal. Math. Model., 1 (1986), pp.293-320.
[34] Lin F. R., M. K. Ng and Chan R., *Preconditioners for Wiener-Hopf equations with high order quadrature rules*, SIAM J. Numer. Anal., 34 (1997), pp.1418-1431.
[35] Parter S. V. and Wong S. P., *Preconditioning second-order elliptic operators: condition numbers and the distribution of the singular values*, J. Sci. Comput., 6 (1991), pp.129-157.
[36] Moret I., *A note on the superlinear convergence of GMRES*, SIAM J. Numer. Anal., 34 (1997), pp. 513-516.
[37] Nachtigal N. M., Reddy S. and Trefethen N., *How fast are nonsymmetric matrix iterations?*, SIAM J. Matr. Anal. Appl., 13 (1992), pp.778–795.
[38] Reichel L., *A method for preconditioning matrices arising from linear integral equations for elliptic boundary value problems*, Computing, 37 (1986), pp.125-136.
[39] Reichel L., *Parallel iterative methods for the solution of Fredholm integral equations of the second kind*, in Hypercube Multiprocessors 1987, ed. M. T. Heath, SIAM publications, pp. 520-529, 1987.
[40] Reichel L., *A matrix problem with application to rapid solution of integral equations*, SIAM J. Sci. Comput., 11 (1990), pp.263-280.
[41] Saad Y., *Iterative solution for sparse linear systems*, PWS Int. Thompson Pub. (ITP), 1996.
[42] Tang W. P. and Wan W. L., *Sparse approximate inverse smoother for multi-grid*, SIAM J. Matr. Anal. Appl., 21 (2000), pp.1236-1252.
[43] Vavasis S., *Preconditioning for boundary integral equations*, SIAM J. Matr. Anal. Appl., 13 (1992), pp. 905–925.
[44] Nabors K., Korsmeyer F. T., Leighton F. T. and White J., *Preconditioned, adaptive, multipole-accelerated iterative methods for three-dimensional potential integral equations of the first kind*, to appear (1999).
[45] Winther R., *Some superlinear convergence results for the conjugate gradient method*, SIAM J. Numer. Anal., 17 (1980), pp. 14-17.
[46] Yan Y., *Sparse preconditioned iterative methods for dense linear systems*, *SIAM J. Sci. Comp.*, 15 (1994), pp. 1190–1200.

Appendix — Proof of Theorem 1.

Proof. For an orthogonal matrix $Q = [q_1, \cdots, q_n]$, let the QR-decomposition of $\hat{A}_j$ be

$$\hat{A}_j = Q^T \begin{pmatrix} R \\ 0 \end{pmatrix}$$

and let $Qe_j = q_j$. Define $q_j = [\hat{c}^T \quad \hat{d}^T]^T$ where $\hat{c}^T$ is of size 3 and $\hat{d}^T$ is of size $n-3$. Then equation (9) is equivalent in the least squares sense to $R\hat{m}_j = \hat{c}^T$. The solution is $\hat{m}_j = R^{-1}\hat{c}^T$. The residual error will be $r_j = e_j - \hat{A}_j\hat{m}_j = Q^T[0 \quad \hat{d}^T]^T$.

The first result is trivial because q_j is a unit vector and so $\|r_j\|_2 = \|\hat{d}\|_2 \le \|q_j\|_2 = 1$. For the second result, we first multiply a permutation matrix $P_{1,j}$ (also orthogonal), permuting rows 1 and j, to equation (9). The resulting equation can be applied by 3 Householder transformations giving rise to the reduced 4×3 problem.

Further, a sequence of 3 successive Givens transformations

$$\begin{pmatrix} 1 & & & \\ & 1 & & \\ & & \cos\theta_3 & \sin\theta_3 \\ & & -\sin\theta_3 & \cos\theta_3 \end{pmatrix} \begin{pmatrix} 1 & & & \\ & \cos\theta_2 & \sin\theta_2 & \\ & -\sin\theta_2 & \cos\theta_2 & \\ & & & 1 \end{pmatrix} \begin{pmatrix} \cos\theta_1 & \sin\theta_1 & & \\ -\sin\theta_1 & \cos\theta_1 & & \\ & & 1 & \\ & & & 1 \end{pmatrix}$$

can reduce the 4×3 matrix B_j to an upper triangular matrix $(R^T \quad 0^T)^T$ and the right hand side to

$$\hat{c} = [\cos\theta_1 \quad -\sin\theta_1\cos\theta_2 \quad \sin\theta_1\sin\theta_2\cos\theta_3 \quad -\sin\theta_1\sin\theta_2\sin\theta_3]^T$$

for some θ_i's. Note that $|\bar{r}_j| \le 1$ but if $b_{11} = A_{1j_1} \ne 0$, $|\sin\theta_1| \ne 1$ so $\|r_j\|_2 < 1$. Thus the second result follows. □

Fig. 1. *Example* 1 *of eigenvalue and singular value distributions of* 4 *cases: original matrix* A, *OSP, LSAI, DBAI*

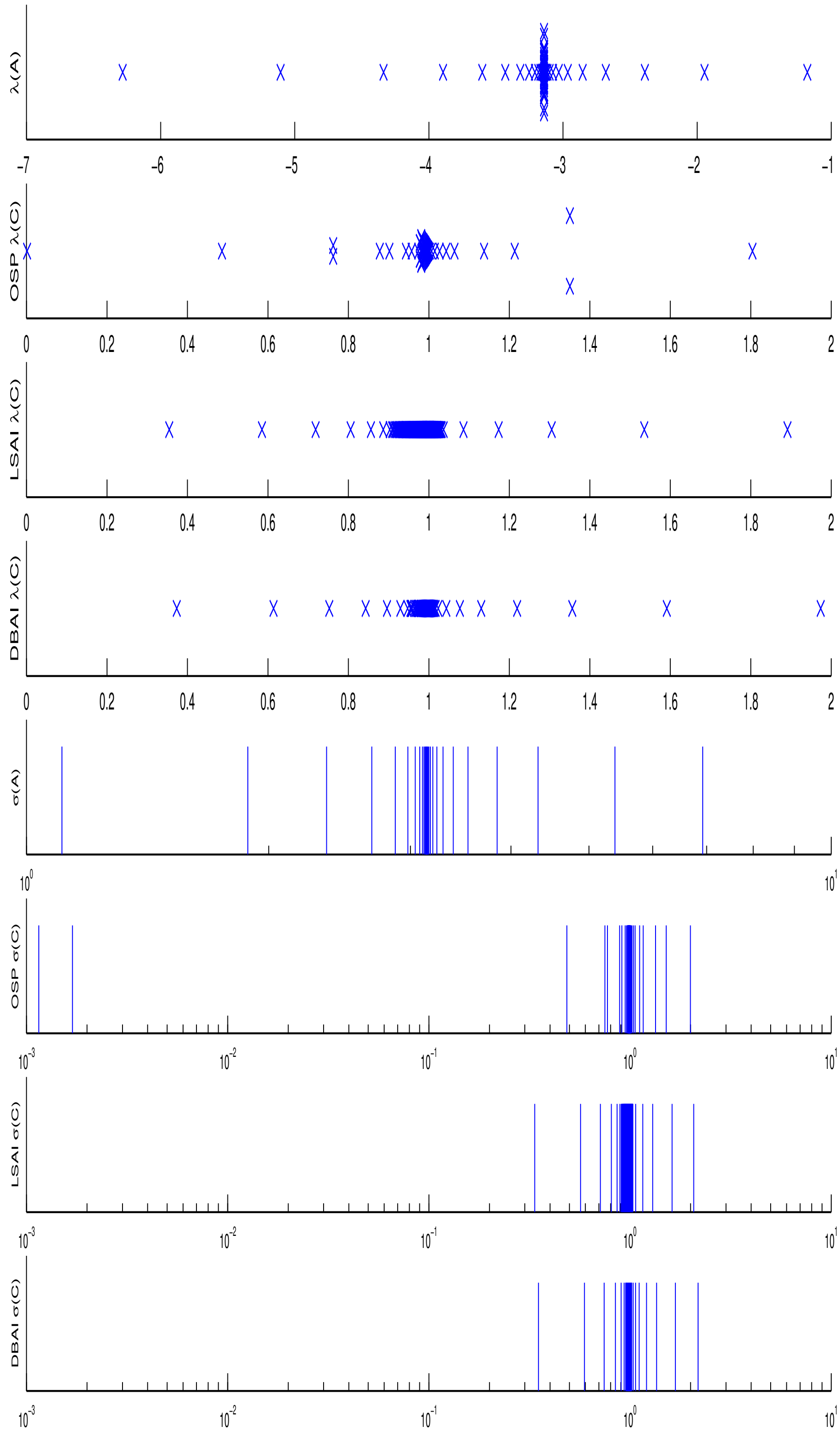

FIG. 2. *Example 2 of eigenvalue and singular value distributions of 4 cases: original matrix A, OSP, LSAI, DBAI*

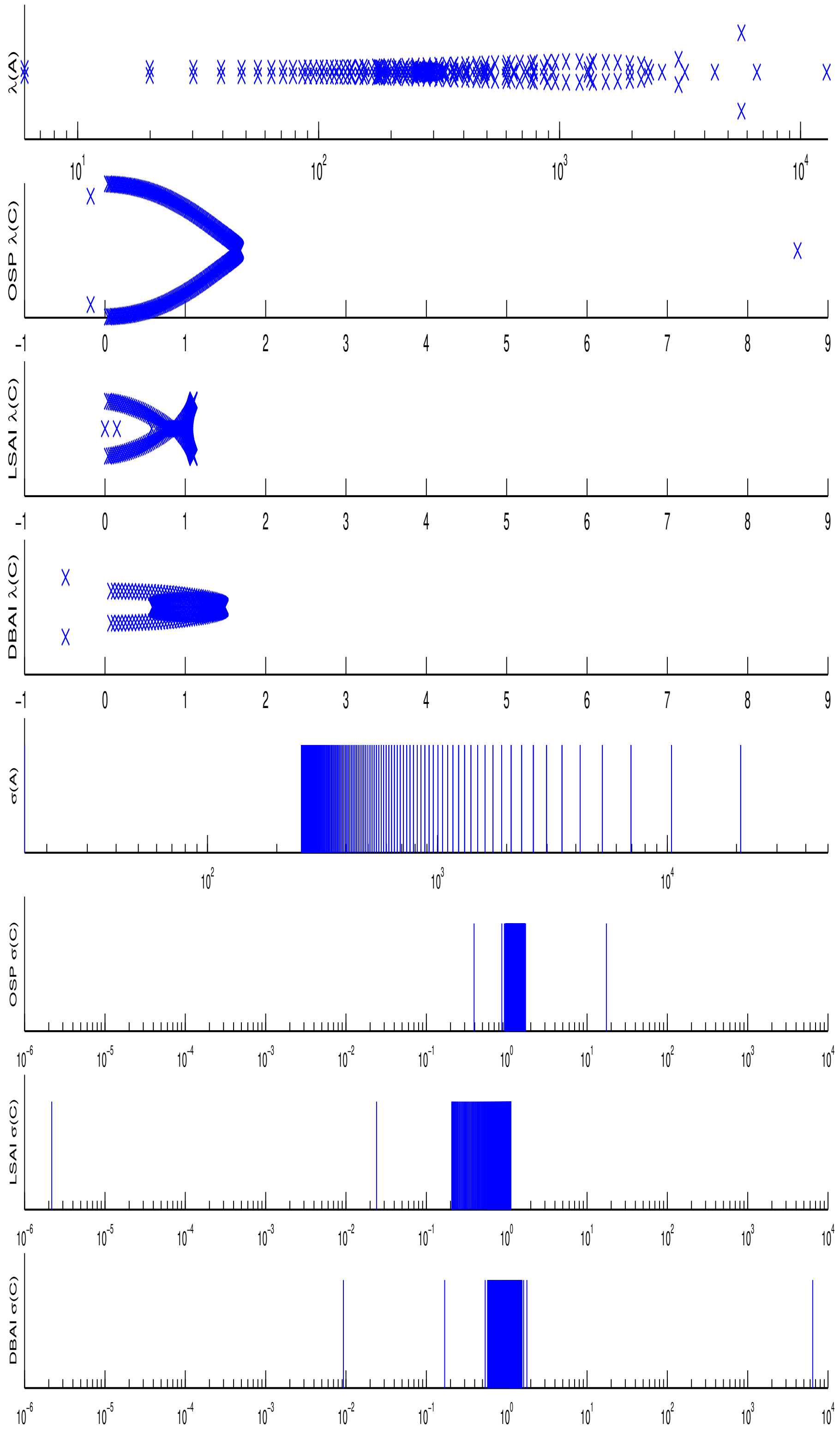

FIG. 3. *Illustration of operator splitting of DBAI* $(n = 14)$