

ADAPTIVE CROSS APPROXIMATION FOR TIKHONOV REGULARIZATION IN GENERAL FORM

T. MACH*, L. REICHEL†, AND M. VAN BAREL‡

Dedicated to Claude Brezinski on the occasion of his 80th birthday.

Abstract. Many problems in Science and Engineering give rise to linear integral equations of the first kind with a square integrable kernel. Discretization of the integral operator yields a matrix, whose singular values cluster at the origin. We describe the approximation of such matrices by adaptive cross approximation, which avoids forming the entire matrix. The choice of the number of steps of adaptive cross approximation is discussed. The discretized right-hand side represents data that commonly are contaminated by measurement error. Solution of the linear system of equations so obtained is not meaningful because the matrix determined by adaptive cross approximation is rank-deficient. We remedy this difficulty by using Tikhonov regularization and discuss how a fairly general regularization matrix can be used. Computed examples illustrate that the use of a regularization matrix different from the identity can improve the quality of the computed approximate solutions significantly.

Key words. ill-posed problem, inverse problem, sparse discretization, regularization, adaptive cross approximation

1. Introduction. Linear integral equations of the first kind,

$$\int_{\Omega_1} \kappa(s, t)x(t)dt = g(s), \quad s \in \Omega_2, \quad (1.1)$$

with a square integrable kernel κ arise in many applications, including remote sensing, computerized tomography, and image restoration. Here Ω_i denotes a subset of \mathbb{R}^{d_i} for some positive integer d_i . The integral operator in (1.1) is compact. Therefore, it has a singular value expansion, whose singular values cluster at the origin. This makes the solution of (1.1) an *ill-posed problem*; see, e.g., [13, 22] for introductions to ill-posed problems.

Discretization of (1.1) yields a linear system of equations

$$A\mathbf{x} = \mathbf{g}, \quad A \in \mathbb{R}^{n \times n}, \quad \mathbf{g} \in \mathbb{R}^n, \quad (1.2)$$

with a matrix, whose singular values coalesce at the origin. This makes the matrix A severely ill-conditioned and possibly rank-deficient; we measure the conditioning of a matrix with its condition number, which is the ratio of the largest and smallest singular values. Linear systems of equations with a matrix of this kind are often referred to as *discrete ill-posed problems*; see, e.g., [23]. We will for notational simplicity assume the matrix A to be square, however, the method described also can be applied, after minor modifications, when A is rectangular, in which case the linear system of equations (1.2) is replaced by a least-squares problem.

In many applications, the right-hand side vector \mathbf{g} in (1.2) represents measured data and is contaminated by a measurement error \mathbf{e} . Due to the severe ill-conditioning of A , straightforward solution of (1.2) typically yields a computed solution that is

*Institute for Mathematics, University of Potsdam, 14476 Potsdam, Germany. E-mail: mach@uni-potsdam.de.

†Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA. E-mail: reichel@math.kent.edu.

‡Department of Computer Science, KU Leuven, Celestijnenlaan 200A, 3001 Leuven (Heverlee), Belgium. E-mail: marc.vanbare1@cs.kuleuven.be.

severely contaminated by propagated error, and therefore is not useful. To circumvent this difficulty, the linear system of equations (1.2) commonly is replaced by a nearby problem, whose solution is less sensitive to the error \mathbf{e} in \mathbf{g} . This replacement is referred to as *regularization*. Tikhonov regularization is possibly the most popular and well understood regularization method. It replaces the linear system of equations (1.2) by a penalized least-squares problem of the form

$$\min_{\mathbf{x} \in \mathbb{R}^n} \{\|A\mathbf{x} - \mathbf{g}\|^2 + \mu\|L\mathbf{x}\|^2\}, \quad (1.3)$$

where $L \in \mathbb{R}^{p \times n}$ is referred to as the regularization matrix and $\mu > 0$ as the regularization parameter. The problem (1.3) is said to be in *standard form* when L is the identity; otherwise (1.3) is in *general form*. Throughout this paper $\|\cdot\|$ denotes the Euclidean vector norm or the spectral matrix norm.

The choice of regularization parameter μ is important for the quality of the computed solution: a too small value results in a computed solution that is contaminated by needlessly much propagated error, while a too large value yields an unnecessarily smooth solution that may lack details of interest. Generally, a suitable value of μ is not known a priori, but has to be computed during the solution process. This typically requires that (1.3) be solved for several μ -values. Methods for determining a suitable value of μ include the L-curve criterion, generalized cross validation, and the discrepancy principle; see, e.g., [6, 7, 15, 28, 32, 33] for discussions of properties of these and other methods.

The matrix L is assumed to be chosen such that

$$\mathcal{N}(A) \cap \mathcal{N}(L) = \{\mathbf{0}\}, \quad (1.4)$$

where $\mathcal{N}(M)$ denotes the null space of the matrix M . Then the Tikhonov minimization problem (1.3) has the unique solution

$$\mathbf{x}_\mu := (A^T A + \mu L^T L)^{-1} A^T \mathbf{g} \quad (1.5)$$

for any $\mu > 0$; see, e.g., [23] for details. Here and below the superscript T denotes transposition. We are interested in the situation when the matrices A and L are so large that it is impossible or undesirable to compute the solution (1.5) by Cholesky factorization of the matrix $A^T A + \mu L^T L$. In fact, we would like to avoid evaluating all the entries of A . We describe how A can be approximated by a much smaller matrix, without evaluating all matrix entries, by applying adaptive cross approximation.

The application of cross approximation to matrices that stem from the discretization of Fredholm integral equations of the second kind has received considerable attention in the literature, see, e.g., [3, 4, 17, 20, 37]; however, the use of cross approximation in the context of solving linear discrete ill-posed problems has not been thoroughly studied.

The use of adaptive approximation to determine the approximate solution of (1.3) when L is the identity is discussed in [29]. This paper extends this discussion to general regularization matrices L . Our interest in this extension of the method in [29] stems from the fact that the use of a suitably chosen regularization matrix L can deliver solutions of higher quality than $L = I$; see, e.g., [25, 31, 35] for illustrations and discussions on various ways of constructing regularization matrices. Roughly, L should be chosen so as not to damp known important features of the desired solution, while damping the propagated error stemming from the error in \mathbf{g} .

In the computed examples of Section 4, we use the *discrepancy principle* to determine $\mu > 0$. Let $\widehat{\mathbf{g}} \in \mathbb{R}^n$ denote the unknown error-free vector associated with the right-hand side \mathbf{g} in (1.2), i.e., $\mathbf{g} = \widehat{\mathbf{g}} + \mathbf{e}$. Assume that the linear system of equations with the error-free right-hand side,

$$A\mathbf{x} = \widehat{\mathbf{g}}, \quad (1.6)$$

is consistent and that a fairly accurate bound $\|\mathbf{e}\| \leq \delta$ is known. Let $\widehat{\mathbf{x}}$ denote the minimal-norm solution of (1.6). Then

$$\|A\widehat{\mathbf{x}} - \mathbf{g}\| = \|\mathbf{e}\| \leq \delta. \quad (1.7)$$

The discrepancy principle prescribes that the regularization parameter $\mu > 0$ be determined such that the Tikhonov solution (1.5) satisfies

$$\|A\mathbf{x}_\mu - \mathbf{g}\| = \eta\delta, \quad (1.8)$$

where $\eta > 1$ is a user-specified parameter that is independent of δ . It can be shown that when δ tends to zero, \mathbf{x}_μ converges to $\widehat{\mathbf{x}}$; see, e.g., [13] for a proof in a Hilbert space setting. We remark that the determination of $\mu > 0$ such that \mathbf{x}_μ satisfies (1.8) typically requires the solution of (1.3) for several μ -values.

The present paper is concerned with the situation when the matrix $A \in \mathbb{R}^{n \times n}$ in (1.2) is large. Then the evaluation of all entries of A can be quite time-consuming. Cross approximation, also known as skeleton approximation, of A reduces this time by approximating A by a matrix $M_k \in \mathbb{R}^{n \times n}$ that consists of $k \ll n$ rows and columns of A . We would like to choose the rows and columns of A so that M_k approximates A well and is easy to compute with.

This paper is organized as follows. Section 2 reviews the application of adaptive cross approximation to the approximation of A by a matrix of low rank. In Section 3, we describe the application of adaptive cross approximation to the approximation of the Tikhonov equation (1.3). Section 4 reports a few computed examples, and concluding remarks can be found in Section 5.

We conclude this section with an example that leads to a large linear discrete ill-posed problem, whose solution is difficult to compute using straightforward discretization.

Example 1.1. Consider the solution of the Fredholm integral equation of the first kind

$$\int_S \frac{\sigma(\mathbf{y})}{4\pi\epsilon_0\|\mathbf{x} - \mathbf{y}\|} d\mathbf{y} = \phi(\mathbf{x}), \quad \mathbf{x} \in S, \quad (1.9)$$

where ϕ is a given electric potential, $S \in \mathbb{R}^3$ is a surface with electrodes, $\sigma(\mathbf{y})$ denotes the density of the charge on S , and ϵ_0 stands for the electric permittivity in vacuum. We would like to determine σ from ϕ , and assume that ϕ is chosen so that (1.9) has a solution. The integral operator in (1.9) is compact. Therefore its singular values cluster at the origin; due to the singularity of the kernel at $\mathbf{x} = \mathbf{y}$, we expect the singular values to decay to zero with increasing index at a moderate rate. Nevertheless, the computation of a solution of (1.9) is an ill-posed problem. Using a weak formulation and discretization lead to a dense symmetric matrix $K = [k_{ij}]_{i,j=1}^n \in \mathbb{R}^{n \times n}$ with entries

$$k_{ij} = \int_S \int_S \frac{v_j(\mathbf{x})v_i(\mathbf{y})}{4\pi\epsilon_0\|\mathbf{x} - \mathbf{y}\|} d\mathbf{y} d\mathbf{x}, \quad i, j = 1, 2, \dots, n.$$

The singular values of the matrix K cluster at the origin; the matrix may be numerically rank-deficient when n is large.

This matrix k can be expensive to store and handle when the discretization is fine. Employing a hierarchical compression with \mathcal{H}^2 -matrices reduces the required storage to $O(n)$, with a large constant hidden in the $O(\cdot)$, and allows matrix-vector product evaluations in $O(n)$ arithmetic floating point operations (flops); see [5].

A fine discretization with $n = 262,146$ nodes results in a large, $n \times n$, dense matrix. We used the H2Lib library [27] for the computations and base this example on one of the standard examples provided in this library. Without compression, 512 GB of memory are needed to store the matrix. On a laptop computer with an Intel Core i710710U CPU and 16 GB of RAM it took 1103 s to assemble the matrix K in the compressed \mathcal{H}^2 -matrix format. The matrix required 15.45 GB of storage, thus almost all the available RAM. Carrying out one matrix-vector product evaluation required 1596 s, that is 44% more time than for assembling the matrix. The reason for this is that the $O(n)$ flops require a significant amount of communication between faster and slower storage. This example illustrates that there are linear discrete ill-posed problems of interest that are difficult to solve on a laptop computer, even if a significant amount of memory is available. It therefore is important to develop methods that are able to determine approximations of dense matrices that requires less computer storage and less CPU time for the evaluation of matrix-vector products. Unfortunately, it is not possible to solve Tikhonov regularized problems obtained from (1.9) for large values of n in reasonable time. We therefore are not able to carry out computations with this example. \square

2. Adaptive cross approximation. Cross approximation of a large matrix $A \in \mathbb{R}^{n \times n}$ determines an approximation $M_k \in \mathbb{R}^{n \times n}$ of rank at most $k \ll n$. All entries of M_k can be evaluated much faster than all entries of A , because M_k is constructed from only k rows and columns of A . We would like M_k to be an accurate approximation of A . This is achieved by a careful choice of the k rows and columns of A that define the matrix M_k . A cross approximation method is said to be *adaptive* when the rows and columns of A that determine M_k (and k) are chosen depending on properties of A revealed during the computations; see [2, 4].

We outline the adaptive cross approximation method for a general square non-symmetric matrix described in [29]. This method is an adaptation of the scheme in [17] to the approximation of the matrix of linear discrete ill-posed problems. When A is symmetric, the matrix M_k can be chosen to be symmetric. This roughly halves the storage requirement for M_k . Both the situations when A is symmetric positive definite or symmetric indefinite are discussed in [29]. We therefore will not dwell on these special cases in the present paper.

Let the matrix $A \in \mathbb{R}^{n \times n}$ be nonsymmetric and choose k rows of A with indices in $\mathbb{N} = \{1, 2, \dots, n\}$. We let $\mathbf{i}, \mathbf{j} \in \mathbb{N}^k$ denote index vectors with k entries in \mathbb{N} . The submatrices $A_{(\mathbf{i},:)}$ and $A_{(:,\mathbf{j})}$ of A are made up of the k rows with indices \mathbf{i} and the k columns with indices \mathbf{j} , respectively. Moreover, the *core matrix* $A_{(\mathbf{i},\mathbf{j})}$ is made up of k rows and columns of A . Assume that this matrix is nonsingular. Then the rows and columns of the matrix

$$M_k = A_{(:,\mathbf{j})} A_{(\mathbf{i},\mathbf{j})}^{-1} A_{(\mathbf{i},:)}$$

are equal to the corresponding rows and columns of A ; when A is of rank k , we have $M_k = A$.

Goreinov et al. [21] show that it is beneficial to choose the index vectors \mathbf{i} and \mathbf{j} so that $A_{(\mathbf{i},\mathbf{j})}$ is a submatrix of A of maximal volume, i.e., so that the modulus of the determinant of $A_{(\mathbf{i},\mathbf{j})}$ is maximal. However, it is difficult to determine such index vectors \mathbf{i} and \mathbf{j} . We therefore seek to determine a low-rank matrix M_k that is a sufficiently accurate approximation of A by a greedy algorithm. Suppose that we already have computed an approximation

$$M_{k-1} = \sum_{\ell=1}^{k-1} \mathbf{w}_\ell^{(c)} (\mathbf{w}_\ell^{(r)})^T, \quad \mathbf{w}_\ell^{(c)}, \mathbf{w}_\ell^{(r)} \in \mathbb{R}^n,$$

of rank at most $k-1$ of A . To compute the next approximation, M_k , of A of rank at most k , we determine a row index i^* and a column index j^* by looking for the index of the maximum element in magnitude in the previously computed vectors $\mathbf{w}_{k-1}^{(c)}$ (for index i^*) and $\mathbf{w}_{k-1}^{(r)}$ (for index j^*). The vector $(\mathbf{w}_1^{(r)})^T$ can be chosen as an arbitrary row of A . We will let $(\mathbf{w}_1^{(r)})^T$ be the first row of A in the computed examples of Section 4. The vector $\mathbf{w}_1^{(c)}$ can be chosen in a similar way.

In the simplest form of cross approximation, the determination of the vectors $\mathbf{w}_k^{(c)}$ and $\mathbf{w}_k^{(r)}$ only requires the entries in row i^* and column j^* of A and the elements of already computed vectors $\mathbf{w}_\ell^{(c)}$ and $(\mathbf{w}_\ell^{(r)})^T$, $\ell = 1, 2, \dots, k-1$:

$$\begin{aligned} (\mathbf{w}_k^{(r)})_j &= A_{i^*,j} - \sum_{\ell=1}^{k-1} (\mathbf{w}_\ell^{(c)})_{i^*} (\mathbf{w}_\ell^{(r)})_j, \quad \delta = (\mathbf{w}_k^{(r)})_{j_k^*}, \\ (\mathbf{w}_k^{(c)})_i &= \frac{1}{\delta} \left(A_{i,j_k^*} - \sum_{\ell=1}^{k-1} (\mathbf{w}_\ell^{(c)})_i (\mathbf{w}_\ell^{(r)})_{j_k^*} \right). \end{aligned}$$

A new skeleton is obtained from the remainder,

$$R_k = A - \sum_{\ell=1}^k \mathbf{w}_\ell^{(c)} (\mathbf{w}_\ell^{(r)})^T,$$

without explicitly computing all entries of the matrix R_k .

The required number of rank-one matrices, k , that make up M_k is generally not known a priori. We would like the difference $A - M_k$ to be of small norm. However, we cannot evaluate this difference, because most entries of A are not known. Following [17], we include t randomly chosen matrix entries A_{i_ℓ, j_ℓ} , for $\ell = 1, 2, \dots, t$, with $i_\ell, j_\ell \in \mathbb{N}$. Define for future reference the set

$$\Pi = \{(i_\ell, j_\ell), \forall \ell = 1, 2, \dots, t\}. \quad (2.1)$$

When a new skeleton is determined, the values of these entries are updated by subtraction from the available skeletons,

$$(R_k)_{i_\ell, j_\ell} = (R_{k-1})_{i_\ell, j_\ell} - (\mathbf{w}_k^{(c)})_{i_\ell} (\mathbf{w}_k^{(r)})_{j_\ell}, \quad (2.2)$$

with $(R_0)_{i_\ell, j_\ell} = A_{i_\ell, j_\ell}$. The values $(R_k)_{i_\ell, j_\ell}$ are used in Subsection 3.3 as part of the stopping criterion to determine the final value k^* for k . The value of t is a percentage of the total number of entries. The choice of t should depend on properties of the matrix A ; see [17, 29] for further details. An algorithm is presented in [29].

3. Tikhonov regularization in general form. This section discusses how to combine adaptive cross approximation with Tikhonov regularization in general form.

3.1. Using adaptive cross approximation. The matrix M_k , whose computation was outlined in the previous section, is of the form

$$M_k = W_k^{(c)}(W_k^{(r)})^T, \quad (3.1)$$

where

$$W_k^{(c)} = [\mathbf{w}_1^{(c)}, \mathbf{w}_2^{(c)}, \dots, \mathbf{w}_k^{(c)}] \in \mathbb{R}^{n \times k}, \quad W_k^{(r)} = [\mathbf{w}_1^{(r)}, \mathbf{w}_2^{(r)}, \dots, \mathbf{w}_k^{(r)}] \in \mathbb{R}^{n \times k}.$$

Compute the skinny QR factorizations

$$W_k^{(c)} = Q_k^{(c)} R_k^{(c)}, \quad W_k^{(r)} = Q_k^{(r)} R_k^{(r)}, \quad (3.2)$$

where the matrices $Q_k^{(c)}, Q_k^{(r)} \in \mathbb{R}^{n \times k}$ have orthonormal columns and the matrices $R_k^{(c)}, R_k^{(r)} \in \mathbb{R}^{k \times k}$ are upper triangular. The factorizations (3.2) can be computed by the Householder-QR method or by factorization methods that are designed to perform efficiently on modern computers, such as the methods described in [9, 14, 38].

Combining (3.1) and (3.2) yields

$$M_k = Q_k^{(c)} R_k^{(c)} (R_k^{(r)})^T (Q_k^{(r)})^T. \quad (3.3)$$

Replacing A by M_k in (1.3) gives the minimization problem

$$\min_{\mathbf{x} \in \mathbb{R}^n} \{ \|M_k \mathbf{x} - \mathbf{g}\|^2 + \mu \|L \mathbf{x}\|^2 \}, \quad (3.4)$$

which can be solved in several ways. If the matrix L has a special structure, such as being banded with small bandwidth, then it may be attractive to transform (3.4) to standard form by a technique described by Eldén [12]. Regularization matrices L with a small bandwidth arise, e.g., when L represents a finite difference approximation of a differential operator in one space-dimension. It also is easy to transform (3.4) to standard form when L is an orthogonal projector; see [30].

In the remainder of this section, we discuss the situation when L is such that transformation of (3.4) to standard form as described in [12] is too expensive to be attractive. This is the case, for instance, when L represents a finite difference approximation of a differential operator in two or more space-dimensions. This kind of matrices L will be used in computed examples of Section 4.

We describe several approaches to compute a solution of (3.4) and start with the simplest one. The matrix M_k has a null space of large dimension (at least $n - k$). Therefore the Tikhonov minimization problem (3.4) is not guaranteed to have a unique solution. To remedy this difficulty, we require the solution of (3.4) to live in a subspace of fairly low dimension. A simple solution method is obtained when using the solution subspace $\mathcal{R}(Q_k^{(r)})$. Then we obtain the minimization problem

$$\min_{\mathbf{y} \in \mathbb{R}^k} \{ \|M_k Q_k^{(r)} \mathbf{y} - \mathbf{g}\|^2 + \mu \|L Q_k^{(r)} \mathbf{y}\|^2 \}, \quad (3.5)$$

which has a unique solution if and only if

$$\mathcal{N}(M_k Q_k^{(r)}) \cap \mathcal{N}(L Q_k^{(r)}) = \{\mathbf{0}\}. \quad (3.6)$$

This holds, in particular, when the triangular matrices $R_k^{(c)}$ and $R_k^{(r)}$ in (3.3) are nonsingular. We found (3.6) to hold in all computed examples that we solved.

Introduce the QR factorization

$$LQ_k^{(r)} = Q_k^{(L)}R_k^{(L)}, \quad (3.7)$$

where the matrix $Q_k^{(L)} \in \mathbb{R}^{n \times k}$ has orthonormal columns and $R_k^{(L)} \in \mathbb{R}^{k \times k}$ is upper triangular. We note that since the matrix L typically is very sparse and k is not large, the left-hand side of (3.7) generally can be evaluated quite quickly also when n is large. The minimization problem (3.5) yields the small problem

$$\min_{\mathbf{y} \in \mathbb{R}^k} \{ \|R_k^{(c)}(R_k^{(r)})^T \mathbf{y} - (Q_k^{(c)})^T \mathbf{g}\|^2 + \mu \|R_k^{(L)} \mathbf{y}\|^2 \}. \quad (3.8)$$

This problem can be solved in several ways: We may compute a generalized SVD (GSVD) of the matrix pair $\{R_k^{(c)}(R_k^{(r)})^T, R_k^{(L)}\}$ (see, e.g., [10, 23]), or apply a cheaper reduction of the matrix pair that can be used when the generalized singular values of the matrix pair are not explicitly required; see [11].

When the matrix $R_k^{(L)}$ is nonsingular and not very ill-conditioned, which is the case in many applications, one may consider transforming the minimization problem (3.8) to standard form by the substitution $\mathbf{z} = R_k^{(L)} \mathbf{y}$. This yields the problem

$$\min_{\mathbf{z} \in \mathbb{R}^k} \{ \|R_k^{(c)}(R_k^{(r)})^T (R_k^{(L)})^{-1} \mathbf{z} - (Q_k^{(c)})^T \mathbf{g}\|^2 + \mu \|\mathbf{z}\|^2 \}, \quad (3.9)$$

which easily can be solved, e.g., by computing the singular value decomposition of the matrix $R_k^{(c)}(R_k^{(r)})^T (R_k^{(L)})^{-1}$. The solution \mathbf{z}_μ of (3.9) yields the solution $\mathbf{y}_\mu = (R_k^{(L)})^{-1} \mathbf{z}_\mu$ of (3.5), from which we determine the approximate solution $\tilde{\mathbf{x}}_\mu = Q_k^{(r)} \mathbf{y}_\mu$ of (1.3). The solution of (3.9) is cheaper than the solution of (3.8) with the aid of the GSVD; see [19] for counts of the arithmetic floating point operations necessary to compute the GSVD of a pair of $k \times k$ matrices, and the SVD of a $k \times k$ matrix.

3.2. The discrepancy principle. We turn to the computation of the regularization parameter $\mu > 0$ by the discrepancy principle. Assume for the moment that the matrix A is available. Then we can solve equation (1.8) for $\mu > 0$ by using a zero-finder such as Newton's method or one of the zero-finders described in [8, 34]. The theoretical justification of the discrepancy principle requires that the unavailable error-free vector $\hat{\mathbf{g}}$ associated with the available error-contaminated vector \mathbf{g} satisfies $\hat{\mathbf{g}} \in \mathcal{R}(A)$.

Now consider the application of the discrepancy principle to the determination of the regularization parameter in (3.4). Generally, $\hat{\mathbf{g}} \notin \mathcal{R}(M_k)$ and, therefore, the discrepancy principle cannot be applied when solving (3.4) without modification. In the computed examples, we determine μ so that the computed solution \mathbf{x}_μ of (3.4) satisfies

$$\|M_k \mathbf{x}_\mu - Q_k^{(c)}(Q_k^{(c)})^T \mathbf{g}\| = \eta \delta; \quad (3.10)$$

cf. (1.8). If the matrices $R_k^{(c)}$ and $R_k^{(r)}$ in (3.3) are nonsingular, which generally is the case, then $Q_k^{(c)}(Q_k^{(c)})^T$ is an orthogonal projector onto $\mathcal{R}(M_k)$, and $Q_k^{(c)}(Q_k^{(c)})^T \hat{\mathbf{g}}$ lives in $\mathcal{R}(M_k)$. Equation (3.10) can be solved for $\mu \geq 0$ by using a zero-finder.

Assume for simplicity that the matrices $R_k^{(c)}$ and $R_k^{(r)}$ are nonsingular. Then the minimal-norm solution of the problem

$$\min_{\mathbf{x} \in \mathbb{R}^n} \|M_k \mathbf{x} - Q_k^{(c)}(Q_k^{(c)})^T \hat{\mathbf{g}}\|$$

is given by

$$\widehat{\mathbf{x}}_k = Q_k^{(r)}(R_k^{(c)}(R_k^{(r)})^T)^{-1}(Q_k^{(r)})^T \widehat{\mathbf{g}}.$$

It satisfies

$$\|M_k \widehat{\mathbf{x}}_k - Q_k^{(c)}(Q_k^{(c)})^T \mathbf{g}\| = \|Q_k^{(c)}(Q_k^{(r)})^T \mathbf{e}\| \leq \|\mathbf{e}\| \leq \delta.$$

This is analogous to (1.7). However, the vector $\widehat{\mathbf{x}}_k$ depends on k and may differ from $\widehat{\mathbf{x}}$.

3.3. Stopping criterion for the adaptive cross approximation algorithm.

In view of (1.8), we would like to determine a value of the regularization parameter $\mu > 0$ such that

$$\|A \mathbf{x}_\mu(M_k) - \mathbf{g}\| = \eta \delta. \quad (3.11)$$

Even though the matrix A is not available, we can determine an approximate upper bound for the left-hand side of (3.11) as follows:

$$\begin{aligned} \|A \mathbf{x}_\mu(M_k) - \mathbf{g}\| &= \|(A - M_k) \mathbf{x}_\mu(M_k) + M_k \mathbf{x}_\mu(M_k) - \mathbf{g}\| \\ &\leq \|A - M_k\| \|\mathbf{x}_\mu(M_k)\| + \|M_k \mathbf{x}_\mu(M_k) - \mathbf{g}\| \\ &\lesssim S_k \|\mathbf{x}_\mu(M_k)\| + \|M_k \mathbf{x}_\mu(M_k) - \mathbf{g}\|, \end{aligned} \quad (3.12)$$

where S_k is an approximation of $\|A - M_k\|$. Based on the values of $(R_k)_{i_\ell, j_\ell}$ from (2.2) such an approximation S_k can be computed as

$$\begin{aligned} \|A - M_k\| &\leq \|A - M_k\|_F \\ &\approx \sqrt{\frac{n^2}{|\Pi|} \sum_{(i_\ell, j_\ell) \in \Pi} |(A - M_k)_{i_\ell, j_\ell}|^2} \\ &= \sqrt{\frac{n^2}{|\Pi|} \sum_{(i_\ell, j_\ell) \in \Pi} |(R_k)_{i_\ell, j_\ell}|^2} \\ &=: S_k, \end{aligned} \quad (3.13)$$

where $\|\cdot\|_F$ denotes the matrix Frobenius norm, and the set Π is defined by (2.1) with $|\Pi| = t$ elements.

The number of step $k = k^*$ of the adaptive cross approximation algorithm is chosen to be as small as possible such that there exists a $\mu > 0$ such that

$$S_k \|\mathbf{x}_\mu(M_k)\| = \eta_1 \delta \text{ and } \|M_k \mathbf{x}_\mu(M_k) - \mathbf{g}\| \leq \eta_2 \delta.$$

In this case we have, based on (3.12)

$$\begin{aligned} \|A \mathbf{x}_\mu(M_k) - \mathbf{g}\| &\lesssim S_k \|\mathbf{x}_\mu(M_k)\| + \|M_k \mathbf{x}_\mu(M_k) - \mathbf{g}\| \\ &\lesssim (\eta_1 + \eta_2) \delta. \end{aligned}$$

In the computed examples of the next section, we set $\eta = \eta_1 = \eta_2 = 1.0$. The computed results are not very sensitive to this choice.

4. Numerical experiments. This section describes a few computed examples with the adaptive cross approximation method. For problems in one space-dimension, we will use the regularization matrices $L_0 = I$, where $I \in \mathbb{R}^{n \times n}$ denotes the identity matrix, as well as

$$L_1 = \frac{1}{2} \begin{bmatrix} 1 & -1 & & & 0 \\ & 1 & -1 & & \\ & & 1 & -1 & \\ & & & \ddots & \ddots \\ 0 & & & & 1 & -1 \end{bmatrix} \in \mathbb{R}^{(n-1) \times n} \quad (4.1)$$

or

$$L_2 = \frac{1}{4} \begin{bmatrix} -1 & 2 & -1 & & 0 \\ & -1 & 2 & -1 & \\ & & \ddots & \ddots & \ddots \\ 0 & & & -1 & 2 & -1 \end{bmatrix} \in \mathbb{R}^{(n-2) \times n}, \quad (4.2)$$

which approximate a multiple of the first and second order derivative operators, respectively, assuming that \mathbf{x}_μ is the discretization of a function x_μ at equidistant points on a bounded interval Ω_1 . For problems in two space-dimensions, we use the regularization matrices

$$L_{1,\otimes} = \begin{bmatrix} I & \otimes & L_1 \\ L_1 & \otimes & I \end{bmatrix} \quad (4.3)$$

or

$$L_{2,\otimes} = \begin{bmatrix} I & \otimes & L_2 \\ L_2 & \otimes & I \end{bmatrix}, \quad (4.4)$$

where \otimes stands for the Kronecker product. These choices of regularization matrices are fairly common; see, e.g., [23, 26, 32] for illustrations.

The examples are taken from the Regularization Tools [24] and from IR Tools [18]. In all examples, except for Experiment 6, the number of elements in the set Π is $50n$. The matrix A is of size $n \times n$ with $n \gg 50$ in these experiments. In Experiment 6, the number of elements in the set Π is $4n$.

Experiment 1: We consider the problem “gravity” of size $n = 1024$ from [24]. The aim of this experiment is to illustrate that the quantities S_k defined by (3.13) provide quite accurate approximations of $\|A - M_k\|$. This is displayed by Figure 4.1.

Experiment 2: We again consider the example “gravity” from [24] of size $n = 1024$. Let $\delta = 10^{-2}$. This example illustrates that for certain problems only a fairly small number of steps of the adaptive cross approximation algorithm suffices to yield a satisfactory result. The example also shows that it may be very beneficial to use a regularization matrix different from the identity matrix. The maximum number of adaptive cross approximation steps is 30. Results are shown for $L \in \{L_0, L_1, L_2\}$. The quality of the computed solution $\mathbf{x}_{\text{computed}}$ is measured by the relative error $\|\mathbf{x}_{\text{computed}} - \hat{\mathbf{x}}\|/\|\hat{\mathbf{x}}\|$. The horizontal axis of Figure 4.2 shows the number of steps of adaptive cross approximation; the vertical line indicates that for each one of the choices of L , the stopping criterion for the method is satisfied at step $k^* = 20$.

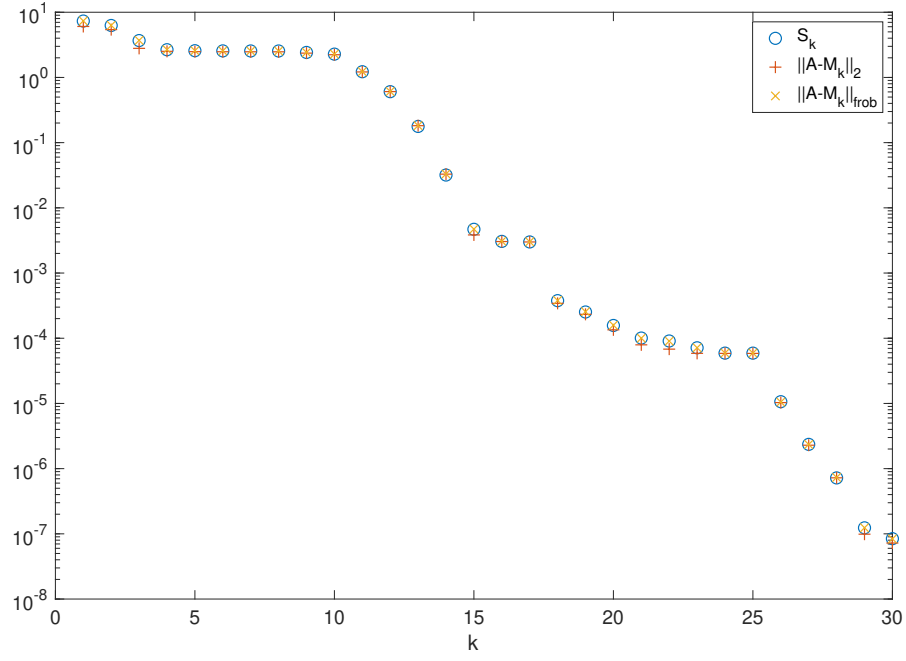


FIG. 4.1. For the “gravity” problem the values of S_k , $\|A - M_k\|$, and $\|A - M_k\|_F$ are plotted as a function of the number of steps k of the adaptive cross approximation algorithm.

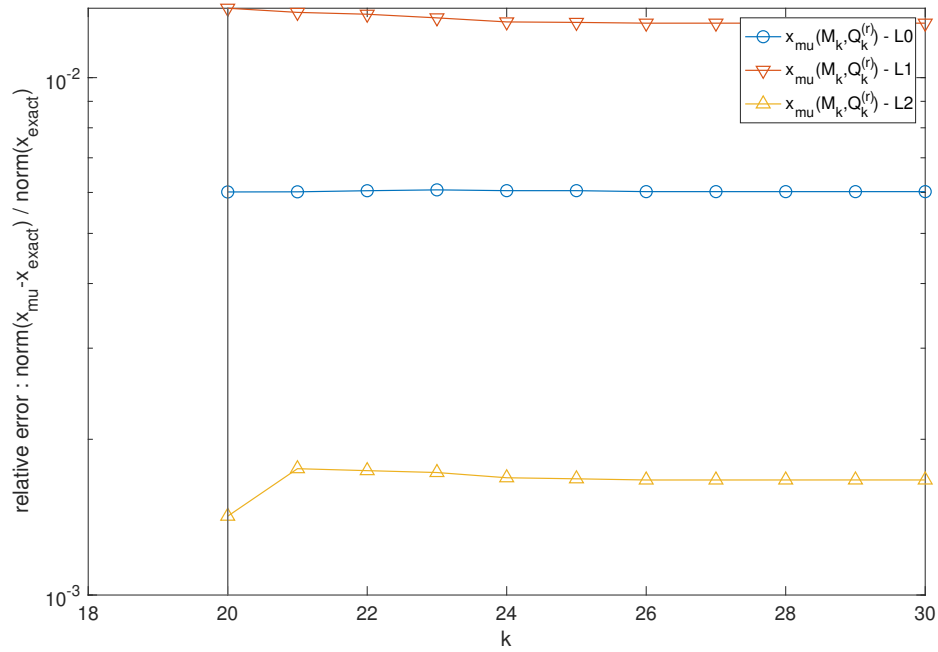


FIG. 4.2. For the “gravity” problem the relative error between the exact solution and the computed approximate solution is plotted as a function of the iteration step k and the chosen regularization matrix L_0, L_1, L_2 . The size of the problem is $n = 1024$ with $\delta = 1.0e - 2$ and $\eta = \eta_1 = \eta_2 = 1.0$. The vertical line indicates where the stopping criterion is satisfied for adaptive rank approximation.

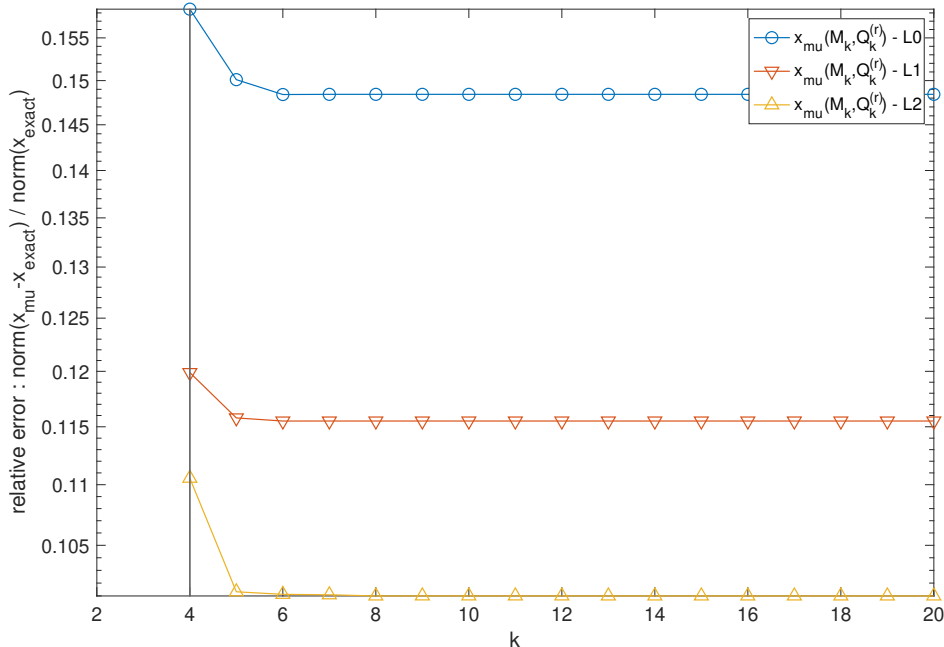


FIG. 4.3. For the “baart” problem the relative error between the exact solution and the computed approximate solution is plotted as a function of the iteration step k and the chosen regularization matrix: L_0, L_1, L_2 . The size of the problem is $n = 1024$ with $\delta = 1.0e - 2$ and $\eta = \eta_1 = \eta_2 = 1.0$. The vertical line indicates where the stopping criterion is satisfied for adaptive rank approximation.

Experiment 3: This experiment is similar to Experiment 2, but for problem “baart” from [24]. Results are displayed in Figure 4.3. Also in this example it is beneficial to use a regularization matrix different from the identity.

Experiment 4: This experiment is similar to Experiment 2, but for problem “phillips” from [24]. The result is shown in Figure 4.4. For this example all three regularization matrices used perform about equally well. The singular values of the matrix A decay to zero slower for this example than for the previous examples. Therefore more steps with the adaptive cross approximation algorithm have to be carried out.

Experiment 5: We consider the example EXdiffusion_rrgmres from IR Tools [18]. The size of the problem is 4096 and $\delta = 5.0e - 3 \|\hat{g}\|$. The other parameters are as in Experiment 2. The “best” solution determined by the example script in IR Tools has relative error (when compared with the exact solution) 0.1875, while for $L = L_2$ our algorithm in iteration step $k = 76$ reaches a relative error of 0.1910. Although RRGMRRES gives a slightly better solution, using a Krylov method such as RRGMRRES requires knowledge of the whole matrix allowing to compute matrix-vector products. Our method only requires knowing part of the elements of the matrix.

Experiment 6: We consider $A = B \otimes B$ with $B \in \mathbb{R}^{40 \times 40}$ the matrix of the “baart” problem in [24]. Let $\hat{z} \in \mathbb{R}^{40}$ be the “exact solution” furnished by the “baart” problem and compute the associated right-hand side $\hat{b} = B\hat{z}$. Then $B\hat{z}\hat{z}^T B^T = \hat{b}\hat{b}^T$. Let $\hat{x} = \text{vec}(\hat{z}\hat{z}^T)$ and $\hat{g} = \text{vec}(\hat{b}\hat{b}^T)$, where vec denotes the vectorization operator. Then $A\hat{x} = \hat{g}$. The singular values of B decay to zero quite quickly with increasing index number. Since the singular values of A are pairwise products of the the singular

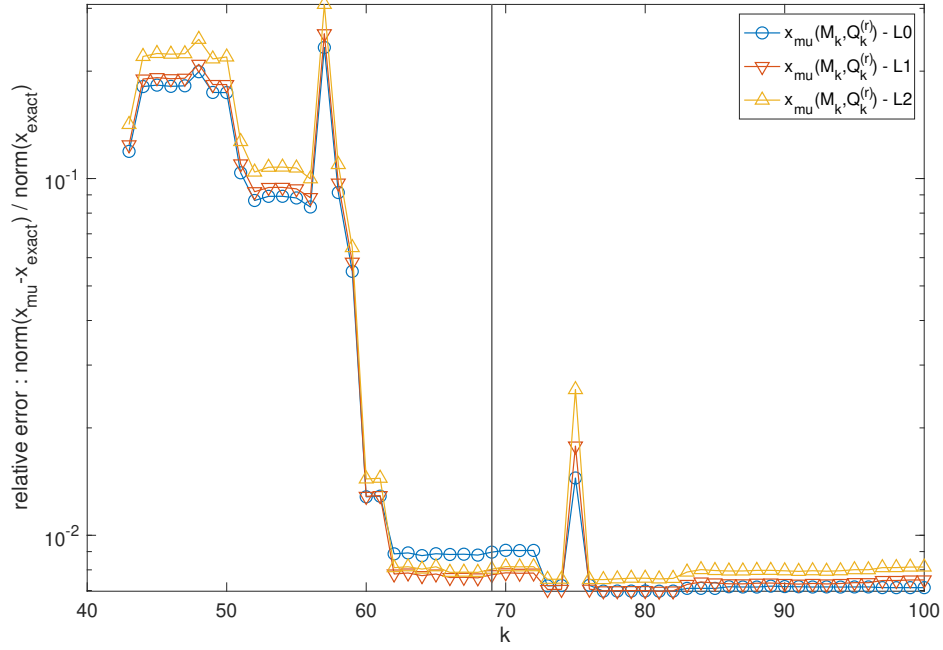


FIG. 4.4. For the “phillips” problem the relative error between the exact solution and the computed approximate solution is plotted as a function of the iteration step k and the chosen regularization matrix L_0, L_1, L_2 . The size of the problem is $n = 1024$ with $\delta = 1.0e-2$ and $\eta = \eta_1 = \eta_2 = 1.0$. The vertical line indicates where the stopping criterion is satisfied for adaptive rank approximation.

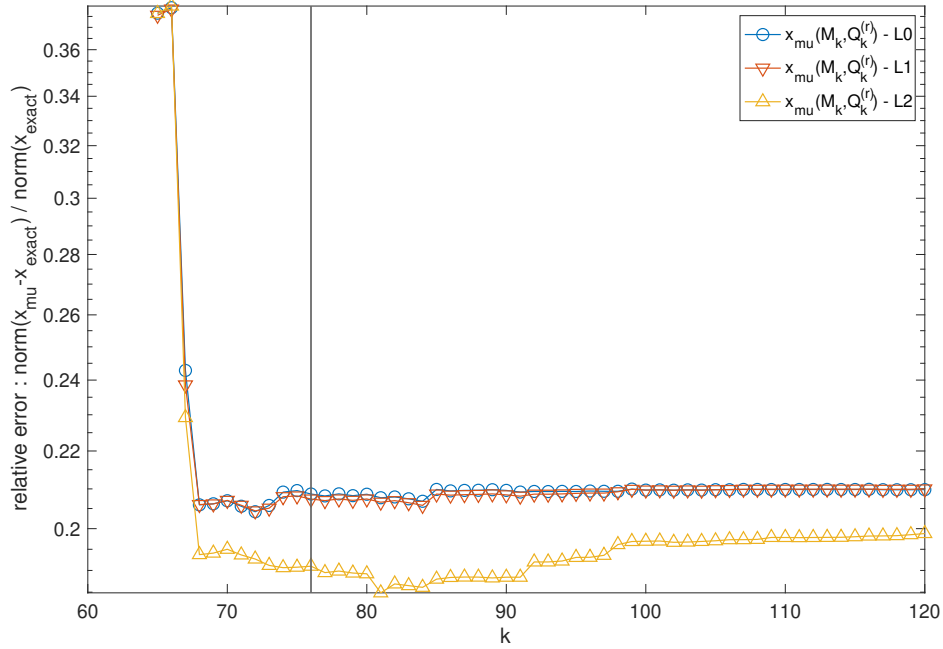


FIG. 4.5. For the “diffusion” problem the relative error between the exact solution and the computed approximate solution is plotted as a function of the iteration step k and the chosen regularization matrix L . The size of the problem is $n = 4096$ with $\delta = 5.0e-3 \|\hat{g}\|$ and $\eta = \eta_1 = \eta_2 = 1.0$. The vertical line indicates where the stopping criterion is satisfied for adaptive rank approximation.

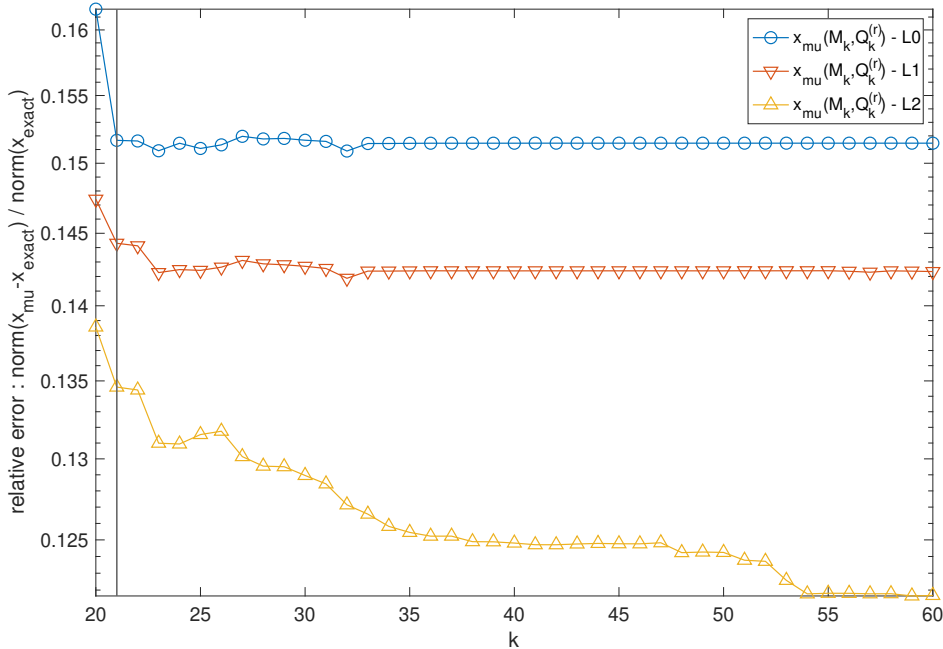


FIG. 4.6. For the 2D “baart” problem the relative error between the exact solution and the computed approximate solution is plotted as a function of the iteration step k and the chosen 2D regularization matrix based on L_0, L_1, L_2 . The size of the problem is $n = 40^2$ with $\delta = 1.0e - 3$ and $\eta = \eta_1 = \eta_2 = 1.0$. The vertical line indicates where the stopping criterion is satisfied for adaptive rank approximation.

values of A , they also decrease to zero quickly with increasing index. Let $\delta = 1e - 2$ and generate the error contaminated right-hand side vector \mathbf{g} associated with $\hat{\mathbf{g}}$. We use (4.4) for L_0, L_1 and L_2 as regularization matrices. The relative errors of the computed solutions are displayed in Figure 4.6.

5. Conclusion. This paper discusses the application of adaptive cross approximation to Tikhonov regularization problems in general form. The computed examples illustrate that often only quite few cross approximation steps are required to yield useful approximate solutions. Particular attention is given to the stopping criterion for adaptive cross approximation.

Acknowledgment. We would like to thank the referee for comments that lead to improvements of the presentation. This research was partially supported by the Fund for Scientific ResearchFlanders (Belgium), Structured Low-Rank Matrix/Tensor Approximation: Numerical Optimization-Based Algorithms and Applications: SeLMA, EOS 30468160, the KU Leuven Research Fund, Numerical Linear Algebra and Polynomial Computations, OT C14/17/073. The research of Thomas Mach has been partially funded by the Deutsche Forschungsgemeinschaft (DFG) - Project-ID 318763901 - SFB1294.

Data availability statement. Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Conflict of interest. The authors declare that they have no conflict of interest.

REFERENCES

- [1] M. L. Baart, *The use of auto-correlation for pseudo-rank determination in noisy ill-conditioned least-squares problems*, IMA J. Numer. Anal., 2 (1982), pp. 241–247.
- [2] M. Bebendorf, *Approximation of boundary element matrices*, Numer. Math., 86 (2000), pp. 565–589.
- [3] M. Bebendorf and R. Grzibovski, *Accelerating Galerkin BEM for linear elasticity using adaptive cross approximation*, Math. Methods Appl. Sci., 29 (2006), pp. 1721–1747.
- [4] M. Bebendorf and S. Rjasanow, *Adaptive low-rank approximation of collocation matrices*, Computing, 70 (2003), pp. 1–24.
- [5] S. Börm, *Efficient Numerical Methods for Non-Local Operators: \mathcal{H}_2 -Matrix Compression, Algorithms and Analysis*, European Math. Society, Zürich, 2010.
- [6] C. Brezinski, G. Rodriguez, and S. Seatzu, *Error estimates for linear systems with applications to regularization*, Numer. Algorithms, 49 (2008), pp. 85–104.
- [7] C. Brezinski, M. Redivo-Zaglia, G. Rodriguez, and S. Seatzu, *Multi-parameter regularization techniques for ill-conditioned linear systems*, Numer. Math., 94 (2003), pp. 203–224.
- [8] A. Buccini, M. Pasha, and L. Reichel, *Generalized singular value decomposition with iterated Tikhonov regularization*, J. Comput. Appl. Math., 373 (2020), Art. 112276
- [9] D. Calvetti, J. Petersen, and L. Reichel, *A parallel implementation of the GMRES algorithm*, in Numerical Linear Algebra, eds. L. Reichel, A. Ruttan, and R. S. Varga, de Gruyter, Berlin, 1993, pp. 31–46.
- [10] L. Dykes, S. Noschese, L. Reichel, *Rescaling the GSVD with application to ill-posed problems*, Numer. Algorithms, 68 (2015), pp. 531–545.
- [11] L. Dykes, L. Reichel, *Simplified GSVD computations for the solution of linear discrete ill-posed problems*, J. Comput. Appl. Math., 255 (2013), pp. 15–27.
- [12] L. Eldén, *A weighted pseudoinverse, generalized singular values, and constrained least squares problems*, BIT Numer. Math., 22 (1982), pp. 487–501.
- [13] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of Inverse Problems*, Kluwer, Dordrecht, 1996.
- [14] J. Erhel, *A parallel GMRES version for general sparse matrices*, Electron. Trans. Numer. Anal., 3 (1995), pp. 160–176.
- [15] C. Fenu, L. Reichel, G. Rodriguez, and H. Sadok, *GCV for Tikhonov regularization by partial SVD*, BIT Numer. Math., 57 (2017), pp. 1019–1039.
- [16] L. Fox and E. T. Goodwin, *The numerical solution of non-singular linear integral equations*, Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci., 245:902 (1953), pp. 501–534.
- [17] K. Frederix and M. Van Barel, *Solving a large dense linear system by adaptive cross approximation*, J. Comput. Appl. Math., 234 (2010), pp. 3181–3195.
- [18] S. Gazzola, P. C. Hansen, and J. G. Nagy, *IR Tools: A MATLAB package of iterative regularization methods and large-scale test problems*, Numer. Algorithms, 81 (2019), pp. 773–811.
- [19] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed., Johns Hopkins University Press, Baltimore, 2013.
- [20] S. A. Goreinov, E. E. Tyrtyshnikov, and N. L. Zamarashkin, *A theory of pseudo-skeleton approximation*, Linear Algebra Appl., 261 (1997), pp. 1–21.
- [21] S. A. Goreinov, N. L. Zamarashkin, and E. E. Tyrtyshnikov, *Pseudo-skeleton approximations by matrices of maximal volume*, Math. Notes, 62 (1997), pp. 515–519.
- [22] C. W. Groetsch, *The Theory of Tikhonov Regularization for Fredholm Equations of the First Kind*, Pitman, Boston, 1984.
- [23] P. C. Hansen, *Rank-Deficient and Discrete Ill-Posed Problems*, SIAM, Philadelphia, 1998.
- [24] P. C. Hansen, *Regularization Tools version 4.0 for Matlab 7.3*, Numer. Algorithms, 46 (2007), pp. 189–194. Software is available in Netlib at <http://www.netlib.org>.
- [25] G. Huang, S. Noschese, and L. Reichel, *Regularization matrices determined by matrix nearness problems*, Linear Algebra Appl., 502 (2016), pp. 41–57.
- [26] G. Huang, L. Reichel, and F. Yin, *On the choice of subspace for large-scale Tikhonov regularization problems in general form*, Numer. Algorithms, 81 (2019), pp. 33–55.
- [27] *H2Lib*, <http://www.h2lib.org/>, 2015–2020.
- [28] S. Kindermann and K. Raik, *A simplified L-curve method as error estimator*, Electron. Trans. Numer. Anal., 53 (2020), pp. 217–238.
- [29] T. Mach, L. Reichel, M. Van Barel, and R. Vandebril, *Adaptive cross approximation for ill-posed problems*, J. Comput. Appl. Math., 303 (2016), pp. 206–217.
- [30] S. Morigi, L. Reichel, and F. Sgallari, *Orthogonal projection regularization operators*, Numer. Algorithms, 44 (2007), pp. 99–114.
- [31] S. Noschese and L. Reichel, *Inverse problems for regularization matrices*, Numer. Algorithms,

- 60 (2012), pp. 531–544.
- [32] Y. Park, L. Reichel, G. Rodriguez, and X. Yu, *Parameter determination for Tikhonov regularization problems in general form*, J. Comput. Appl. Math., 343 (2018), pp. 12–25.
 - [33] L. Reichel and G. Rodriguez, *Old and new parameter choice rules for discrete ill-posed problems*, Numer. Algorithms, 63 (2013), pp. 65–87.
 - [34] L. Reichel and A. Shyshkov, *A new zero-finder for Tikhonov regularization*, BIT Numer. Math., 48 (2008), pp. 627–643.
 - [35] L. Reichel and Q. Ye, *Simple square smoothing regularization operators*, Electron. Trans. Numer. Anal., 33 (2009), pp. 63–83.
 - [36] C. B. Shaw, Jr., *Improvements of the resolution of an instrument by numerical solution of an integral equation*, J. Math. Anal. Appl., 37 (1972), pp. 83–112.
 - [37] E. E. Tyrtyshnikov, *Incomplete cross approximation in the mosaic-skeleton method*, Computing, 64 (2000), pp. 367–380.
 - [38] Y. Yamamoto, Y. Nakatsukasa, Y. Yanagisawa, and T. Fukaya, *Roundoff error analysis of the CholeskyQR2 algorithm*, Electron. Trans. Numer. Anal., 44 (2015), pp. 306–326.