Randomized Algorithms for Large-scale Inverse Problems with General Tikhonov Regularizations

Hua Xiang [*] Jun Zou [†]

Abstract

We shall investigate randomized algorithms for solving large-scale linear inverse problems with general Tikhonov regularizations. Our first approach transforms inverse problems of general form into the ones of standard form, then apply randomized algorithms to reduce large-scale systems of standard form to much smaller-scale systems and seek their regularized solutions in combination with some popular choice rules for regularization parameters. Our second approach involves a new random generalized SVD algorithm that can essentially reduce the sizes of the original large-scale ill-posed systems. The reduced systems can provide approximate regularized solutions with about the same accuracy as the ones by the classical generalized SVD, but they are much more stable and much less expensive as they need only to work on problems of much smaller sizes. Numerical results are presented to demonstrate the efficiency and accuracy of the algorithms.

Keywords. Generalized SVD, randomized algorithm, general Tikhonov regularizations, inverse problems

1 Introduction

Tikhonov regularization is one of the most popular and effective techniques for the ill-conditioned linear system $Ky = b$ arising from the discretization of some linear or nonlinear inverse problems [1, 3, 7], where K is an $m \times n$ matrix and b is an $m \times 1$ vector obtained from measurement data. The standard Tikhonov regularization of this problem is of the form

$$\min_y ||Ky - b||^2 + \mu^2 ||y||^2, \tag{1}$$

where μ is the regularization parameter, and $\|\cdot\|$ is the 2-norm that will be used throughout this work unless otherwise specified. We will call formulation (1) as the standard form, where the second term is for the regularization of the solution and the identity operator is used for the regularization. The identity operator is the simplest and most convenient regularization, but it may not be so desirable in most applications. When we know additional a priori information about the physical solution for a practical problem, we may apply some other more effective

[*] School of Mathematics and Statistics, Wuhan University, Wuhan 430072, P. R. China. The research of this author was supported by the National Natural Science Foundation of China under grants 10901125. (hxiang@whu.edu.cn).

[†] Department of Mathematics, The Chinese University of Hong Kong, Shatin, New Territories, Hong Kong. The work of this author was substantially supported by Hong Kong RGC grants (Projects 405513 and 404611). (zou@math.cuhk.edu.hk).

regularizations. To differentiate from the standard one (1), we will adopt other notation for the general linear ill-conditioned system, namely $Ax = b$, where A is an $m \times n$ matrix and b is an $m \times 1$ vector. Then the associated Tikhonov regularization can be of the following more general form:

$$\min_x ||Ax - b||^2 + \mu^2 ||Lx||^2, \tag{2}$$

where the matrix L is a $p \times n$ matrix, which may be a discrete approximation to some differential operator, for example, the discrete Laplacian or gradient operator. When the null spaces of A and L intersect trivially, i.e., $\mathscr{N}(A) \cap \mathscr{N}(L) = \{0\}$, the regularized Tikhonov solutions of (2) are unique. As we shall see, the general form (2) can be transformed into the standard one (1) for very general regularizations L.

When system (1) or (2) is large-scale, the traditional methods based on singular value decomposition (SVD) or generalized SVD (GSVD) are very expensive and unstable, and often infeasible for practical implementations. For large-scale discrete systems (1) of standard form, we can apply the randomized SVD (RSVD) we developed earlier in [18] to essentially reduce the problem size, then combine the L-curve, generalized cross-validation (GCV), or other heuristic methods to locate reasonable regularization parameters for solving the reduced regularization systems. In this work, we shall focus on the solution of the general form (2) of the Tikhonov regularization. In section 2, we discuss several techniques to transform the general form into the standard one, then use the randomized strategies in [18] to solve the standard system. In section 3, we consider the general form directly, and introduce a new random generalized SVD (RGSVD) to reduce the problem size and then seek the regularized solution. Numerical experiments are presented in section 4.

2 Transformation into standard form and RSVD

In this section we present a general strategy, the first approach in this work, to transform the problem (2) of general form into the standard one (1), then apply the similar strategy as used in [18], which combines the randomized SVD with some choice rules on regularization parameters, to solve the standard system.

2.1 General transformation into the standard system

We demonstrate now how to transform the problem (2) of general form with different regularization operators L into the standard one (1). Usually we assume $\mathscr{N}(A) \cap \mathscr{N}(L) = \{0\}$ so that the solutions of (2) are unique. For the cases where the matrix L is of full column rank, this assumption is automatically satisfied. But we shall consider the most general case without this assumption, so including both cases $\mathscr{N}(A) \cap \mathscr{N}(L) \neq \{0\}$ and $\mathscr{N}(A) \cap \mathscr{N}(L) = \{0\}$. For the case with $\mathscr{N}(A) \cap \mathscr{N}(L) \neq \{0\}$, the solutions of (2) are not unique, so least-squares solutions with minimum norm will be sought. In this work, for any matrix X we shall often use its Moore-Penrose generalized inverse $X^\dagger$ [8], and write $\mathscr{R}(X)$ for its range and $\mathscr{R}^\perp(X)$ for its orthogonal complement.

We start with the following theorem which unifies the transformations of problem (2) into the standard one (1) for all different possibilities, and will discuss in section 2.3 several cases where A and L have special structures and hence the results can be much simplified.

Theorem 1. *Let W and Z be any matrices satisfying*

$$\mathscr{R}(W) = \mathscr{N}(L), \quad \mathscr{R}(Z) = \mathscr{R}(L), \quad Z^T Z = I,$$

and $L^{\#} = (I - W(AW)^{\dagger}A)L^{\dagger}$. Then the least-squares solution with minimum norm to the problem (2) of general form can be given by

$$x_{\mu} = L^{\#} Z y_{\mu} + W(AW)^{\dagger} b, \tag{3}$$

where y_{μ} is the minimizer of the following problem

$$\min_{y}\{||AL^{\#}Zy - b||^2 + \mu^2 ||y||^2\}. \tag{4}$$

Equivalently, y_{μ} can be obtained by solving

$$\min_{y} ||P^T A L^{\dagger} Z y - P^T b||^2 + \mu^2 ||y||^2, \tag{5}$$

where P satisfies $\mathscr{R}(P) = \mathscr{R}^{\perp}(AW)$ and $P^T P = I$.

Proof. Consider the SVD of matrix L:

$$L = U\Sigma V^T = \begin{pmatrix} U_1, U_2 \end{pmatrix} \begin{bmatrix} \Sigma_1 & \\ & 0 \end{bmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix} = U_1 \Sigma_1 V_1^T, \tag{6}$$

where $U = [U_1, U_2]$ and $V = [V_1, V_2]$ are unitary matrices, and Σ_1 contains the nonzero singular values in the diagonal matrix Σ.

For any vector $x \in \mathbb{R}^n$, we write it as $x = V_1 \Sigma_1^{-1} \widehat{y} + V_2 \widehat{z}$, i.e., $x = L^{\dagger} U_1 \widehat{y} + V_2 \widehat{z}$. Since $\mathscr{R}(Z) = \mathscr{R}(L) = \mathscr{R}(U_1)$ and $\mathscr{R}(W) = \mathscr{N}(L) = \mathscr{R}(V_2)$, we can set

$$x = L^{\dagger} Z y + W z.$$

Now we apply the complete orthogonal decomposition on the matrix AW [8]:

$$Q^T (AW) \Pi = \begin{bmatrix} T_{11} & 0 \\ 0 & 0 \end{bmatrix},$$

where Q and Π are orthogonal matrices, and T_{11} is nonsingular matrix with the dimension determined by the rank of AW. For the case where $\mathscr{N}(A) \cap \mathscr{N}(L) = \{0\}$, the matrix AW is of full column rank, and the zero matrix on the right side of T_{11} disappears. For the more general case with $\mathscr{N}(A) \cap \mathscr{N}(L) \neq \{0\}$, the matrix AW is not of full column rank. Let $Q = [Q_1, Q_2]$ be partitioned in compatible dimensions with T_{11}. Then we have $\mathscr{R}(Q_1) = \mathscr{R}(AW)$, and $\mathscr{R}(Q_2) = \mathscr{R}^{\perp}(AW)$ [8]. And it is easy to verify that

$$\begin{aligned} & ||Ax - b||^2 + \mu^2 ||Lx||^2 \\ = \; & ||Q^T A L^{\dagger} Z y + Q^T A W \Pi (\Pi^T z) - Q^T b||^2 + \mu^2 ||L L^{\dagger} Z y||^2 \\ = \; & ||Q_1^T A L^{\dagger} Z y + T_{11} \bar{z}_1 - Q_1^T b||^2 + ||Q_2^T A L^{\dagger} Z y - Q_2^T b||^2 + \mu^2 ||y||^2, \end{aligned}$$

with $\Pi^T z = [\bar{z}_1^T, \bar{z}_2^T]^T$, where $\bar{z}_1$ has the compatible dimension with T_{11}.

Minimizing the quadratic form above, we obtain $\bar{z}_1 = T_{11}^{-1} Q_1^T (b - AL^\dagger Z y_\mu)$, where y_μ is given by the following subproblem of standard form

$$\min_y ||Q_2^T AL^\dagger Zy - Q_2^T b||^2 + \mu^2 ||y||^2,$$

which is in fact the subproblem (5), since we can see from the above proof that Q_2 can be replaced by any orthonormal matrix P satisfying $\mathscr{R}(P) = \mathscr{R}(Q_2) = \mathscr{R}^\perp(AW)$.

Now we can verify that the solution of problem (2) is given by

$$x_\mu = L^\dagger Z y_\mu + W\Pi \begin{bmatrix} \bar{z}_1 \\ \bar{z}_2 \end{bmatrix} \quad \forall\ \bar{z}_2.$$

But by the property of the Moore-Penrose generalized inverse, we know $L^\dagger Z y_\mu \in \mathscr{N}^\perp(L) = \mathscr{R}^\perp(W)$. This implies $||x_\mu||^2 = ||L^\dagger Z y_\mu||^2 + ||[\bar{z}_1^T, \bar{z}_2^T]^T||^2$, hence yields the following solution with minimum norm when $\bar{z}_2 = 0$:

$$\begin{aligned} x_\mu &= L^\dagger Z y_\mu + W\Pi \begin{bmatrix} \bar{z}_1 \\ 0 \end{bmatrix} \\ &= L^\dagger Z y_\mu + W\Pi \begin{bmatrix} T_{11}^{-1} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} (b - AL^\dagger Z y_\mu) \\ &= L^\dagger Z y_\mu + W(AW)^\dagger (b - AL^\dagger Z y_\mu) \\ &= (I - W(AW)^\dagger A) L^\dagger Z y_\mu + W(AW)^\dagger b \\ &= L^{\#} Z y_\mu + W(AW)^\dagger b. \end{aligned}$$

We can see that for the case with $\mathscr{N}(A) \cap \mathscr{N}(L) = \{0\}$, the vector $\bar{z}_2$ disappears automatically, so the above solution is the unique least-squares solution of (2).

We can easily verify that the minimizer of (5) is given by

$$y_\mu = [Z^T (AL^\dagger)^T PP^T (AL^\dagger) Z + \mu^2 I]^{-1} Z^T (AL^\dagger)^T PP^T b.$$

Note that PP^T is the orthogonal projection onto $\mathscr{R}^\perp(AW)$. From the uniqueness of orthogonal projection, we have $PP^T = I - AW(AW)^\dagger$. It is straightforward to check that

$$\begin{aligned} PP^T AL^\dagger &= [I - AW(AW)^\dagger] AL^\dagger = [A - AW(AW)^\dagger A] L^\dagger = AL^{\#}\,, \\ (AL^{\#})^T (AL^{\#}) &= (AL^\dagger)^T PP^T AL^\dagger. \end{aligned}$$

Hence the solution can be rewritten as

$$y_\mu = [Z^T (AL^{\#})^T (AL^{\#}) Z + \mu^2 I]^{-1} Z^T (AL^{\#})^T b,$$

which is obviously the minimizer of (4). □

We know that the columns of matrix Z span the range of L and $Z^T Z = I$, any vector $\widehat{y} \in \mathscr{R}(L)$ can be expressed as $\widehat{y} = Zy$. So the problem (4) is equivalent to

$$\min_{\widehat{y} \in \mathscr{R}(L)} \{||AL^{\#} \widehat{y} - b||^2 + \mu^2 ||\widehat{y}||^2\}\,, \tag{7}$$

whose minimizer is given by

$$\widehat{y}_\mu = Z[Z^T (AL^{\#})^T (AL^{\#}) Z + \mu^2 I]^{-1} Z^T (AL^{\#})^T b = Z y_\mu.$$

2.2 Practical realisation of the transformation

Theorem 1 gives a unified transformation that works for all possible choices of regularization matrix L in (2). We now discuss some practical realisation of the matrices W, Z, P and the oblique pseudoinverse $L^{\#}$ involved in the transformation as stated in Theorem 1. By means of the standard SVD (6) of the matrix L, we can choose $Z = U_1$ and $W = V_2$ such that $\mathscr{R}(Z) = \mathscr{R}(L)$ and $\mathscr{R}(W) = \mathscr{N}(L)$. But the SVD is usually expensive. Instead we may use the complete orthogonal factorization [8] in practical computations when L is not of full rank:

$$U^T L V = \begin{bmatrix} T & 0 \\ 0 & 0 \end{bmatrix},$$

where $U = [u_1, \cdots, u_p]$ and $V = [v_1, \cdots, v_n]$ are orthogonal matrices, and T is a $r \times r$ nonsingular matrix, with $r = rank(L)$. Then we have [8]

$$\mathscr{R}(L) = span\{u_1, \cdots, u_r\}, \quad \mathscr{N}(L) = span\{v_{r+1}, \cdots, v_n\}.$$

So we can choose $W = [v_{r+1}, \cdots, v_n]$ and $Z = [u_1, \cdots, u_r]$. When matrix L is of full rank, the matrices W and Z can be determined by QR, or QR with column pivoting, which are special cases of the complete orthogonal factorization.

For the choice of matrix P, we perform QR with column pivoting on the matrix AW

$$(AW)\Pi = [Q_1, Q_2] \begin{bmatrix} T_1 \\ 0 \end{bmatrix} = Q_1 T_1,$$

where Π is a permutation matrix, T_1 is of full row rank, and $[Q_1, Q_2]$ is an orthogonal matrix. Then we have $\mathscr{R}(Q_1) = \mathscr{R}(AW)$, and $\mathscr{R}(Q_2) = \mathscr{R}^{\perp}(AW)$. So we can choose $P = Q_2$ in Theorem 1. On the other hand, we know from the proof of Theorem 1 that problem (5) is the same as the minimisation (4). So if we choose to solve system (4) instead of (5), then the matrix P is not needed in all computations.

For the oblique pseudoinverse $L^{\#}$, it involves the Moore-Penrose inverse of AW and can be computed as follows:

$$(AW)^{\dagger} = (Q_1 T_1 \Pi^T)^{\dagger} = \Pi (Q_1 T_1)^{\dagger} = \Pi T_1^{\dagger} Q_1^T = \Pi T_1^T (T_1 T_1^T)^{-1} Q_1^T. \tag{8}$$

For the special case with $\mathscr{N}(A) \cap \mathscr{N}(L) = \{0\}$, the matrix AW is of full column rank, and we can use QR factorization. Correspondingly we have $\Pi = I$ and $(AW)^{\dagger} = T_1^{-1} Q_1^T$. For most applications, the dimension of null space $\mathscr{N}(L)$ is very low, for example, $\mathscr{N}(L)$ may be spanned simply by a single vector $[1, 1, \cdots, 1]^T$ or $[1, 2, \cdots, n]^T$. So the matrix AW is very tall skinny, T_1 is a very small matrix, and the cost for computing the Moore-Penrose inverse of AW is negligible.

Now we can summarise the solution to the problem (2) of general form in Algorithm 1. Clearly, if we solve the subproblem (4) in Step 4, then the matrix P is unnecessary to be formed. Moreover, as we shall see in the next section 2.3, some steps of Algorithm 1 can be omitted for matrices L and A of special properties, which are listed below:

1. When L is of full column rank, we have $W = 0$, hence Step 2 can be dropped, and the terms involving W do not appear in Steps 3 and 5. In this case, we have $P = I$, $L^{\#} = L^{\dagger}$, and the subproblems (4) and (5) are exactly the same.

Algorithm 1 Regularized solution by standard form transformation.

```
1. Generate $W$ and $Z$ by matrix decomposition on $L$ such that
   $\mathscr{R}(W) = \mathscr{N}(L)$, $\mathscr{R}(Z) = \mathscr{R}(L)$ and $Z$ is column orthogonal.
2. Form $AW$, generate the column orthogonal matrix $P$ such that $\mathscr{R}(P) = \mathscr{R}^{\perp}(AW)$, and
   compute $(AW)^{\dagger}$, for example, by (8).
3. Compute $L^{\dagger}$, and form the generalized inverse $L^{\#} = (I - W(AW)^{\dagger}A)L^{\dagger}$.
4. Find the minimizer $y_{\mu}$ of subproblem (5) or (4).
5. Form the regularized solution $x_{\mu} = L^{\#}Zy_{\mu} + W(AW)^{\dagger}b$.
```

2. When L is of full row rank, we have $Z = I$, then (4) is reduced to the problem

$$\min_{y}\{||AL^{\#}y - b||^2 + \mu^2||y||^2\}.$$

3. If $\mathscr{N}(L) \subseteq \mathscr{N}(A)$, we have $AW = 0$, hence Step 2 can be dropped, and the terms involving AW disappear in Steps 3 and 5. Since $P = I$ and $L^{\#} = L^{\dagger}$ in this case, the subproblems (4) and (5) are the same.

4. When L is of full row rank and $\mathscr{N}(A) \cap \mathscr{N}(L) = \{0\}$, it is unnecessary to form the pseudo-inverses $(AW)^{\dagger}$ and $L^{\#}$ explicitly, instead we can solve the subproblem (5).

5. If L is rank-deficient and $\mathscr{N}(A) \cap \mathscr{N}(L) = \{0\}$, then AW is of full column rank, and the Moore-Penrose inverse $(AW)^{\dagger}$ can be directly achieved by a QR decomposition.

6. One may use iterative methods to avoid forming the matrix $L^{\#}$ explicitly in Step 3. Instead we need only to have a solver for the linear system $Lg = h$ with given right-hand sides h to achieve $g = L^{\dagger}h$ approximately.

For Step 4, one may apply the randomised SVD to first reduce the system size essentially, then solve the reduced system in combination with some strategies for regularization parameters, as we did in [18]. The standard form transformation described above is an effective and efficient approach for solving the ill-posed problem (2), provided that the operations with L^{-1}, $L^{\dagger}$ or $L^{\#}$ can be efficiently implemented. When L is the discrete Laplacian, the actions of inverses can be done by the algebraic multigrid method efficiently [14]. For the cases where L is of special structures, such as Hankel or Toeplitz, there exist many fast solvers for implementing the operations with L^{-1} or $L^{\dagger}$ [2].

2.3 Special cases

In this subsection, we consider a few important special cases. Although all these cases have the solutions of same form (3) (see Theorem 1), the solutions may be realised very differently as it is shown below.

Case 1: $\mathscr{N}(A) \cap \mathscr{N}(L) \neq \{0\}$**, and** $\mathscr{N}(L) \subseteq \mathscr{N}(A)$. This happens in certain practical problems, for example, the lead-field matrix and the Laplacian have a vector of all ones in their null spaces for the inverse problem from electrocardiography. For this case, $AW = 0$ and $L^{\#} = L^{\dagger}$. The least-squares solution with minimum norm is given by $x_{\mu} = L^{\dagger}Zy_{\mu}$, where y_{μ} solves the minimisation of standard form

$$\min_{y} ||AL^{\dagger}Zy - b||^2 + \mu^2||y||^2. \tag{9}$$

Suppose we write the SVD of matrix L as $L = [Z, U_2] \begin{bmatrix} \Sigma_1 & \\ & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \equiv U\Sigma V^T$, where Σ_1 is the diagonal matrix formed by the nonzero singular values, $U \equiv [Z, U_2]$ and $V \equiv [V_1, V_2]$ are orthogonal matrices. That is, $L = Z\Sigma_1 V_1^T$, $L^\dagger = V_1\Sigma_1^{-1}Z^T$, and $\mathscr{R}(V_2) = \mathscr{N}(L)$. Then using $\mathscr{N}(L) \subseteq \mathscr{N}(A)$, we know $AV = [AV_1, 0]$. Now by direct computing, we can derive

$$\begin{aligned} (A^TA + \mu^2 L^T L)^\dagger &= (VV^TA^TAVV^T + \mu^2 V\Sigma^2 V^T)^\dagger \\ &= V(V^TA^TAV + \mu^2\Sigma^2)^\dagger V^T \\ &= V \begin{bmatrix} (V_1^TA^TAV_1 + \mu^2\Sigma_1^2)^{-1} & 0 \\ 0 & 0 \end{bmatrix} V^T \\ &= V_1(V_1^TA^TAV_1 + \mu^2\Sigma_1^2)^{-1}V_1^T. \end{aligned}$$

But the minimizer y_μ of (9) can be given by

$$\begin{aligned} y_\mu &= [(AL^\dagger Z)^T AL^\dagger Z + \mu^2 I]^{-1}(AL^\dagger Z)^T b \\ &= [(AV_1\Sigma_1^{-1})^T AV_1\Sigma_1^{-1} + \mu^2 I]^{-1}(AV_1\Sigma_1^{-1})^T b \\ &= \Sigma_1[(AV_1)^T AV_1 + \mu^2\Sigma_1^2]^{-1}V_1^TA^Tb. \end{aligned}$$

Therefore, the least-squares solution with minimum norm for this case is expressed by

$$\begin{aligned} x_\mu &= L^\dagger Z y_\mu = V_1\Sigma_1^{-1}y_\mu \\ &= V_1[(AV_1)^TAV_1 + \mu^2\Sigma_1^2]^{-1}V_1^TA^Tb \\ &= (A^TA + \mu^2L^TL)^\dagger A^Tb. \end{aligned}$$

Case 2: L is of full row rank, and $\mathscr{N}(A) \cap \mathscr{N}(L) = \{0\}$. Different from the transformation used in Theorem 1, there is an alternative approach in [6], which applies the following two QR decompositions:

$$L^T = Q\begin{pmatrix} R \\ 0 \end{pmatrix}, \quad Q = [Q_1, W]; \qquad AW = U\begin{pmatrix} T \\ 0 \end{pmatrix}, \quad U = [U_1, P],$$

where Q, U are orthogonal matrices, R and T are nonsingular upper triangular matrices. For this case, the matrix Z can be chosen as the identity. So the three matrices W, P and Z in Theorem 1 are all well defined. Next we shall derive the solution of (2). Note that the range $\mathscr{R}(W) = \mathscr{R}^\perp(L^T) = \mathscr{N}(L)$, hence AW is of full rank, and $\mathscr{R}(P) = \mathscr{R}^\perp(AW)$. Suppose $x = L^\dagger y + Wz$, with $L^\dagger = L^T(LL^T)^{-1} = Q_1R^{-T}$, we can derive that

$$\begin{aligned} & ||Ax - b||^2 + \mu^2||Lx||^2 \\ = \;& ||AL^\dagger y + AWz - b||^2 + \mu^2||LL^\dagger y||^2 \\ = \;& ||U^TAL^\dagger y + U^TAWz - U^Tb||^2 + \mu^2||y||^2 \\ = \;& ||U_1^TAL^\dagger y + Tz - U_1^Tb||^2 + ||P^TAL^\dagger y - P^Tb||^2 + \mu^2||y||^2. \end{aligned}$$

Then we can show the minimisation (2) is equivalent to the following two separated subproblems:

$$\min_y ||P^TAL^\dagger y - P^Tb||^2 + \mu^2||y||^2 \;\text{ and }\; \min_z ||U_1^TAL^\dagger y + Tz - U_1^Tb||.$$

The first subproblem is the same as (5) in Theorem 1 corresponding to the matrix $Z = I$. We can compute $z = T^{-1}U_1^T(b - AL^\dagger y_\mu)$, where y_μ is the minimizer of the first subproblem. Hence,

$$x_\mu = L^\dagger y_\mu + Wz = L^\dagger y_\mu + WT^{-1}U_1^T(b - AL^\dagger y_\mu).$$

Though the matrix $(AW)^\dagger$ does not appear explicitly, this solution is in fact equivalent to (3), by the fact that $(AW)^\dagger = T^{-1}U_1^T$ and $Z = I$. The solution of this case can be rewritten as

$$x_\mu = (I - WT^{-1}U_1^T A)L^\dagger y_\mu + WT^{-1}U_1^T b = L^\# y_\mu + W(AW)^\dagger b.$$

For the QR factorization of large matrices, we may use the recently developed new technique, communication-avoiding QR (CAQR), which invokes tall skinny QR (TSQR) for each block column factorization, to speed up the computation [4, 5].

Case 3: L is of full column rank. Using the skinny QR decomposition $L = Q_1R$, where R is nonsingular and upper triangular, and Q_1 is column orthogonal, we have $||Lx|| = ||Rx||$. Hence the problem (2) of general form is equivalent to the following system

$$\min_x ||AR^{-1}Rx - b||^2 + \mu^2||Rx||^2. \tag{10}$$

Then we can easily transform the system (2) to the standard form (1) by using $y = Rx$ and $K = AR^{-1}$. This is efficient for practical computing since we need only a skinny QR decomposition and a upper triangular solver. The problem (10) is actually the same as the problem (4) or (5) with $y = Q_1Rx$, by noting the facts that $W = 0$, $P = I$, $Z = Q_1$ and $L^\# = L^\dagger = (L^TL)^{-1}L^T = R^{-1}Q_1^T$ in this case.

Case 4: L is a square and nonsingular matrix. As L is nonsingular, we can simply set

$$K = AL^{-1}, \quad x = L^{-1}y, \tag{11}$$

then the problem (2) is rewritten in the standard form (1). The transformation (11) is applicable whenever the actions of L^{-1} can be performed efficiently. This is the case when L is sparse, banded, or of some special structure.

As we have seen from the above cases, by using the transformation that may involve (generalized) matrix inverses, we can transform the problem (2) of general form into the problem (1) of standard form. Then existing methods for the standard form can be applied as we discuss in the next subsection.

2.4 Solution of the standard system (1) by randomized SVD

As we have seen in subsections 2.1-2.3, the regularized solution x_μ of general form (2) can be reduced to the solution to the standard system (1). When the standard system (1) is large-scale, we can first apply randomized SVD algorithm (see Algorithm 2 for $m \leq n$) to reduce it to a much smaller system, then solve it by combining with some existing choice rules for regularization parameters [18]. Similar algorithm can be formulated for $m > n$ [9].

The randomized SVD is much cheaper than the classical SVD. In fact, the flops count of the classical SVD for matrix K is about $4mn^2 + 8n^3$ [8], while the cost of Algorithm 2 is only about $4mnl$ [18]. For the cases where singular values decay rapidly, we can choose $l \ll m$. The ratio of the costs between RSVD and the classical SVD is of the order $O(l/n)$ according to the flops.

Algorithm 2 (RSVD). Given $K \in \mathbb{R}^{m\times n}(m \leq n)$ and $l < m$, compute an approximate rank-l SVD: $K \approx U\Sigma V^T$ with $U \in \mathbb{R}^{m\times l}$, $\Sigma \in \mathbb{R}^{l\times l}$ and $V \in \mathbb{R}^{n\times l}$.

```
1. Generate an $l \times m$ Gaussian random matrix $\Omega$ .
2. Compute the $l \times n$ matrix $Y = \Omega K$.
3. Compute the $n \times l$ orthonormal matrix $Q$ via QR factorization $Y^T = QR$.
4. Form the $m \times l$ matrix $B = KQ$.
5. Compute the SVD of a small matrix $B$: $B = U\Sigma H^T$.
6. Form the $n \times l$ matrix $V = QH$, then $K \approx U\Sigma V^T$.
```

We can see that Algorithm 2 generates an approximate decomposition $K \approx KQQ^T = U\Sigma V^T$, where the columns of Q span approximately the range of K^T, or the right singular vectors. The RSVD in Algorithm 2 was formulated in [18], and can be directly applied for the matrix $K = P^T AL^\dagger Z$ or $K = AL^\# Z$ in the standard system (5) or (4) transformed from the system (2) of general form. The operations with K involve now the operations with L^{-1}, $L^\dagger$, or $L^\#$, which can be implemented efficiently in many applications. For example, when L is the discrete Laplacian the actions of inverses can be done by the algebraic multigrid method efficiently. For the cases where L is of special structures, such as Hankel or Toeplitz, there exist many fast solvers for implementing the operations with L^{-1} or $L^\dagger$ [2].

Suppose that we have an SVD approximation $K \approx U\Sigma V^T$ (by Algorithm 2), where Σ is diagonal with the form $\Sigma = \text{diag}(\sigma_1, \cdots, \sigma_l)$, $U = (u_1, \cdots, u_l)$ and $V = (v_1, \cdots, v_l)$ are orthonormal matrices. Then the approximate Tikhonov regularized solution of (1) can be expressed as

$$x_\mu = \sum_{i=1}^{l} \frac{\sigma_i^2}{\sigma_i^2 + \mu^2} \frac{u_i^T b}{\sigma_i} v_i.$$

The regularization parameter μ can be determined by several existing popular methods, such as L-curve, GCV function, or some other heuristic methods. If we discard the small diagonal elements in Σ, we obtain the truncated SVD (TSVD) of K. With an abuse of notation, we denote the approximate TSVD of K by $K \approx U\Sigma V^T$, where $\Sigma = \text{diag}(\sigma_1, \cdots, \sigma_k)$, either of U and V has k orthonormal columns. Then the approximate TSVD regularized solution x_k is given by

$$x_k = \sum_{i=1}^{k} \frac{u_i^T b}{\sigma_i} v_i.$$

3 Inverse problems of general form and solutions by random generalized SVD

As we have discussed in the last section, the problem (2) of general form can be transformed into the problem of standard form (1), then the classical or randomized SVD method is applied to seek the regularized solution. The classical SVD is usually very expensive, while the randomized SVD method is much cheaper. In this section we shall discuss an alternative strategy for solving the problem (2) of general form by using the generalized SVD (GSVD) of the matrix pair (A, L). But again the classical GSVD are expensive, so we try to reduce the problem size and then seek an approximate solution. We will show that the approximate regularized solution can be achieved by some randomized algorithms.

3.1 Regularized solution with exact GSVD

We consider the problem (2) of general form and the matrix pair (A, L) with $A \in \mathbb{R}^{m\times n}$, $L \in \mathbb{R}^{p\times n}$. We assume that $\mathscr{N}(A) \cap \mathscr{N}(L) = \{0\}$. The classical generalized SVD (CGSVD) is obtained as follows. We first perform a QR factorization for the pair (A, L):

$$\begin{bmatrix} A \\ L \end{bmatrix} = \begin{bmatrix} Q_A \\ Q_L \end{bmatrix} R,$$

where $Q_A \in \mathbb{R}^{m\times n}$, $Q_L \in \mathbb{R}^{p\times n}$, the matrix $[Q_A^T, Q_L^T]^T$ is column orthonormal, and R is an $n \times n$ upper triangular matrix. Then the CS decomposition [8, 17] can be applied to this column orthogonal matrix to yield

$$\begin{bmatrix} Q_A \\ Q_L \end{bmatrix} = \begin{bmatrix} U & 0 \\ 0 & V \end{bmatrix} \begin{bmatrix} C \\ S \end{bmatrix} W^T,$$

where $U \in \mathbb{R}^{m\times m}, V \in \mathbb{R}^{p\times p}, W \in \mathbb{R}^{n\times n}$ are orthogonal matrices, $C \in \mathbb{R}^{m\times n}$ and $S \in \mathbb{R}^{p\times n}$ are diagonal matrices and will be discussed further below. Let $G^{-1} = W^T R$, then we have the classical generalized SVD (CGSVD) of the matrix pair (A, L) as follows [8]:

$$A = UCG^{-1}, \qquad L = VSG^{-1}. \tag{12}$$

The columns of matrices G, U and V are defined by $G = (g_1, \cdots, g_n)$, $U = (u_1, \cdots, u_m)$ and $V = (v_1, \cdots, v_p)$ respectively. Next we shall derive the explicit formulas for the solutions of the regularized system (2) for 4 different cases.

First for the case with $m \geq n$ and $p \geq n$, we have $C = \mathrm{diag}(c_1, \cdots, c_n) \in \mathbb{R}^{m\times n}$, $S = \mathrm{diag}(s_1, \cdots, s_n) \in \mathbb{R}^{p\times n}$, $0 \leq c_1 \leq \cdots \leq c_n$, $s_1 \geq \cdots \geq s_n \geq 0$, and $c_i^2 + s_i^2 = 1$. Using the right singular vectors g_i, and the two sets of left singular vectors u_i and v_i, we can rewrite the CGSVD as

$$Ag_i = c_i u_i, \quad Lg_i = s_i v_i \quad \text{for} \;\; i = 1, 2, \cdots, n.$$

Now using the above CGSVD, we can find the solution of the regularized system (2):

$$\begin{aligned} x_\mu &= (A^T A + \mu^2 L^T L)^{-1} A^T b = G(C^T C + \mu^2 S^T S)^{-1} C^T U^T b \\ &= \sum_{i=1}^{n} \frac{c_i^2}{c_i^2 + \mu^2 s_i^2} \frac{u_i^T b}{c_i} g_i \,. \end{aligned} \tag{13}$$

Next, we consider the case with $m \geq n$ and $p \leq n$. In this case matrices C and S can be expressed in the form [15]:

$$\begin{bmatrix} C \\ S \end{bmatrix} = \begin{bmatrix} \bar{C} & 0 \\ 0 & I \\ 0 & 0 \\ \bar{S} & 0 \end{bmatrix},$$

where $\bar{C} = \mathrm{diag}(c_1, \cdots, c_p)$ and $\bar{S} = \mathrm{diag}(s_1, \cdots, s_p)$ are the $p \times p$ diagonal matrices. Some other submatrices may be empty and their sizes can be determined easily. For this case the generalized singular vectors satisfy the relations:

$$\begin{aligned} &Ag_i = c_i u_i, \qquad && Lg_i = s_i v_i \qquad && \text{for} \;\; i = 1, \cdots, p; \\ &Ag_i = u_i, && Lg_i = 0 && \text{for} \;\; i = p+1, \cdots, n. \end{aligned}$$

Using these relations, we can express the regularized solution of the system (2) as

$$x_\mu = \sum_{i=1}^{p} \frac{c_i^2}{c_i^2+\mu^2 s_i^2}\frac{u_i^T b}{c_i} g_i + \sum_{i=p+1}^{n} (u_i^T b) g_i. \tag{14}$$

For $p < n$, g_i for $i = p+1, \cdots, n$ are the null vectors of L. We can see that these components $\{g_i\}_{i=p+1}^n$ are incorporated into the solution directly without any regularization.

For the case with $m \le n$ and $p \ge n$, we can write [15]

$$\begin{bmatrix} C \\ S \end{bmatrix} = \begin{bmatrix} \bar{C} & 0 \\ \bar{S} & 0 \\ 0 & I \\ 0 & 0 \end{bmatrix},$$

where $\bar{C} = \mathrm{diag}(c_1, \cdots, c_m)$ and $\bar{S} = \mathrm{diag}(s_1, \cdots, s_m)$ are the $m \times m$ diagonal matrices. The submatrices 0 or I above may not exist. Now the regularized solution of system (2) can be expressed by

$$x_\mu = \sum_{i=1}^{m} \frac{c_i^2}{c_i^2+\mu^2 s_i^2}\frac{u_i^T b}{c_i} g_i. \tag{15}$$

The last case is for $m \le n$ and $p \le n$. In this case we have [15]

$$\begin{bmatrix} C \\ S \end{bmatrix} = \begin{bmatrix} \bar{C} & 0 & 0 \\ 0 & I & 0 \\ \bar{S} & 0 & 0 \\ 0 & 0 & I \end{bmatrix},$$

where $\bar{C} = \mathrm{diag}(c_1, \cdots, c_q)$ and $\bar{S} = \mathrm{diag}(s_1, \cdots, s_q)$ are the $q \times q$ diagonal matrices, with $q = m+p-n$. The sizes of the identity matrices and zero submatrices above can be determined accordingly. Then the regularized solution of the system (2) is of the form

$$x_\mu = \sum_{i=1}^{q} \frac{c_i^2}{c_i^2+\mu^2 s_i^2}\frac{u_i^T b}{c_i} g_i + \sum_{i=q+1}^{m} (u_i^T b) g_i. \tag{16}$$

One can easily see that the explicit formulae (13)-(16) for all 4 different cases can be unified in the following formula, with $d = \min\{p,\ n,\ m+p-n,\ m\}$:

$$x_\mu = \sum_{i=1}^{d} \frac{c_i^2}{c_i^2+\mu^2 s_i^2}\frac{u_i^T b}{c_i} g_i + \sum_{i=d+1}^{\min\{m,n\}} (u_i^T b) g_i, \tag{17}$$

and the second term above does not appear when $d = \min\{m, n\}$.

Similarly to TSVD, a truncated version of GSVD can be naturally extended for the problem (2) of the general form. The truncated GSVD (TGSVD) solution reads

$$x_k = \sum_{i=d-k+1}^{d} \frac{u_i^T b}{c_i} g_i + \sum_{i=d+1}^{\min\{m,n\}} (u_i^T b) g_i.$$

We can use the GSVD based Tikhonov regularization to solve (2), but the classical GSVD (CGSVD) above is expensive and impractical for large-scale problems. We shall derive a randomized GSVD algorithm that helps us reduce the large-scale problem size essentially and seek an approximate regularized solution.

3.2 Problem size reduction and approximate solution

For the sake of exposition we only discuss the case with $m \leq n$ in this section. Suppose that the matrix A in (2) has the SVD of the form $A = U\Sigma V^T$, where $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ are unitary matrices, and $\Sigma \in \mathbb{R}^{m\times n}$ is a diagonal matrix with the diagonal elements $\sigma_1 \geq \cdots \geq \sigma_m \geq 0$. We divide the matrix Σ into two parts: $\Sigma = \mathrm{diag}(\Sigma_1, \Sigma_2)$, where $\Sigma_1 \in \mathbb{R}^{r\times r}$ and $\Sigma_2 \in \mathbb{R}^{(m-r)\times(n-r)}$. Correspondingly we partition the matrices U and V as $U = [U_1, U_2]$, $V = [V_1, V_2]$, where $U_1 \in \mathbb{R}^{m\times r}$, $U_2 \in \mathbb{R}^{m\times(m-r)}$, $V_1 \in \mathbb{R}^{n\times r}$, and $V_2 \in \mathbb{R}^{n\times(n-r)}$. Then we can split matrix A into two parts:

$$A = \begin{pmatrix} U_1, U_2 \end{pmatrix} \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix} = U_1\Sigma_1V_1^T + U_2\Sigma_2V_2^T. \tag{18}$$

Suppose that there is a gap among the singular values. The diagonals of Σ_2 correspond to the smaller singular values, while the diagonals of Σ_1 include the larger ones. Then the matrix can be approximated by $A \approx U_1\Sigma_1V_1^T$. Since the singular vectors associated with smaller singular values have more sign changes in their components, we may seek the solution of the form $x_\mu = V_1\bar{x}$ to the system (2) and come to solve the following problem of the reduced size:

$$\min_{\bar{x}} ||AV_1\bar{x} - b||^2 + \mu^2||LV_1\bar{x}||^2. \tag{19}$$

It is equivalent to

$$\min_{\bar{x}} \left\| \begin{bmatrix} A \\ \mu L \end{bmatrix} V_1\bar{x} - \begin{bmatrix} b \\ 0 \end{bmatrix} \right\|.$$

Since $\mathscr{N}(A) \cap \mathscr{N}(L) = \{0\}$, then $\begin{bmatrix} A \\ \mu L \end{bmatrix}$ is of full column rank. As V_1 is column orthogonal, so $\begin{bmatrix} A \\ \mu L \end{bmatrix} V_1$ is also of full column rank. Hence the reduced problem (19) has a unique solution:

$$\bar{x}_{\text{LS}} = \left(V_1^T(A^TA + \mu^2L^TL)V_1\right)^{-1} V_1^TA^Tb. \tag{20}$$

Note that for this approximate regularized solution to (19) we only need to work with the matrix pair (AV_1, LV_1) with the size of $m \times r$ and $p \times r$ respectively, while the original matrix pair (A, L) is of size $m \times n$ and $p \times n$ respectively. We often take $r \ll n$, so the size of the approximate system (19) is essentially smaller than the original one (2). We shall only work on the reduced problem, hence the memory requirement and CPU time can be significantly reduced.

Next, we shall compare the approximate solution (20) to the reduced system (19) with the exact solution to the system (2) of general form. To do so, we represent $\bar{x}_{\text{LS}}$ in terms of the SVD (18) of A. Define $\bar{\Sigma}_2^2 = \Sigma_2^T\Sigma_2 = \mathrm{diag}(\sigma_{r+1}^2, \cdots, \sigma_m^2, 0, \cdots, 0) \in \mathbb{R}^{(n-r)\times(n-r)}$. Now a direct computing yields that

$$\begin{aligned} A^TA + \mu^2L^TL &= V\left(\begin{bmatrix} \Sigma_1^2 & \\ & \bar{\Sigma}_2^2 \end{bmatrix} + \mu^2(LV)^T(LV)\right)V^T \\ &= V\begin{bmatrix} F & B \\ B^T & D \end{bmatrix}V^T, \end{aligned} \tag{21}$$

where F, B and D are given by

$$\begin{aligned} F &= \Sigma_1^2 + \mu^2 (LV_1)^T LV_1 \in \mathbb{R}^{r\times r}, \\ B &= \mu^2 (LV_1)^T LV_2 \in \mathbb{R}^{r\times (n-r)}, \\ D &= \bar{\Sigma}_2^2 + \mu^2 (LV_2)^T LV_2 \in \mathbb{R}^{(n-r)\times(n-r)}. \end{aligned}$$

It is easy to see that F is nonsingular, and we can write the solution of (19) as follows:

$$\begin{aligned} \bar{x}_{\text{LS}} &= \left(V_1^T V \begin{bmatrix} F & B \\ B^T & D \end{bmatrix} V^T V_1 \right)^{-1} V_1^T V \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2^T \end{bmatrix} U^T b \\ &= \left([I, 0] \begin{bmatrix} F & B \\ B^T & D \end{bmatrix} \begin{bmatrix} I \\ 0 \end{bmatrix} \right)^{-1} [I, 0] \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2^T \end{bmatrix} U^T b \\ &= F^{-1} \Sigma_1 U_1^T b. \end{aligned}$$

Then we obtain an approximate regularized solution of (2):

$$x_\mu \approx \bar{x}_\mu \equiv V_1 \bar{x}_{\text{LS}} = V_1 F^{-1} \Sigma_1 U_1^T b. \tag{22}$$

For the special case where $L = I$, the formula (22) is simplified as

$$x_\mu \approx \sum_{i=1}^r \frac{\sigma_i^2}{\sigma_i^2 + \mu^2} \frac{u_i^T b}{\sigma_i} v_i,$$

which is further reduced for $\mu = 0$ as

$$x_\mu \approx \sum_{i=1}^r \frac{u_i^T b}{\sigma_i} v_i.$$

This can be regarded as a regularized solution by truncation.

Next, we further analyze the difference between the approximate solution $\bar{x}_\mu$ in (22) and the original exact solution x_μ. This process will show us how to obtain this approximate solution alternatively. For this purpose, we define the Schur complement $S = D - B^T F^{-1} B$ associated with the governing matrix in (21). It is easy to check that

$$y^T S y = y^T \bar{\Sigma}_2^2 y + \mu^2 (LV_2 y)^T [I - \mu^2 LV_1 F^{-1} (LV_1)^T] LV_2 y.$$

Since μ is a small parameter, it is reasonable to assume that $y^T S y > 0$ for any $y \neq 0$, hence S is nonsingular. We can verify that

$$\begin{bmatrix} F & B \\ B^T & D \end{bmatrix}^{-1} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & S^{-1} \end{bmatrix},$$

where $M_{11} = F^{-1} + F^{-1} B S^{-1} B^T F^{-1}$, and $M_{21} = M_{12}^T = -S^{-1} B^T F^{-1}$. We can write the solution of (2) as

$$\begin{aligned} x_\mu &= (A^T A + \mu^2 L^T L)^{-1} A^T b \\ &= [V_1, V_2] \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & S^{-1} \end{bmatrix} \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2^T \end{bmatrix} \begin{bmatrix} U_1^T \\ U_2^T \end{bmatrix} b \\ &= V_1 M_{11} \Sigma_1 U_1^T b + V_2 M_{21} \Sigma_1 U_1^T b + (V_1 M_{12} + V_2 S^{-1}) \Sigma_2^T U_2^T b \\ &= V_1 (I + F^{-1} B S^{-1} B^T) F^{-1} \Sigma_1 U_1^T b - V_2 S^{-1} B^T F^{-1} \Sigma_1 U_1^T b \\ &\quad + (V_2 - V_1 F^{-1} B) S^{-1} \Sigma_2^T U_2^T b. \end{aligned} \tag{23}$$

We shall show how to achieve the approximate regularized solution (22) from the exact solution (23) by truncation. Suppose that there exists a gap between Σ_1 and Σ_2. And we further assume that $\Sigma_1 = O(1)$, and $\Sigma_2 = O(\epsilon)$ or $o(\epsilon)$. Then the last term in (23) may be ignored, leading to

$$x_\mu \approx V_1(I + F^{-1}BS^{-1}B^T)F^{-1}\Sigma_1 U_1^T b - V_2 S^{-1}B^T F^{-1}\Sigma_1 U_1^T b.$$

Note that V_2 is associated with the smallest singular values and its columns are highly oscillatory and have more frequent sign changes. If we drop the second term involving V_2 in (23) like we do with TSVD, we then derive

$$x_\mu \approx V_1(I + F^{-1}BS^{-1}B^T)F^{-1}\Sigma_1 U_1^T b.$$

The expected regularization parameters are usually small, and it is reasonable to assume that $\mu = O(\epsilon)$. Then we can see that $B = O(\epsilon^2)$, $F^{-1} = O(1)$, $S = O(\epsilon^2)$, and $F^{-1}BS^{-1}B^T = O(\epsilon^2)$. If we further omit the higher order terms in the above expression, then we obtain the following approximate solution

$$x_\mu \approx V_1 F^{-1}\Sigma_1 U_1^T b, \tag{24}$$

which is exactly the approximate solution $\bar{x}_\mu$ in (22) that we will use in the following randomized algorithm. So we can see that the approximate solution $\bar{x}_\mu$ can be achieved by dropping the higher order terms and those oscillatory terms.

We end this subsection with some remarks on a few important cases where we may further simplify the representation of solutions.

Case 1: $L = I$. For this simplest case, we can check that $B = 0$, $F = \Sigma_1^2 + \mu^2 I$, and $S = D = \bar{\Sigma}_2^2 + \mu^2 I$. Then the exact solution (23) of (2) can be expressed by

$$x_\mu = V_1 F^{-1}\Sigma_1 U_1^T b + V_2 S^{-1}\Sigma_2^T U_2^T b,$$

which is the same as $(A^T A + \mu^2 I)^{-1}A^T b$ by using the SVD (18) of A. If we chop off the second oscillation term, we have $x_\mu \approx V_1 F^{-1}\Sigma_1 U_1^T b$ in (24).

Case 2: $\mathscr{R}(V_1) = \mathscr{R}(A^T)$. In this case we know Σ_2 is simply a zero matrix, and the last term in (23) vanishes. Then the exact solution (23) of (2) reduces to

$$x_\mu = V_1 M_{11}\Sigma_1 U_1^T b + V_2 M_{21}\Sigma_1 U_1^T b.$$

Note that the second term above lies in $\mathscr{N}(A)$ and is perpendicular to the first term. Hence the least-squares solution with minimum norm is given by

$$x_\mu = V_1 M_{11}\Sigma_1 U_1^T b.$$

As we have seen before, this term can be further approximated as $x_\mu \approx V_1 F^{-1}\Sigma_1 U_1^T b$.

Case 3: $\mathscr{R}(V_2) \subseteq \mathscr{N}(L) \subseteq \mathscr{N}(A)$. For this case, we can see that all the matrices B, Σ_2, S and D vanish. Then the exact solution (23) of (2) is simplified to

$$x_\mu = V_1 F^{-1}\Sigma_1 U_1^T b,$$

which is exactly the same as $(A^T A + \mu^2 L^T L)^\dagger A^T b$.

3.3 Regularized solution by random GSVD

As discussed in section 3.2, we need to obtain a good approximation V_1 required in the system (19), in order to reduce the size of the problem (2) of general form and find its approximate solution. If we directly perform the SVD of A and choose V_1 from its right singular vectors, it will be very expensive and impractical for large-scale systems. We now seek an economic way to obtain a good approximation V_1. Suppose that we have an approximate SVD of A, that is, $A \approx U\Sigma V^T$, where $U \in \mathbb{R}^{m\times l}$ and $V \in \mathbb{R}^{n\times l}$ are orthonormal. There is an abuse of notations U and V here in GSVD of (A, L), but they can be differentiated from the context. The approximate SVD of A can be achieved by RSVD, i.e., Algorithm 2 with K replaced by A. We write this approximation as $A \approx U\Sigma\tilde{V}_1^T$, then we can seek the solution of the form $x = \tilde{V}_1\widehat{x}$, where the $n \times l$ column orthogonal matrix $\tilde{V}_1$ forms the approximate right singular vectors of A. Now the transformed problem (19) reads as follows:

$$\min_{\widehat{x}} ||A\tilde{V}_1\widehat{x} - b||^2 + \mu^2||L\tilde{V}_1\widehat{x}||^2. \tag{25}$$

Define $\widehat{A} = A\tilde{V}_1$ and $\widehat{L} = L\tilde{V}_1$. This problem still has the general form

$$\min_{\widehat{x}} ||\widehat{A}\widehat{x} - b||^2 + \mu^2||\widehat{L}\widehat{x}||^2, \tag{26}$$

where $\widehat{A} \in \mathbb{R}^{m\times l}$ and $\widehat{L} \in \mathbb{R}^{p\times l}$. Since $\mathscr{N}(A) \cap \mathscr{N}(L) = \{0\}$ and $\tilde{V}_1$ is column orthogonal, then $[\widehat{A}^T, \mu\widehat{L}^T]^T$ is of full column rank. Hence the reduced problem (26) has a unique solution. But the matrix pair $(\widehat{A}, \widehat{L})$ is of much smaller size compared with the original matrix pair (A, L), and we can easily apply the classical GSVD (CGSVD) to this matrix pair. The solution procedure is summarized in Algorithm 3.

Algorithm 3 Approximate regularized solution by RGSVD.

```
1. Use RSVD, i.e., Algorithm 2 with $K$ replaced by $A$, to obtain
     $A \approx U\Sigma\tilde{V}_1^T$ with $U \in \mathbb{R}^{m\times l}$, $\Sigma \in \mathbb{R}^{l\times l}$ and $\tilde{V}_1 \in \mathbb{R}^{n\times l}$.
2. Apply CGSVD to the matrix stencil $(A\tilde{V}_1, L\tilde{V}_1)$ of smaller size.
3. Solve the Tikhonov regularization problem (25) and obtain the minimizer $\widehat{x}_{\text{LS}}$.
4. Achieve the approximate regularized solution $\widetilde{x}_\mu = \tilde{V}_1\widehat{x}_{\text{LS}}$.
```

Suppose that the CGSVD of the matrix pair $(\widehat{A}, \widehat{L})$ has the similar form as (12),

$$\widehat{A} = \widehat{U}\widehat{C}\widehat{G}^{-1}, \qquad \widehat{L} = \widehat{V}\widehat{S}\widehat{G}^{-1},$$

where $\widehat{U} \in \mathbb{R}^{m\times m}$, $\widehat{V} \in \mathbb{R}^{p\times p}$ are orthogonal matrices, $\widehat{C} \in \mathbb{R}^{m\times l}$, $\widehat{S} \in \mathbb{R}^{p\times l}$ are diagonal matrices, and $\widehat{G} \in \mathbb{R}^{l\times l}$ is nonsingular. Then the solution $\widehat{x}$ of (26) can be expressed by

$$\widehat{x}_{\text{LS}} = \widehat{G}(\widehat{C}^T\widehat{C} + \mu^2\widehat{S}^T\widehat{S})^{-1}\widehat{C}^T\widehat{U}^Tb.$$

And this gives us an approximate solution of the original problem (2):

$$x_\mu \approx \widetilde{x}_\mu \equiv \tilde{V}_1\widehat{x}_{\text{LS}} = \tilde{V}_1\widehat{G}(\widehat{C}^T\widehat{C} + \mu^2\widehat{S}^T\widehat{S})^{-1}\widehat{C}^T\widehat{U}^Tb. \tag{27}$$

In the following we will demonstrate that this approximation can be regarded as the least-squares solution with minimum norm of a nearby problem. To this aim, we define

$$\begin{bmatrix} \widetilde{A} \\ \widetilde{L} \end{bmatrix} := \begin{bmatrix} A \\ L \end{bmatrix} \tilde{V}_1\tilde{V}_1^T = \begin{bmatrix} \widehat{A} \\ \widehat{L} \end{bmatrix} \tilde{V}_1^T = \begin{bmatrix} \widehat{U}\widehat{C}\widehat{G}^{-1}\tilde{V}_1^T \\ \widehat{V}\widehat{S}\widehat{G}^{-1}\tilde{V}_1^T \end{bmatrix}. \tag{28}$$

Lemma 1. *The approximate solution* (27) *is the solution to the following problem with minimum norm:*

$$\min_x ||\widetilde{A}x - b||^2 + \mu^2 ||\widetilde{L}x||^2 . \tag{29}$$

Proof. We can check that $\mathscr{N}(\widetilde{L}) = \mathscr{N}(\widetilde{A}) = \mathscr{R}\{\tilde{V}_1\}^{\perp}$, which is included in Case 1 in section 2.3, from where we know minimisation (29) has the least-squares solution with minimum norm:

$$\widetilde{x}_\mu = (\widetilde{A}^T\widetilde{A} + \mu^2 \widetilde{L}^T\widetilde{L})^{\dagger}\widetilde{A}^T b.$$

Using the decomposition (28) and the property of Moore-Penrose inverse, this regularized solution can be further expressed as

$$\begin{aligned} \widetilde{x}_\mu &= [\tilde{V}_1 \widehat{G}^{-T}(\widehat{C}^T\widehat{C} + \mu^2\widehat{S}^T\widehat{S})\widehat{G}^{-1}\tilde{V}_1^T]^{\dagger}\widetilde{A}^T b \\ &= \tilde{V}_1\widehat{G}(\widehat{C}^T\widehat{C} + \mu^2\widehat{S}^T\widehat{S})^{-1}\widehat{G}^T\tilde{V}_1^T\widetilde{A}^T b \\ &= \tilde{V}_1\widehat{G}(\widehat{C}^T\widehat{C} + \mu^2\widehat{S}^T\widehat{S})^{-1}\widehat{C}^T\widehat{U}^T b . \end{aligned}$$

This is exactly the solution (27). □

In summary, for the least-squares solution of general form (2), we can approximate it by solving the problem (26) and set $x = \tilde{V}_1\widehat{x}$, which is equivalent to the least-squares solution of (29) with minimum norm. Note that the size of the matrix pair $(\widehat{A}, \widehat{L})$ is much smaller than the matrix pair (A, L), and we need to work only on the matrix pair $(\widehat{A}, \widehat{L})$ in practice. Using Algorithm 3, we can obtain an approximate solution (22) based on the GSVD on the matrix pair $(\widehat{A}, \widehat{L})$ with $\widehat{A} \in \mathbb{R}^{m\times l}$ and $\widehat{L} \in \mathbb{R}^{p\times l}$, while the original problem of general form has the regularized solution (13) when CGSVD is applied directly on the matrix pair (A, L). Furthermore, we know the approximate solution is spanned by the columns of $\tilde{V}_1$. So if we have some a priori information about the solution, we may incorporate it into $\tilde{V}_1$ by orthogonalization with its columns.

Similarly to the truncated SVD (TSVD), we can use the truncated GSVD (TGSVD) to seek the regularized solution [11]. That is, after using Algorithm 3, we chop off those smallest singular values to achieve a truncated version of RGSVD. Then we use the L-curve, GCV, or some other heuristic methods to determine the regularization parameter. This procedure is similar to Tikhonov regularization, and we do not go into further details about this.

4 Numerical experiments

For the system (1) of standard form, we studied RSVD and tested various linear inverse problems in [18]. In this section, we shall mainly focus on the system (2) of general form directly and test the newly proposed Algorithm 3 with examples from different linear inverse problems to illustrate the performance of the algorithm. As we shall see, we can essentially reduce the problem size by using RGSVD, but still obtain reasonably good approximate regularized solutions. The following tests are done using Matlab R2012a in a laptop with Intel(R) Core(TM) i5 CPU M480 @2.67G.

In this section we will test some examples from Regularization Tools [10]. Most of the cases are related to the Fredholm integral equation of the first kind,

$$\int_a^b K(s,t)x(t)dt = g(s), \qquad c \le s \le d, \tag{30}$$

where K is the square integrable kernel function. In the tests the kernel K and the solution x are given and discretized to yield the matrix A and the vector x, then the discrete right-hand side is determined by $b = Ax$. Matrix L in the general form (2) is the discrete approximation to the first or second order differential operator. In particular we will try the following choices:

$$L = \begin{pmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1 \end{pmatrix} \quad \text{and} \quad L = \begin{pmatrix} 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2 & 1 \end{pmatrix}.$$

We will write them by $L = bidiag(1, -1) \in \mathbb{R}^{(n-1)\times n}$ and $L = tridiag(1, -2, 1) \in \mathbb{R}^{(n-2)\times n}$ respectively.

In all our numerical experiments, the observation data b^δ is generated from the exact data b by adding the noise in the form

$$b^\delta = b + \delta \frac{||b||}{||s||} s = b + \varepsilon \frac{s}{||s||},$$

where s is a random vector, $s = randn(n, 1)$ if not specified otherwise, $\varepsilon = \delta ||b||$ is the so-called noise level, and δ is the relative noise level [12]. In our numerical tests, we choose the relative noise level δ =1e-4 as in [13].

In the following tests, we shall compare the performance of 5 different algorithms for solving the general problem (2), including RGSVD (the random GSVD, i.e., Algorithm 3), the classical GSVD (CGSVD), two methods 'std+CSVD' and 'std+RSVD', which transform first the general problem (2) into the standard form, then apply CSVD and RSVD respectively to achieve the regularized solution, and the method which uses CSVD directly for the problem (2) but with a trivial regularization operator $L = I$. In all our numerical tests, the sampling size is set to $l = 50$ and the regularisation operator to $L = tridiag(1, -2, 1)$ if not specified otherwise. In all subsequent tables we shall use the following notation: μ stands for the regularization parameter, err for the relative error $||x_\mu - x||_2/||x||_2$ of the computed Tikhonov regularized solution x_μ to the exact solution x, and n for the problem size, and T for the computational time (in seconds). To fairly compare the efficiency and accuracy of 5 different methods, we shall fix the regularization parameter μ for the results in Tables 2 and 3. That is, we shall solve the general problem (2) with the same parameter μ. Thus each computational time T in Tables 2 and 3 will include the time for the (approximate) matrix factorisation and the computing of the Tikhonov regularized solution, but not the time for determining the regularization parameter. However, for a more complete comparison among all the methods, we shall report also the total computational time required for computing a Tikhonov regularized solution, namely including also the time for determining the regularization parameter, by the parameter choice rule GCV function, and the numerical results are shown in Table 1.

Example 1 (Shaw). This is a one-dimensional model of an image reconstruction problem. It arises from discretization of the integral equation (30) with the kernel K being the point spread function for an infinitely long slit:

$$K(s,t) = (\cos s + \cos t)^2 \left(\sin u/u\right)^2, \quad u = \pi(\sin s + \sin t).$$

The exact solution is given by $x(t) = a_1 \exp(-c_1(t - t_1)^2) + a_2 \exp(-c_2(t - t_2)^2)$, where the parameters a_1, a_2, c_1, c_2, t_1 and t_2 are constants chosen to give two different humps [10].

	std + CSVD			std + RSVD			CGSVD			RGSVD			CSVD($L = I$)		
n	μ	$T(s)$	err	μ	$T(s)$	err	μ	$T(s)$	err	μ	$T(s)$	err	μ	$T(s)$	err
500	8.26E-02	0.34	1.40E-02	8.26E-02	0.09	1.40E-02	8.26E-02	0.41	1.40E-02	8.26E-02	0.05	1.40E-02	3.14E-04	0.41	2.66E-02
1000	1.52E-01	2.24	1.75E-02	1.52E-01	0.27	1.75E-02	1.52E-01	2.41	1.75E-02	1.52E-01	0.08	1.75E-02	1.00E-04	1.31	2.96E-02
2000	8.52E-01	18.24	1.20E-02	8.52E-01	1.03	1.20E-02	8.52E-01	19.30	1.20E-02	8.52E-01	0.18	1.20E-02	1.47E-04	9.21	2.08E-02

Table 1: Example SHAW. Comparison of the total computational times and solution accuracies.

The computational times, the regularization parameters and the relative errors of the computed solutions are listed in Table 1. The regularization parameters are determined by GCV functions in combination with 4 methods, i.e., std+CSVD, std+RSVD, CGSVD and RGSVD, and they are almost the same, so leading to about the same accuracy for all these methods; see Figure 1 and the errors listed in Table 1. But the randomized algorithms, namely std+RSVD and RGSVD, are much faster than the classical methods std+CSVD and CGSVD, as the problems they need to solve are much smaller than those solved by the classical methods. As we can see from Table 1, for the problem of size $n = 2000$, the new method RGSVD is about 100 times faster than the classical method CGSVD. And such outperformance of RGSVD becomes more significant when the problem size increases. Also from the relative errors shown in Table 1, we can see that the regularization with $L = tridiag(1, -2, 1)$ is more accurate than the identity regularisation $L = I$.

Similar observations above apply also to the subsequent testing examples FOXGOOD, GRAVITY, HEAT, PHILLIPS from [10], so will not be mentioned any more below.

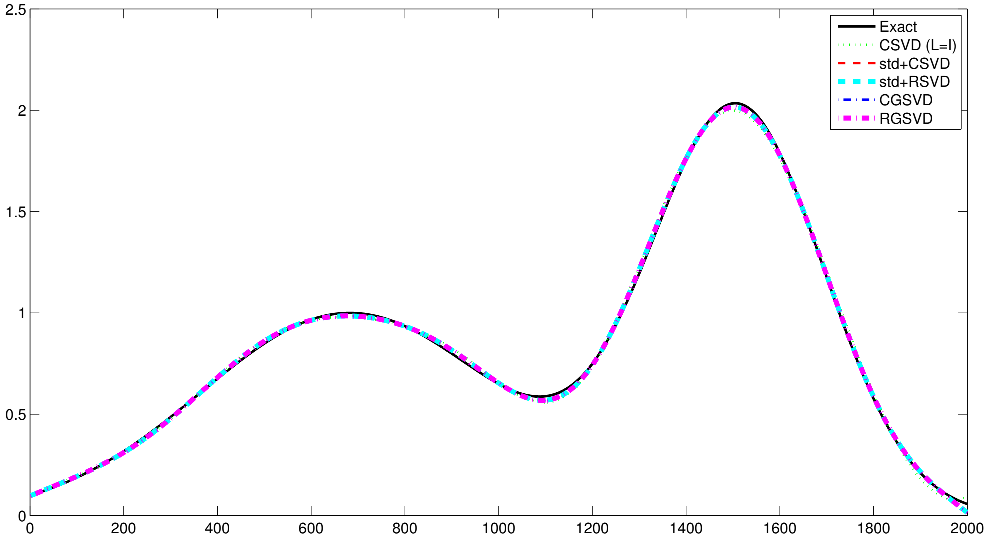

Figure 1: Example SHAW of size $n = 2000$.

Example 2 (I_LAPLACE). This test problem is the inverse Laplace transformation, a Fredholm first kind integral equation, discretized by Gauss-Laguerre quadrature. The kernel K is given by $K(s, t) = exp(-st)$.

Regularization Tools [10] provides the test problem I_LAPLACE(n, e_g), with n being the matrix size, and $e_g = 1, 2, 3$ or 4 corresponding to 4 different examples ($e_g = 1$ by default). The test problem I_LAPLACE(n, 1) has the exact solution $x(t) = exp(-t/2)$, and I_LAPLACE(n, 3) gives $x(t) = t^2 exp(-t/2)$. For these two cases, the solutions are very smooth. The method RGSVD works very well with regularization $L = bidiag(1, -1)$ and small sample size $l = 50$. Its accuracy is also comparable or even better than that of the classical method CGSVD, but the CPU times and memories are significantly smaller. We do not report the numerical results for these two relatively simple cases, but focus on the other two cases, $e_g = 2$, 4, which are more difficult due to the sudden change and strong discontinuity in the solutions. For these two cases, the regularization in general form (2) is necessary to ensure a meaningful numerical solution.

First for the test problem I_LAPLACE(n, 2), the exact solution is $x(t) = 1 - exp(-t/2)$, which has a horizontal asymptote. For this problem, the identity regularization $L = I$ can not give a good reconstruction [13] (see Figure 2), and the regularization $L = tridiag(1, -2, 1)$ does not work well either. Instead, the regularization $L = bidiag(1, -1)$ is very effective to capture the rapid change in the solution.

Let e be the vector of all ones, i.e., $e = ones(n, 1)$. Clearly e is the basis vector of the null space of operator $L = bidiag(1, -1)$. This suggests us to incorporate a constant mode into the matrix W in Algorithm 3. Suppose we have the approximate SVD of the matrix A by randomized algorithm. That is, $A \approx U\Sigma W^T$. Let $w = (I - WW^T)e$, then we enlarge the matrix W by adding one more column vector, namely $w/||w||$. This is equivalent to finding the orthogonal projection onto $\mathscr{R}(W)^\perp$, then adding it to matrix W as a new column.

For most of our cases, the sample size can be as small as 50. But for this difficult case, we need to use larger sample sizes for RGSVD: for the problems of sizes $n = 500$, 1000, 2000, we choose the sample sizes $l = 150$, 300, 600 respectively. However, the method std+RSVD still works very well with small sample sizes ($l = 50$), and is hence much faster in terms of computational times. Even with larger sample size, the computational time of RGSVD is still much less than the classical methods, and the approximate solution is still quite accurate; see Figure 2 and Table 2 for more details. In Table 2, the regularization parameter μ is chosen to be the same for all 4 methods and $L \neq I$. For $L = I$, the parameter is chosen by GCV, but for a fair comparison the computational time T includes also only the time for matrix factorization and the computing of the regularized solution, without the time for choosing the regularization parameter.

We have observed from all the examples we have tested, our method working on a much smaller-sized problem is often more robust and stable than the classical method. This is also clearly observed in our test with I_LAPLACE(n, 4) from Regularization Tools [10]. The exact solution to the problem has a big jump:

$$x(t) = \begin{cases} 0, & t \leq 2; \\ 1, & t > 2. \end{cases}$$

We still choose the regularization $L = bidiag(1, -1)$, and other parameters are the same as for I_LAPLACE(n, 2). Our tests show that our randomized algorithms can always provide the regularized solutions with desired accuracy but require much less computational times; see more details in Figure 3 and Table 2.

As our last set of numerical tests, we consider 6 different inverse problems taken from [10], namely the problems FOXGOOD, GRAVITY, HEAT, I_LAPLACE, PHILLIPS, SHAW as listed there. All the problems are solved with the same regularisation parameter μ and the numerical results are shown in Table 3. One can easily see from the table that RGSVD is the most efficient

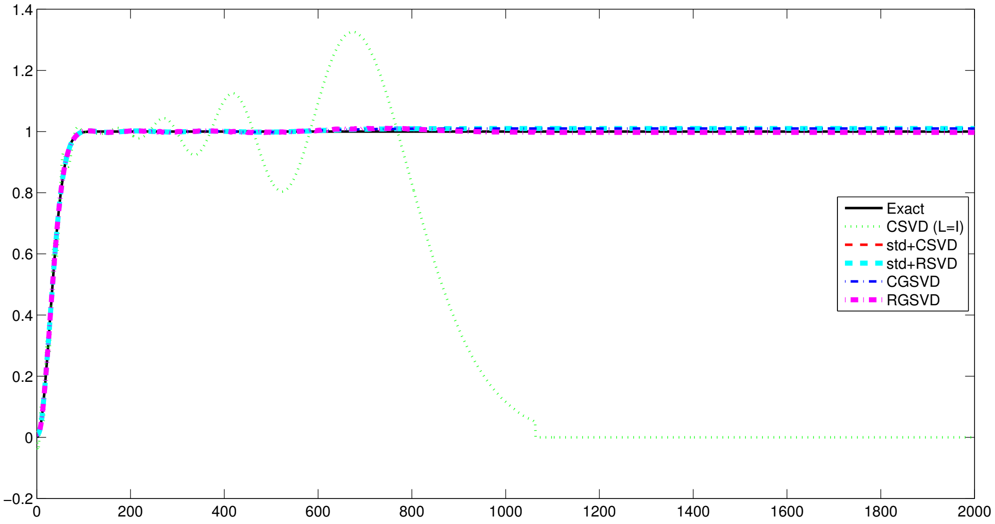

Figure 2: Example I_LAPLACE(n,2) of size $n = 2000$.

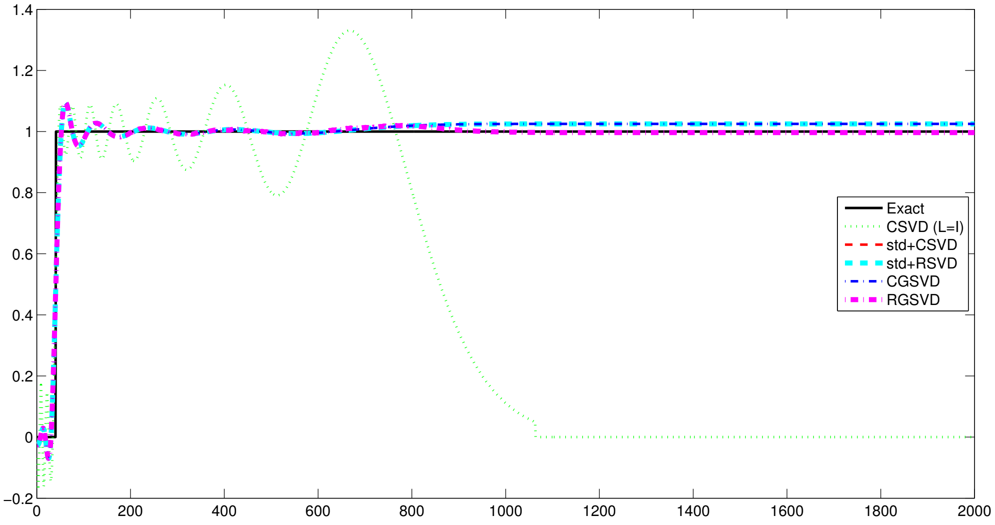

Figure 3: Example I_LAPLACE(n,4) of size $n = 2000$.

		$(L \neq I)$	std + CSVD		std + RSVD		CGSVD		RGSVD		$(L = I)$	CSVD	
e_g	n	μ	$T(s)$	err	$T(s)$	err	$T(s)$	err	$T(s)$	err	μ	$T(s)$	err
2	500	0.05	0.35	3.70E-03	0.09	3.70E-03	0.34	3.70E-03	0.16	2.20E-03	2.60E-04	0.11	7.56E-01
	1000	0.15	2.86	2.10E-03	0.29	2.10E-03	2.33	2.10E-03	0.94	5.00E-03	2.52E-04	1.03	7.49E-01
	2000	0.20	21.95	9.50E-03	1.06	9.50E-03	17.48	9.50E-03	5.41	5.50E-03	6.42E-05	8.07	7.66E-01
		$(L \neq I)$	std + CSVD		std + RSVD		CGSVD		RGSVD		$(L = I)$	CSVD	
e_g	n	μ	$T(s)$	err	$T(s)$	err	$T(s)$	err	$T(s)$	err	μ	$T(s)$	err
4	500	0.05	0.37	4.34E-02	0.09	4.34E-02	0.33	4.34E-02	0.16	4.42E-02	2.60E-04	0.12	7.54E-01
	1000	0.15	2.86	3.68E-02	0.30	3.68E-02	2.30	3.68E-02	0.94	3.71E-02	2.52E-04	0.92	7.50E-01
	2000	0.20	21.95	3.58E-02	1.06	3.58E-02	17.65	3.58E-02	5.36	3.06E-02	3.28E-04	7.94	7.47E-01

Table 2: Example I_LAPLACE(n, e_g) with $e_g = 2$, 4. Comparison of the computational times and solution accuracies.

		std + CSVD		std + RSVD		CGSVD		RGSVD	
	μ	$T(s)$	err	$T(s)$	err	$T(s)$	err	$T(s)$	err
FOXGOOD	5E-03	35.57	1.6673E-02	1.40	1.6673E-02	35.68	1.6673E-02	0.20	1.6674E-02
GRAVITY	5E-02	34.68	1.3886E-02	1.41	1.3886E-02	35.01	1.3886E-02	0.18	1.3886E-02
HEAT	2E-03	36.12	1.2441E-02	1.59	1.2325E-02	35.74	1.2441E-02	0.18	1.9250E-02
I_LAPLACE	1E-03	42.83	4.8063E-01	1.64	4.8063E-01	33.98	4.8063E-01	0.50	2.2425E-01
PHILLIPS	5E-03	36.79	7.2537E-02	1.41	4.0793E-02	36.61	7.2537E-02	0.17	3.0862E-02
SHAW	2E-02	34.75	2.2152E-02	1.40	2.2152E-02	36.93	2.2152E-02	0.17	2.2152E-02

Table 3: The computational time and accuracy of RGSVD compared with other three methods when the same regularized problem is solved with size $n = 2500$ and $L = bidiag(1, -1)$.

method: it can provide a regularized solution of the same accuracy as the one by the classical methods but requires much less computational time.

We remark that the truncated version of RGSVD also works quite well for all the examples tested in this section. Since the numerical results are very similar to the ones by the Tikhonov regularization, we do not present the numerical results by TGSVD in this work.

In this section we have tested two different approaches we have proposed for solving the Tikhonov regularization problems of general form (2). The first approach transforms a general problem (2) into a standard one (1), then solves the standard problem using the randomised SVD we developed in [18], while the second approach applies the generalized SVD directly for the general form (2). Based on many of our numerical experiments, the second approach is mostly more efficient than the first approach, as we can clearly see from the numerical results in Tables 1 and 3 for 6 different inverse problems. However, the first approach may work equally well as the second approach in some special cases, where the sample size does not need to be large and very efficient evaluations of the generalized inverse are available as the generalized inverses contribute to the major computational efforts in the first approach. Indeed this can be confirmed clearly from the numerical results shown in Table 2 for Example I_LAPLACE(n,e_g) with $e_g = 2$ or 4.

5 Concluding remarks

We have considered the randomized algorithms for the solutions of discrete ill-posed problems in general form. Several strategies are discussed to transform the problem of general form into the standard one, then the randomized strategies in [18] can be applied. The second approach we have proposed is to work on the problem of general form directly. We first reduce the original large-scale problem essentially by using the randomized algorithm RGSVD, so flops and memory are significantly saved. Our numerical experiments show that, using RGSVD we can still achieve the approximate regularized solutions of the same accuracy as the classical GSVD, but gain obvious robustness, stability and computational time as we need only to work on problems of much smaller size.

Acknowledgements

The authors would like to thank two anonymous referees for their many insightful and constructive comments and suggestions, which have led to a great improvement of the results and organisation of this work.

References

[1] H. Banks and K. Kunisch, Parameter Estimation Techniques for Distributed Systems, 1989, Boston, MA: Birkhauser.

[2] R. H. Chan and X. Q. Jin, An Introduction to Iterative Toeplitz Solvers, SIAM, 2007.

[3] D. Colton and R. Kress, Inverse Acoustic and Electromagnetic Scattering Theory, 1998, 2nd edn, Berlin: Springer.

[4] J. Demmel, L. Grigori, M. Hoemmen, and J. Langou, *Communication-optimal parallel and sequential QR and LU factorizations*, SIAM J. Sci. Comp. 34 (2012), pp. A206-A239.

[5] J. Demmel, L. Grigori, M. Gu, and H. Xiang, *Communication avoiding rank revealing QR factorization with column pivoting*, SIAM J. Matrix Anal. Appl., 36 (2015), pp. 55-89.

[6] L. Elden, *Algorithms for the regularization of ill-conditioned least squares problems*, BIT, 17 (1977), 134-145.

[7] H. W. Engl, M. Hanke and A. Neubauer, Regularization of Inverse Problems, 1996, Dordrecht: Kluwer.

[8] G. H. Golub and C. F. Van Loan, Matrix Computations, 4th edn, John Hopkins University Press, Baltimore, 2013.

[9] N. Halko, P. Martinsson and J. Tropp, *Finding structures with randomness: probabilistic algorithms for constructing approximate matrix decompositions*, SIAM Rev. 53 (2011), pp. 217-88.

[10] P. C. Hansen, *Regularization tools: A Matlab package for analysis and solution of discrete ill-posed problems*, Numer. Algorithms, 6 (1994), pp. 1-35. Software is available in Netlib at the web site http://www.netlib.org.

[11] P. C. Hansen, Rank Deficient and Discrete Ill-Posed Problems, SIAM, Philadelphia, 1998.

[12] P. C. Hansen, Discrete Inverse Problems: Insight and Algorithms, SIAM, Philadelphia, 2010.

[13] P. C. Hansen, *Oblique projections and standard-form transformations for discrete inverse problems*, Numer. Lin. Algebra Appl. 20 (2013), pp. 250-258.

[14] J. W. Ruge and K. Stüben, *Algebraic multigrid (AMG)*, in Multigrid Methods, ed.. S. F. McCormick, volume 3 of Frontiers in Applied Mathematics. SIAM, Philadelphia, PA. (1987), pp. 73-130.

[15] G. W. Stewart, *Computing the CS decomposition of a partitioned orthonormal matrix*, Numer. Math. 40 (1982), pp. 297-306.

[16] S. Van Huffel and J. Vandewalle, The Total Least Squares Problem: Computational Aspects and Analysis, SIAM, Philadelphia, PA, 1991.

[17] C. Van Loan, *Computing the CS and the generalized singular value decompositions*, Numer. Math. 46 (1985), pp. 479-491.

[18] H. Xiang and J. Zou, *Regularization with randomized SVD for large-scale discrete inverse problems*, Inverse Problem 29 (2013), 085008.