

A NEW INTERPRETATION OF THE WEIGHTED PSEUDOINVERSE AND ITS APPLICATIONS*

HAIBO LI†

Abstract. Consider the generalized linear least squares (GLS) problem $\min \|Lx\|_2$ such that $\|M(Ax - b)\|_2 = \min$. The weighted pseudoinverse A_{ML}^\dagger is the matrix that maps b to the minimum 2-norm solution of this GLS problem. By introducing a linear operator induced by $\{A, M, L\}$ between two finite-dimensional Hilbert spaces, we show that the minimum 2-norm solution of the GLS problem is equivalent to the minimum norm solution of a linear least squares problem involving this linear operator, and A_{ML}^\dagger can be expressed as the composition of the Moore–Penrose pseudoinverse of this linear operator and an orthogonal projector. With this new interpretation, we establish the generalized Moore–Penrose equations that completely characterize the weighted pseudoinverse, give a closed-form expression of the weighted pseudoinverse using the generalized singular value decomposition (GSVD), and propose a generalized LSQR (gLSQR) algorithm for iteratively solving the GLS problem. We construct several numerical examples to test the proposed iterative algorithm for solving GLS problems. Our results highlight the close connections between GLS, weighted pseudoinverse, GSVD, and gLSQR, providing new tools for both analysis and computations.

Key words. generalized least squares, weighted pseudoinverse, generalized Moore–Penrose equations, GSVD, generalized Golub–Kahan bidiagonalization, generalized LSQR

MSC codes. 15A09, 15A22, 65F10, 65F20

DOI. 10.1137/24M1686073

1. Introduction. Consider the generalized linear least squares (GLS) problem

$$(1.1) \quad \min_{x \in \mathbb{R}^n} \|Lx\|_2 \quad \text{such that} \quad \|M(Ax - b)\|_2 = \min,$$

where $A \in \mathbb{R}^{m \times n}$, $M \in \mathbb{R}^{q \times m}$, and $L \in \mathbb{R}^{p \times n}$. In some of the literature it is also called the weighted linear least squares problem. The GLS problem generalizes the standard least squares (LS) problem $\min \|x\|_2$ such that $\|Ax - b\|_2 = \min$ by incorporating weighting matrices M and L , which introduces additional constraints and objectives tailored to specific data characteristics. For instance, M might be used to increase the relative importance of accurate measurements, while L could adjust the regularization or constraint structure to improve stability or enforce certain properties in the solution [11, Chap. 6.1]. Such problems arise in a variety of practical applications, including scatter data approximation [35], functional data analysis [17], ill-posed inverse problems [9], surface fitting problems [34], and many others.

The GLS problem is relatively simple when L has full column rank, where it must have a unique solution. The case that both L and M have full column rank has been extensively studied in earlier literature; see, e.g., [1, 31]. For general rectangular matrices M and L , in [24] the authors proposed the concept of projection under seminorms to study the existence and structure of the solutions of (1.1). A well-known result is that the uniqueness of the solution of (1.1) is equivalent to $\mathcal{N}(MA) \cap \mathcal{N}(L) = \{\mathbf{0}\}$, where $\mathcal{N}(\cdot)$ is the null space of a matrix. In the case of nonuniqueness, there is a unique minimum 2-norm solution of (1.1). The GLS problem was then intensively

*Received by the editors August 19, 2024; accepted for publication (in revised form) by M. Tůma December 4, 2024; published electronically April 4, 2025.

<https://doi.org/10.1137/24M1686073>

†School of Mathematics and Statistics, The University of Melbourne, Parkville, VIC 3010, Australia (haibo.li@unimelb.edu.au).

studied in [8], where the author gave the expression of the general solutions. Specifically, the author demonstrated that the minimum 2-norm solution can be written as $A_{ML}^\dagger b$, where $A_{ML}^\dagger \in \mathbb{R}^{n \times m}$ is the so-called M, L weighted pseudoinverse of A that shares several properties analogous to the Moore–Penrose pseudoinverse.

The weighted pseudoinverse is a generalization of the Moore–Penrose pseudoinverse of a single matrix. Since the concept of the pseudoinverse was first introduced by Moore [25, 26] and later by Penrose [29, 30], it has received considerable attention and many applications [3, 10, 21, 22], and there have been several various generalizations of the Moore–Penrose pseudoinverse, such as the restricted pseudoinverse proposed for linear-constrained LS problems [2, 16, 23], the product generalized inverse [6], and another type of weighted pseudoinverse (different from that in this paper) [5, 32, 33]. Among these generalizations, the weighted pseudoinverse proposed in [8] has attracted significant attention. For example, in [13], the authors proposed an algorithm for computing the GSVD of $\{A, L\}$, where at each iteration $L_{IA}^\dagger z$ needs to be computed for some vector z ; here I is the identity matrix. Moreover, using L_{IA}^\dagger , the general-form Tikhonov regularization problem $\min_{x \in \mathbb{R}^n} \{\|Ax - b\|_2^2 + \lambda \|Lx\|_2^2\}$ can be transformed to a standard-form problem, which is much easier for analysis and computations; see, e.g., [14, 15].

However, computing the weighted pseudoinverse and solving the GLS problem are both quite challenging. Existing methods primarily depend on matrix factorizations, which are effective only for small-scale problems. For the case that $\mathcal{N}(MA) \cap \mathcal{N}(L) = \{\mathbf{0}\}$ with $M = I_m$, the identity matrix of order m , the author in [8] gave a closed-form expression of A_{IL}^\dagger using the generalized singular value decomposition (GSVD) of $\{A, L\}$, which can be used to compute the solution of (1.1). Furthermore, he proposed a more efficient algorithm for computing A_{IL}^\dagger based on QR factorizations, which avoids the GSVD computation. For large-scale matrices, however, there is a lack of efficient methods for computing A_{ML}^\dagger or for iteratively computing $A_{ML}^\dagger b$ for a given b . This may partly be due to an insufficient understanding of the properties of the weighted pseudoinverse. In contrast, the properties of the Moore–Penrose pseudoinverse A^\dagger are well-established, and a variety of computational methods are available for it. For example, there is a closed-form expression of A^\dagger by using the singular value decomposition (SVD) of A , and $A^\dagger b$ is the minimum 2-norm solution of $\min_{x \in \mathbb{R}^n} \|Ax - b\|_2$, which can be approximated efficiently by the iterative solver LSQR [28]. It would be beneficial to gain new insights into the weighted pseudoinverse and to establish deeper analogies with the Moore–Penrose pseudoinverse. This could enable the development of efficient iterative methods for computing A_{ML}^\dagger .

In this paper, we provide a new interpretation of the weighted pseudoinverse and use it to design an iterative algorithm for computing $A_{ML}^\dagger b$. To achieve this, we first introduce a linear operator \mathcal{A} between two finite-dimensional Hilbert spaces, with non-Euclidean inner products induced by the matrices $\{A, M, L\}$. Then we formulate an operator-type LS problem involving \mathcal{A} , showing that its minimum norm solution coincides with the minimum 2-norm solution of (1.1). This result establishes a connection between A_{ML}^\dagger and A^\dagger , the Moore–Penrose pseudoinverse of \mathcal{A} . Building on this connection, we derive a set of generalized Moore–Penrose equations that fully characterize the weighted pseudoinverse. Additionally, by using the GSVD of $\{A, L\}$, we give a closed-form expression for A_{IL}^\dagger , which is applicable regardless of whether or not $\mathcal{N}(A) \cap \mathcal{N}(L) = \{\mathbf{0}\}$. To address the practical computational challenges involving A_{ML}^\dagger , we extend the classical Golub–Kahan bidiagonalization (GKB) method [10] and propose a novel iterative algorithm called the *generalized LSQR* (gLSQR).

The design of this algorithm leverages the connection between A_{ML}^\dagger and \mathcal{A}^\dagger , which can efficiently compute $A_{ML}^\dagger b$ by iteratively refining the solutions to the GLS problem (1.1) without requiring any matrix factorizations. To demonstrate the effectiveness of gLSQR, we construct several numerical examples of GLS problems and show its ability to compute solutions with high accuracy across these diverse scenarios. The results in this paper highlight the close connections between GLS, weighted pseudoinverse, GSVD, and gLSQR, providing new tools for both analysis and computations of related applications.

The paper is organized as follows. In section 2 we review several basic properties of the LS problem and the Moore–Penrose pseudoinverse. In section 3 we analyze the GLS problem from the perspective of an equivalent operator-type LS problem. Building on this perspective, we offer a new interpretation of the weighted pseudoinverse and present several basic properties. In section 4 we generalize the GKB method and propose the gLSQR algorithm for iterative computing $A_{ML}^\dagger b$. In section 5 we construct several nontrivial numerical examples to test the gLSQR algorithm. Finally, we conclude the paper in section 6.

Throughout the paper, we denote by $\mathcal{N}(\cdot)$ and $\mathcal{R}(\cdot)$ the null space and range space of a matrix or linear operator, respectively, denote by $\mathbf{0}$ the zero matrix/vector with orders clear from the context, and denote by $\text{span}\{\cdot\}$ the subspace spanned by a group of vectors or columns of a matrix.

2. Linear least squares and pseudoinverse of linear operators. We review several basic properties of the LS problems and the pseudoinverse of linear operators in the context of Hilbert spaces; see, e.g., [1, 12] for more details. These properties will be used in the subsequent sections.

Let \mathcal{X} and \mathcal{Y} be two Hilbert spaces, and let $T : \mathcal{X} \rightarrow \mathcal{Y}$ be a bounded linear operator where its adjoint is denoted by T^* . Consider the linear operator equation $Tx = y$, which has a solution if and only if $y \in \mathcal{R}(T)$. Otherwise, we consider the least squares solution. An element $x \in \mathcal{X}$ is called a least squares solution of $Tx = y$ if it satisfies

$$(2.1) \quad \|Tx - y\|_{\mathcal{Y}} = \inf\{\|Tz - y\|_{\mathcal{Y}} : z \in \mathcal{X}\}.$$

If the set of all least squares solutions has an element of minimum \mathcal{X} -norm, i.e.,

$$(2.2) \quad \|x\|_{\mathcal{X}} = \inf\{\|z\|_{\mathcal{X}} : z \text{ is a least squares solution of } Tx = y\},$$

then we call such an x a best-approximate solution of $Tx = y$. The following well-known result describes the existence and uniqueness of the least squares solution and best-approximate solution.

THEOREM 2.1. *For the linear operator equation $Tx = y$, the following properties hold:*

- (1) *It has a least squares solution if and only if $y \in \mathcal{R}(T) + \mathcal{R}(T)^\perp$.*
- (2) *If (1) is satisfied, then x is a least squares solution if and only if*

$$(2.3) \quad T^*Tx = T^*y$$

holds, which is called the normal equation.

- (3) *If (1) is satisfied, then x is the unique best-approximate solution if and only if (2.3) is satisfied and $x \in \mathcal{N}(T)^\perp$.*

The best-approximate solution is closely related to the Moore–Penrose pseudoinverse of T , which is the linear operator mapping y to the best-approximate solution

of $Tx = y$. Based on Theorem 2.1, the definition of the Moore–Penrose pseudoinverse is as follows.

DEFINITION 2.2. For the bounded linear operator $T : \mathcal{X} \rightarrow \mathcal{Y}$, define its restriction as $\tilde{T} := T|_{\mathcal{N}(T)^\perp} : \mathcal{N}(T)^\perp \rightarrow \mathcal{R}(T)$. The Moore–Penrose pseudoinverse T^\dagger of T is defined as the unique linear extension of \tilde{T}^{-1} to $\mathcal{D}(T^\dagger) := \mathcal{R}(T) + \mathcal{R}(T)^\perp$ such that $\mathcal{N}(T^\dagger) = \mathcal{R}(T)^\perp$.

It has been proved that the following four “Moore–Penrose” equations hold:

$$(2.4) \quad \begin{cases} TT^\dagger T = T, \\ T^\dagger TT^\dagger = T^\dagger, \\ T^\dagger T = \mathcal{P}_{\mathcal{N}(T)^\perp}, \\ TT^\dagger = \mathcal{P}_{\overline{\mathcal{R}(T)}}|_{\mathcal{D}(T^\dagger)}, \end{cases}$$

where $\mathcal{P}_{\mathcal{S}}$ is the orthogonal operator onto a closed subspace \mathcal{S} . Moreover, the Moore–Penrose equations uniquely characterize T^\dagger , i.e., there exists a unique linear operator T^\dagger that satisfies equations (2.4), and the last two conditions can even be relaxed as that TT^\dagger and $T^\dagger T$ are two orthogonal projectors. The following limit property holds:

$$(2.5) \quad T^\dagger = \lim_{\delta \searrow 0} (T^*T + \delta I)^{-1} T^* = \lim_{\delta \searrow 0} T^* (TT^* + \delta I)^{-1},$$

where $I : \mathcal{X} \rightarrow \mathcal{X}$ is the identity operator. Using the pseudoinverse, the following well-known result describes the structure of the least squares solutions.

THEOREM 2.3. Let $y \in \mathcal{D}(T^\dagger)$. Then $x^\dagger := T^\dagger y$ is the unique best-approximate solution of $Tx = y$, and the set of all least squares solutions is $x^\dagger + \mathcal{N}(T)$.

Now we come back to the settings with matrices. By treating $A \in \mathbb{R}^{m \times n}$ as a linear operator between the Euclidean spaces \mathbb{R}^n and \mathbb{R}^m , all the above results directly apply to A . Let the SVD of A be $U^\top AV = \Sigma$, where $\Sigma = \begin{pmatrix} \Sigma_r & \\ & \mathbf{0} \end{pmatrix} \in \mathbb{R}^{m \times n}$ with $\Sigma_r = \text{diag}(\sigma_1, \dots, \sigma_r) \in \mathbb{R}^{r \times r}$, $\sigma_1 \geq \dots \geq \sigma_r > 0$, and $U = (u_1, \dots, u_m) \in \mathbb{R}^{m \times m}$ and $V = (v_1, \dots, v_n) \in \mathbb{R}^{n \times n}$ are orthogonal matrices. The pseudoinverse of A has the expression $A^\dagger = V\Sigma^\dagger U^\top$ with $\Sigma^\dagger = \begin{pmatrix} \Sigma_r^{-1} & \\ & \mathbf{0} \end{pmatrix} \in \mathbb{R}^{n \times m}$, and the LS problem $\min_{x \in \mathbb{R}^n} \|Ax - b\|_2$ has a unique minimum 2-norm solution $x^\dagger = A^\dagger b = \sum_{i=1}^r \frac{u_i^\top b}{\sigma_i} v_i$. We remark that for a compact linear operator T , there exists an analogous decomposition to SVD, called the singular value expansion (SVE). Using the SVE of T , we can also give a similar expression of x^\dagger for the linear operator equation; see, e.g., [19, Chap. 15.4].

For large-scale matrix A , the LSQR algorithm is an efficient iterative approach for the LS problems $\min_{x \in \mathbb{R}^n} \|Ax - b\|_2$. This algorithm is based on the GKB process, where the main computations are matrix-vector products involving A and A^\top . At the k th step, the GKB process of A and b generates two groups of 2-orthogonal vectors, which form orthonormal bases of the Krylov subspaces $\mathcal{K}_{k+1}(AA^\top, b)$ and $\mathcal{K}_k(A^\top A, A^\top b)$, respectively. Meanwhile, it reduces A to a $(k + 1) \times k$ lower bidiagonal matrix, which is then used in the LSQR algorithm to iteratively compute an approximate solution. Besides, the GKB process is often used as a precursor for computing a partial SVD of a large-scale matrix.

To summarize this section, we remark that the LS problem, Moore–Penrose pseudoinverse, SVD, GKB process, and LSQR algorithm are all closely related. Each of them plays an important role in matrix computation problems, from providing theoretical analysis tools to enhancing the efficiency of numerical computations. Clearer relationships among these concepts are illustrated in Figure 4.1 at the end of section 4.

3. Generalized linear least squares and weighted pseudoinverse. First, we present a result that characterizes the solutions of the GLS problem, which allows us to reformulate the GLS problem as an equivalent operator-type LS problem. We then provide a new interpretation of the weighted pseudoinverse and establish several of its basic properties.

3.1. Generalized linear least squares. In the following part, we use $\|x\|_C$ to denote the seminorm $(x^\top Cx)^{1/2}$ for a symmetric positive semidefinite C . It becomes a norm when C is strictly positive definite. The following result provides a criterion for determining a solution of the GLS problem.

THEOREM 3.1. *For the GLS problem*

$$(3.1) \quad \min_{x \in \mathbb{R}^n} \|x\|_Q \quad \text{such that} \quad \|Ax - b\|_P = \min,$$

where $A \in \mathbb{R}^{m \times n}$, and $P \in \mathbb{R}^{m \times m}$, $Q \in \mathbb{R}^{n \times n}$ are symmetric positive semidefinite matrices, let $G = A^\top PA + Q$. The following properties hold:

- (1) If x is a solution of (3.1), then $\mathcal{P}_{\mathcal{R}(G)}x$ is also a solution; conversely, if $x \in \mathcal{R}(G)$ is a solution, then $x + z$ is a solution for any $z \in \mathcal{N}(G)$.
- (2) The vector $x \in \mathbb{R}^n$ is a solution of (3.1) if and only if

$$(3.2) \quad \begin{cases} A^\top P(Ax - b) = \mathbf{0}, \\ x^\top Gz = 0 \quad \forall z \in \mathcal{N}(A^\top PA). \end{cases}$$

Proof. First note that G is symmetric positive semidefinite. Thus, any vector $x \in \mathbb{R}^n$ has the decomposition $x = \mathcal{P}_{\mathcal{R}(G)}x + \mathcal{P}_{\mathcal{N}(G)}x =: x_1 + x_2$ and $x_1 \perp x_2$, where \perp is the orthogonal relation in Euclidean spaces. Using the relation $\mathcal{N}(G) = \mathcal{N}(A^\top PA) \cap \mathcal{N}(Q)$, we can verify that

$$\|A(x_1 + x_2) - b\|_P = \|Ax_1 - b\|_P, \quad \|x_1 + x_2\|_Q = \|x_1\|_Q.$$

The first property immediately follows.

To prove the second property, suppose x is a solution to (3.1). Then it is a solution to the problem $\min_x \|Ax - b\|_P$. Taking the gradient of it leads to $A^\top P(Ax - b) = \mathbf{0}$. Now $\mathcal{P}_{\mathcal{R}(G)}x$ is a solution to (3.1). Note that

$$(3.3) \quad x^\top Gz = (\mathcal{P}_{\mathcal{R}(G)}x)^\top G(G^\dagger Gz) = (\mathcal{P}_{\mathcal{R}(G)}x)^\top G(\mathcal{P}_{\mathcal{R}(G)}z),$$

and $(\mathcal{R}(G), \langle \cdot, \cdot \rangle_G)$ is a Hilbert space with inner product $\langle x, x' \rangle_G := x^\top Gx'$. We only need to prove $\mathcal{P}_{\mathcal{R}(G)}x \perp_G \mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA))$, where \perp_G is the orthogonal relation in $(\mathcal{R}(G), \langle \cdot, \cdot \rangle_G)$. Since $\mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA))$ is a closed subspace of $(\mathcal{R}(G), \langle \cdot, \cdot \rangle_G)$, we have the decomposition $\mathcal{P}_{\mathcal{R}(G)}x = \bar{x}_1 + \bar{x}_2$ such that $\bar{x}_1 \in \mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA))$ and $\bar{x}_2 \perp_G \mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA))$. We only need to prove $\bar{x}_1 = \mathbf{0}$. First we prove $\mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA)) \subseteq \mathcal{N}(A^\top PA)$. To see it, for any $z \in \mathbb{R}^n$ we use the decomposition $z = z_1 + z_2$ such that $z_1 \in \mathcal{N}(G)$ and $z_2 \in \mathcal{N}(G)^\perp = \mathcal{R}(G)$, which indicates $\mathcal{P}_{\mathcal{R}(G)}z = z_2$. Thus, if $z \in \mathcal{N}(A^\top PA)$, then $A^\top PAz_2 = A^\top PAz - A^\top PAz_1 = A^\top PAz = 0$ since $z_1 \in \mathcal{N}(G) \subseteq \mathcal{N}(A^\top PA)$.

Notice that $\bar{x}_1 \in \mathcal{N}(A^\top PA)$, which leads to $\|A(\bar{x}_1 + \bar{x}_2) - b\|_P = \|A\bar{x}_2 - b\|_P$. This indicates that \bar{x}_2 is a solution to $\min_x \|Ax - b\|_P$. Since

$$\bar{x}_2^\top Q\bar{x}_1 = \bar{x}_2^\top (A^\top PA + Q)\bar{x}_1 = \bar{x}_2^\top G\bar{x}_1 = 0,$$

we have

$$\|\mathcal{P}_{\mathcal{R}(G)}x\|_Q^2 = \|\bar{x}_1\|_Q^2 + 2\bar{x}_2^\top Q\bar{x}_1 + \|\bar{x}_2\|_Q^2 = \|\bar{x}_1\|_Q^2 + \|\bar{x}_2\|_Q^2 \geq \|\bar{x}_2\|_Q^2.$$

Specifically, if the second relation of (3.2) is not satisfied, which means $\bar{x}_1 \neq \mathbf{0}$, it must hold that $\|\bar{x}_1\|_Q > 0$, which can be proved as follows. If $\|\bar{x}_1\|_Q = 0$, then $\bar{x}_1 \in \mathcal{N}(Q)$. Combining this with $\bar{x}_1 \in \mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA)) \subseteq \mathcal{N}(A^\top PA)$, it must hold that $\bar{x}_1 \in \mathcal{N}(Q) \cap \mathcal{N}(A^\top PA) = \mathcal{N}(G)$. Using the relation $\bar{x}_1 \in \mathcal{R}(G)$ and $\mathcal{N}(G) \cap \mathcal{R}(G) = \{\mathbf{0}\}$, we obtain $\bar{x}_1 = \mathbf{0}$. Therefore, if $\bar{x}_1 \neq \mathbf{0}$, then $\|\bar{x}_2\|_Q < \|\mathcal{P}_{\mathcal{R}(G)}x\|_Q = \|x\|_Q$, contradicting that x is a solution.

Now we prove that (3.2) is a sufficient condition. Since (3.1) has at least one solution in $\mathcal{R}(G)$, by the first property, we only need to prove that there is only one solution in $\mathcal{R}(G)$ that satisfies (3.2). The existence has already been proved. To see the uniqueness, suppose x_1 and x_2 are two such solutions. Then it must hold that $x_1 = x_2 + z$ with $z \in \mathcal{N}(A^\top PA)$. From $x_1, x_2 \in \mathcal{R}(G)$ and $x_1, x_2 \perp_G \mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA))$ we obtain $z \in \mathcal{R}(G)$ and $z \perp_G \mathcal{P}_{\mathcal{R}(M)}(\mathcal{N}(A^\top PA))$. Combining this with $z \in \mathcal{N}(A^\top PA)$ leads to $z \in \mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA))$. Therefore, we have $z \perp_G z$, leading to $z = \mathbf{0}$. This proves the uniqueness of the solution in $\mathcal{R}(G)$ that satisfies (3.2). \square

From Theorem 3.1 and its proof, we have the following result.

COROLLARY 3.2. *There exists a unique solution of (3.1) in $\mathcal{R}(G)$, which is the minimum 2-norm solution of (3.1). Denoting this solution by x^\dagger , the set of all the solutions of (3.1) is $x^\dagger + \mathcal{N}(G)$.*

Therefore, in order to solve (3.1), a key step is to seek the solution x^\dagger in $\mathcal{R}(G)$. To investigate the property of x^\dagger , we introduce the following linear operator:

$$(3.4) \quad \mathcal{A} : (\mathcal{R}(G), \langle \cdot, \cdot \rangle_G) \rightarrow (\mathcal{R}(P), \langle \cdot, \cdot \rangle_P), \quad v \mapsto \mathcal{P}_{\mathcal{R}(P)}Av,$$

where v and Av are column vectors under the canonical bases of \mathbb{R}^n and \mathbb{R}^m . It is obvious that \mathcal{A} is a bounded linear operator. Let

$$\mathcal{A}^* : (\mathcal{R}(P), \langle \cdot, \cdot \rangle_P) \rightarrow (\mathcal{R}(G), \langle \cdot, \cdot \rangle_G), \quad u \mapsto \mathcal{A}^*u$$

be the adjoint operator of \mathcal{A} , which is defined by the relation $\langle Av, u \rangle_P = \langle \mathcal{A}^*u, v \rangle_G$ for any $v \in (\mathcal{R}(G), \langle \cdot, \cdot \rangle_G)$ and $u \in (\mathcal{R}(P), \langle \cdot, \cdot \rangle_P)$. The following result describes the effect of \mathcal{A} on a vector under the canonical bases.

LEMMA 3.3. *Under the canonical bases of \mathbb{R}^n and \mathbb{R}^m , for any $u \in (\mathcal{R}(P), \langle \cdot, \cdot \rangle_P)$, it holds that*

$$(3.5) \quad \mathcal{A}^*u = G^\dagger A^\top Pu.$$

Proof. Under the canonical bases, it holds that

$$\langle Av, u \rangle_P = \langle \mathcal{A}^*u, v \rangle_G \Leftrightarrow (\mathcal{P}_{\mathcal{R}(P)}Av)^\top Pu = v^\top G(\mathcal{A}^*u).$$

Since $v \in \mathcal{R}(G)$, we have $\mathcal{P}_{\mathcal{R}(G)}v = v$ and

$$(\mathcal{P}_{\mathcal{R}(P)}Av)^\top P = (\mathcal{P}_{\mathcal{R}(P)}A\mathcal{P}_{\mathcal{R}(G)}v)^\top P = v^\top \mathcal{P}_{\mathcal{R}(G)}A^\top \mathcal{P}_{\mathcal{R}(P)}P = v^\top \mathcal{P}_{\mathcal{R}(G)}A^\top P.$$

Thus, we have

$$v^\top \mathcal{P}_{\mathcal{R}(G)}A^\top Pu = v^\top G(\mathcal{A}^*u) \Leftrightarrow v^\top (\mathcal{P}_{\mathcal{R}(G)}A^\top Pu - G(\mathcal{A}^*u)) = 0$$

for any $v \in \mathcal{R}(G)$. Noticing that $\mathcal{P}_{\mathcal{R}(G)}A^\top Pu - G(\mathcal{A}^*u) \in \mathcal{R}(G)$, it follows that $\mathcal{P}_{\mathcal{R}(G)}A^\top Pu = G(\mathcal{A}^*u)$. This equality can also be written as

$$GG^\dagger A^\top Pu = G(\mathcal{A}^*u) \Rightarrow G^\dagger GG^\dagger A^\top Pu = G^\dagger G(\mathcal{A}^*u) \Leftrightarrow G^\dagger A^\top Pu = \mathcal{P}_{\mathcal{R}(G)}(\mathcal{A}^*u).$$

Since $\mathcal{A}^*u \in \mathcal{R}(G)$, we immediately obtain $\mathcal{A}^*u = G^\dagger A^\top Pu$. \square

Now we can reformulate the minimum 2-norm solution of (3.1) as the solution of the following equivalent operator-type LS problem.

THEOREM 3.4. *Let $\mathcal{X} := (\mathcal{R}(G), \langle \cdot, \cdot \rangle_G)$ and $\mathcal{Y} := (\mathcal{R}(P), \langle \cdot, \cdot \rangle_P)$. The minimum $\|\cdot\|_{\mathcal{X}}$ -norm solution of the LS problem*

$$(3.6) \quad \min_{v \in \mathcal{X}} \|\mathcal{A}v - \mathcal{P}_{\mathcal{R}(P)}b\|_{\mathcal{Y}}$$

is the unique solution of (3.1) in $\mathcal{R}(G)$.

Proof. First note that (3.6) has a unique $\|\cdot\|_{\mathcal{X}}$ -norm solution. In fact, v is such a solution if and only if

$$(3.7) \quad \begin{cases} \mathcal{A}^*(\mathcal{A}v - \mathcal{P}_{\mathcal{R}(P)}b) = \mathbf{0}, \\ v \perp_{\mathcal{X}} \mathcal{N}(\mathcal{A}), \end{cases}$$

where $\perp_{\mathcal{X}}$ is the orthogonal relation in the Hilbert space \mathcal{X} . We only need to prove the equivalence between (3.2) and (3.7) for $x \in \mathcal{R}(G)$. For notational consistency, here we uniformly use v instead of x . The proof includes the following two steps.

Step 1: Prove $A^\top P(Av - b) = \mathbf{0} \Leftrightarrow \mathcal{A}^*(Av - \mathcal{P}_{\mathcal{R}(P)}b) = \mathbf{0}$ for any $v \in \mathcal{R}(G)$. By Lemma 3.3, we have

$$\mathcal{A}^*(Av - \mathcal{P}_{\mathcal{R}(P)}b) = G^\dagger A^\top P(\mathcal{P}_{\mathcal{R}(P)}Av - \mathcal{P}_{\mathcal{R}(P)}b) = G^\dagger A^\top P(Av - b).$$

Thus, the “ \Rightarrow ” relation is obvious. To get the “ \Leftarrow ” relation, suppose $G^\dagger A^\top P(Av - b) = G^\dagger u = \mathbf{0}$. Let the Cholesky factorization of P be $P = L_P^\top L_P$. Then

$$u \in \mathcal{N}(G^\dagger) = \mathcal{N}(G) = \mathcal{N}(A^\top PA) \cap \mathcal{N}(Q) = \mathcal{N}(L_P A) \cap \mathcal{N}(Q) \subseteq \mathcal{N}(L_P A),$$

since G is symmetric and $\mathcal{N}(A^\top PA) = \mathcal{N}(L_P A)$. On the other hand, we have $u = A^\top P(Av - b) \in \mathcal{R}((L_P A)^\top) = \mathcal{N}(L_P A)^\perp$. Therefore, $u \in \mathcal{N}(L_P A) \cap \mathcal{N}(L_P A)^\perp = \{\mathbf{0}\}$, leading to $u = \mathbf{0}$. This proves the “ \Leftarrow ” relation.

Step 2: Prove $v \perp_G \mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA)) \Leftrightarrow v \perp_{\mathcal{X}} \mathcal{N}(\mathcal{A})$ for any $v \in \mathcal{R}(G)$. Since v is a vector under the canonical basis, we only need to prove $\mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA)) = \mathcal{N}(\mathcal{A})$. Note that $\mathcal{N}(\mathcal{A}) = \{v \in \mathcal{R}(G) : \mathcal{P}_{\mathcal{R}(P)}Av = \mathbf{0}\}$. In the proof of Theorem 3.1 we have already shown that $\mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA)) = \mathcal{R}(G) \cap \mathcal{N}(A^\top PA)$. Thus, for any $v \in \mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA))$, we have

$$A^\top PAv = \mathbf{0} \Rightarrow L_P Av = \mathbf{0} \Rightarrow \mathcal{P}_{\mathcal{R}(P)}Av = P^\dagger L_P^\top L_P Av = \mathbf{0}.$$

This implies that $v \in \mathcal{N}(\mathcal{A})$, and then $\mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA)) \subseteq \mathcal{N}(\mathcal{A})$. To prove the “ \supseteq ” relation, let $v \in \mathcal{N}(\mathcal{A})$. Then $v \in \mathcal{R}(G)$ and $Av \in \mathcal{N}(P)$. It follows that $A^\top PAv = \mathbf{0}$, leading to $v \in \mathcal{R}(G) \cap \mathcal{N}(A^\top PA)$. This proves $\mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA)) \supseteq \mathcal{N}(\mathcal{A})$. The whole proof is then completed. \square

Theorem 3.4 enables us to study the GLS problem by applying the extensive tools available for LS problems, thereby facilitating both analysis and computations. Building on this theorem, we give a new interpretation of the weighted pseudoinverse in the following subsection.

3.2. Weighted pseudoinverse and its properties. In this subsection, we come back to the GLS problem of the form (1.1). By setting $P = M^T M$ and $Q = L^T L$, the two problems (1.1) and (3.1) are essentially the same. The following result proposed in [8] gives the structure of a solution of the GLS problem.

THEOREM 3.5 (see [8]). *For any $A \in \mathbb{R}^{m \times n}$, $M \in \mathbb{R}^{q \times m}$, and $L \in \mathbb{R}^{p \times n}$, the problem*

$$(3.8) \quad \min_{x \in \mathbb{R}^n} \|Lx\|_2 \quad \text{such that} \quad \|M(Ax - b)\|_2 = \min$$

has the general solution

$$(3.9) \quad x = (I_n - (LP_{\mathcal{N}(MA)})^\dagger L)(MA)^\dagger Mb + \mathcal{P}_{\mathcal{N}(MA)}(I_n - (LP_{\mathcal{N}(MA)})^\dagger LP_{\mathcal{N}(MA)})z,$$

where $z \in \mathbb{R}^n$ is arbitrary.

Remark 3.1. In (3.1), if we let $P = M^T M$ and $Q = L^T L$, then it is equivalent to (3.8). In the remainder of the paper, we always treat (3.1) and (3.8) as two equivalent formulations with $P = M^T M$ and $Q = L^T L$.

It is shown that $\mathcal{N}(MA) \cap \mathcal{N}(L) = \{\mathcal{P}_{\mathcal{N}(MA)}(I_n - (LP_{\mathcal{N}(MA)})^\dagger LP_{\mathcal{N}(MA)})z : z \in \mathbb{R}^n\}$, and $x^\dagger := (I_n - (LP_{\mathcal{N}(MA)})^\dagger L)(MA)^\dagger Mb$ is 2-orthogonal to $\mathcal{N}(MA) \cap \mathcal{N}(L)$; thereby it is the minimum 2-norm solution of (3.8). To describe the map that takes b to x^\dagger , define the matrix

$$(3.10) \quad A_{ML}^\dagger := (I_n - (LP_{\mathcal{N}(MA)})^\dagger L)(MA)^\dagger M,$$

which is called the M, L -weighted pseudoinverse of A . Note that $x^\dagger = A_{ML}^\dagger b$, and note from Theorem 3.4 that $x^\dagger = \mathcal{A}^\dagger \mathcal{P}_{\mathcal{R}(P)} b$ for any $b \in \mathbb{R}^m$. We immediately have the following result.

THEOREM 3.6. *Following the notation in Theorems 3.5 and 3.4, under the canonical bases of \mathbb{R}^n and \mathbb{R}^m , it holds that*

$$(3.11) \quad A_{ML}^\dagger = \mathcal{A}^\dagger \mathcal{P}_{\mathcal{R}(P)}.$$

Theorem 3.6 establishes a connection between the weighted pseudoinverse and the Moore–Penrose pseudoinverse. Specifically, if M has full column rank, then $\mathcal{P}_{\mathcal{R}(P)} = I_m$, and A_{ML}^\dagger is essentially the pseudoinverse \mathcal{A}^\dagger . Using this new interpretation, we derive the following “generalized” Moore–Penrose equations to characterize the weighted pseudoinverse.

THEOREM 3.7. *For the weighted pseudoinverse $X := A_{ML}^\dagger$, the following generalized Moore–Penrose equations hold:*

$$(3.12) \quad \begin{cases} XAX = X, \\ MAXA = MA, \\ (M^T MAX)^\top = M^T MAX, \\ (GXAG^\dagger)^\top = XA, \\ XM^\dagger M = X. \end{cases}$$

Moreover, A_{ML}^\dagger is the unique solution of the above matrix equations.

Proof. Using (3.11) and $\mathcal{P}_{\mathcal{R}(P)} = \mathcal{P}_{\mathcal{N}(P)^\perp} = \mathcal{P}_{\mathcal{N}(M)^\perp} = M^\dagger M$, the fifth identity immediately follows. We use the identities (2.4) to prove the first four identities. Since $\mathcal{A}^\dagger \mathcal{A} \mathcal{A}^\dagger = \mathcal{A}^\dagger$, for any $u \in \mathbb{R}^m$ under the canonical basis, it holds that

$$A_{ML}^\dagger u = \mathcal{A}^\dagger \mathcal{P}_{\mathcal{R}(P)} u = \mathcal{A}^\dagger \mathcal{A} \mathcal{A}^\dagger \mathcal{P}_{\mathcal{R}(P)} u = \mathcal{A}^\dagger \mathcal{P}_{\mathcal{R}(P)} A A_{ML}^\dagger u = A_{ML}^\dagger A A_{ML}^\dagger u,$$

which implies the first identity. Notice that $\mathcal{P}_{\mathcal{R}(P)} A \mathcal{P}_{\mathcal{N}(G)} v = P^\dagger P A \mathcal{P}_{\mathcal{N}(G)} v = \mathbf{0}$ due to $\mathcal{N}(G) \subseteq \mathcal{N}(A^\top P A)$, implying that $\mathcal{P}_{\mathcal{R}(P)} A v = \mathcal{P}_{\mathcal{R}(P)} A \mathcal{P}_{\mathcal{R}(G)} v$ for any $v \in \mathbb{R}^n$. Therefore, it holds that

$$\begin{aligned} \mathcal{P}_{\mathcal{R}(P)} A v &= A \mathcal{P}_{\mathcal{R}(G)} v = \mathcal{A} \mathcal{A}^\dagger A \mathcal{P}_{\mathcal{R}(G)} v \\ &= \mathcal{P}_{\mathcal{R}(P)} A \mathcal{A}^\dagger \mathcal{P}_{\mathcal{R}(P)} A \mathcal{P}_{\mathcal{R}(G)} v = \mathcal{P}_{\mathcal{R}(P)} A A_{ML}^\dagger A v. \end{aligned}$$

This implies $M^\dagger M A = M^\dagger M A A_{ML}^\dagger A$, leading to the second identity. For the third identity, use the relation

$$\langle \mathcal{A} \mathcal{A}^\dagger \mathcal{P}_{\mathcal{R}(P)} u, \mathcal{P}_{\mathcal{R}(P)} u' \rangle_P = \langle \mathcal{P}_{\mathcal{R}(P)} u, (\mathcal{A} \mathcal{A}^\dagger)^* \mathcal{P}_{\mathcal{R}(P)} u' \rangle_P$$

for any $u, u' \in \mathbb{R}^m$, which is equivalent to

$$\begin{aligned} \left(\mathcal{P}_{\mathcal{R}(P)} A A_{ML}^\dagger u \right)^\top P \mathcal{P}_{\mathcal{R}(P)} u' &= \left(\mathcal{P}_{\mathcal{R}(P)} u \right)^\top P (\mathcal{A} \mathcal{A}^\dagger)^* \mathcal{P}_{\mathcal{R}(P)} u' \\ \Leftrightarrow u^\top \left(A A_{ML}^\dagger \right)^\top P u' &= u^\top P (\mathcal{A} \mathcal{A}^\dagger)^* \mathcal{P}_{\mathcal{R}(P)} u'. \end{aligned}$$

It follows that $P (\mathcal{A} \mathcal{A}^\dagger)^* \mathcal{P}_{\mathcal{R}(P)} u' = (A A_{ML}^\dagger)^\top P u'$ for any $u' \in \mathbb{R}^m$. Using the fourth identity of (2.4), which implies $(\mathcal{A} \mathcal{A}^\dagger)^* = \mathcal{A} \mathcal{A}^\dagger$, we have

$$P A A_{ML}^\dagger u' = P \mathcal{A} \mathcal{A}^\dagger \mathcal{P}_{\mathcal{R}(P)} u' = \left(A A_{ML}^\dagger \right)^\top P u',$$

leading to $(A A_{ML}^\dagger)^\top M^\top M = M^\top M A A_{ML}^\dagger$, which is just the third identity. For the fourth identity, use the relation

$$\langle \mathcal{A}^\dagger A \mathcal{P}_{\mathcal{R}(G)} v, \mathcal{P}_{\mathcal{R}(G)} v' \rangle_G = \langle \mathcal{P}_{\mathcal{R}(G)} v, (\mathcal{A}^\dagger A)^* \mathcal{P}_{\mathcal{R}(G)} v' \rangle_G$$

for any $v, v' \in \mathbb{R}^n$, which is equivalent to

$$\begin{aligned} \left(A_{ML}^\dagger A \mathcal{P}_{\mathcal{R}(G)} v \right)^\top G \mathcal{P}_{\mathcal{R}(G)} v' &= \left(\mathcal{P}_{\mathcal{R}(G)} v \right)^\top G (\mathcal{A}^\dagger A)^* \mathcal{P}_{\mathcal{R}(G)} v' \\ \Leftrightarrow v^\top \left(A_{ML}^\dagger A \right)^\top G v' &= v^\top G (\mathcal{A}^\dagger A)^* \mathcal{P}_{\mathcal{R}(G)} v', \end{aligned}$$

where we have used $\mathcal{P}_{\mathcal{R}(P)} A \mathcal{P}_{\mathcal{R}(G)} v = \mathcal{P}_{\mathcal{R}(P)} A v$. It follows that $G (\mathcal{A}^\dagger A)^* \mathcal{P}_{\mathcal{R}(G)} v' = (A_{ML}^\dagger A)^\top G v'$ for any $v' \in \mathbb{R}^n$. Using the third identity of (2.4), which implies $(\mathcal{A}^\dagger A)^* = \mathcal{A}^\dagger A$, we have

$$G A_{ML}^\dagger A v' = G \mathcal{A}^\dagger A \mathcal{P}_{\mathcal{R}(G)} v' = \left(A_{ML}^\dagger A \right)^\top G v',$$

leading to $A_{ML}^\dagger A v' = G^\dagger G A_{ML}^\dagger A v' = G^\dagger (A_{ML}^\dagger A)^\top G v'$. Thus, it holds that $A_{ML}^\dagger A = G^\dagger (A_{ML}^\dagger A)^\top G$, which is the fourth identity.

To show that X is unique, first we prove that if X satisfies the first four identities, then X must satisfy

$$(3.13) \quad \mathcal{P}_{\mathcal{R}(G)} X = X.$$

Using the first and fourth identities, we get

$$G^\dagger GX = G^\dagger GXAX = G^\dagger G(G^\dagger A^\top X^\top G)X = G^\dagger A^\top X^\top GX = XAX = X.$$

Suppose Y is another matrix satisfying equations (3.12); then Y also satisfies (3.13). Note that the fifth identity implies $X = X\mathcal{P}_{\mathcal{R}(P)} = XP^\dagger P$. It follows that

$$\begin{aligned} X &= (XA)X = (GX\mathcal{P}_{\mathcal{R}(P)}AG^\dagger)^\top X = (\mathcal{P}_{\mathcal{R}(P)}AG^\dagger)^\top X^\top GX \\ &\stackrel{(1)}{=} (\mathcal{P}_{\mathcal{R}(P)}A\mathcal{P}_{\mathcal{R}(G)}YAG^\dagger)^\top X^\top GX = (G^\dagger A^\top Y^\top \mathcal{P}_{\mathcal{R}(G)}) A^\top (X\mathcal{P}_{\mathcal{R}(P)})^\top GX \\ &\stackrel{(2)}{=} YAG^\dagger A^\top X^\top GX = YA(XA)X = YAX = YAYAX = (YP^\dagger P)AYAX \\ &= YP^\dagger(PAY)^\top AX = YP^\dagger Y^\top A^\top PAX = Y(X^\top A^\top PAYP^\dagger)^\top \\ &= Y(PAXAYP^\dagger)^\top = Y(PAYP^\dagger)^\top \stackrel{(3)}{=} Y\mathcal{P}_{\mathcal{R}(P)}AY = YAY = Y, \end{aligned}$$

where for “(1)” we use (3.13) and $\mathcal{P}_{\mathcal{R}(P)}A = \mathcal{P}_{\mathcal{R}(P)}AYA$ due to the second identity, for “(2)” we use $G^\dagger A^\top Y^\top \mathcal{P}_{\mathcal{R}(G)} = YAG^\dagger$ due to the fourth identity, and for “(3)” we use $(PAYP^\dagger)^\top = \mathcal{P}_{\mathcal{R}(P)}AY$ due to the third identity. This proves the uniqueness of X . \square

In [8], the author presented four identities similar to the Moore–Penrose equations of the pseudoinverse. The first three of them are identical to the first three listed in (3.12), while the fourth identity is $(L^\top LXA)^\top = L^\top LXA$. However, it remains an open problem whether the weighted pseudoinverse is uniquely determined by these four equations. In contrast, equations (3.12) completely characterize the weighted pseudoinverse. The first four identities correspond to the Moore–Penrose equations of a matrix, while the last identity describes an additional constraint on A_{ML}^\dagger arising from M . Specifically, the fifth identity is trivial when M has full column rank.

The weighted pseudoinverse satisfies the following limit property.

THEOREM 3.8. *Following the notation in Theorems 3.5 and 3.6, let $G = A^\top PA + Q$. It holds that*

$$(3.14) \quad \lim_{\delta \searrow 0} (A^\top PA + \delta G)^\dagger A^\top P = A_{ML}^\dagger.$$

Proof. First we show that for any $\delta > 0$, under the canonical bases of \mathbb{R}^n and \mathbb{R}^m , it holds for any $u \in \mathcal{R}(P)$ that

$$(3.15) \quad (\mathcal{A}^* \mathcal{A} + \delta I)^{-1} \mathcal{A}^* u = (A^\top PA + \delta G)^\dagger A^\top Pu.$$

Notice that $\mathcal{N}(A^\top PA + \delta G) = \mathcal{N}(A^\top PA) \cap \mathcal{N}(G) = \mathcal{N}(G)$, which leads to $\mathcal{R}((A^\top PA + \delta G)^\dagger) = \mathcal{R}(A^\top PA + \delta G) = \mathcal{R}(G)$. Thus, we have $(A^\top PA + \delta G)^\dagger A^\top Pu \in (\mathcal{R}(G), \langle \cdot, \cdot \rangle_G)$, and we only need to prove $(\mathcal{A}^* \mathcal{A} + \delta I)(A^\top PA + \delta G)^\dagger A^\top Pu = \mathcal{A}^* u$. By Lemma 3.3, we have

$$\begin{aligned} (\mathcal{A}^* \mathcal{A} + \delta I)(A^\top PA + \delta G)^\dagger A^\top Pu &= (G^\dagger A^\top P\mathcal{P}_{\mathcal{R}(P)}A + \delta I_n)(A^\top PA + \delta G)^\dagger A^\top Pu \\ &= (G^\dagger A^\top PA + \delta I_n)(A^\top PA + \delta G)^\dagger A^\top Pu =: w, \end{aligned}$$

and $\mathcal{A}^* u = G^\dagger A^\top Pu$. Therefore, we only need to show that w is the minimum 2-norm solution of $\min_x \|Gx - A^\top Pu\|_2$, which is equivalent to showing that $w \in \mathcal{N}(G)^\perp = \mathcal{R}(G)$ and $G^\top Gw = G^\top A^\top Pu$. Since $\mathcal{R}(G^\dagger) = \mathcal{R}(G)$ and $\mathcal{R}((A^\top PA + \delta G)^\dagger) = \mathcal{R}(G)$, it follows that $w \in \mathcal{R}(G)$. Also, using $G = G^\top$ we have

$$\begin{aligned} G^\top Gw &= GG(G^\dagger A^\top PA + \delta I_n)(A^\top PA + \delta G)^\dagger A^\top Pu \\ &= G(A^\top PA + \delta G)(A^\top PA + \delta G)^\dagger A^\top Pu \\ &= G\mathcal{P}_{\mathcal{R}(A^\top PA + \delta G)}A^\top Pu \\ &= G\mathcal{P}_{\mathcal{R}(G)}A^\top Pu = G^\top A^\top Pu. \end{aligned}$$

This proves (3.15).

By (3.15), for any $b \in \mathbb{R}^m$, it holds that

$$(\mathcal{A}^* \mathcal{A} + \delta I)^{-1} \mathcal{A}^* \mathcal{P}_{\mathcal{R}(P)} b = (A^\top PA + \delta G)^\dagger A^\top P \mathcal{P}_{\mathcal{R}(P)} b = (A^\top PA + \delta G)^\dagger A^\top P b,$$

which indicates that $(\mathcal{A}^* \mathcal{A} + \delta I)^{-1} \mathcal{A}^* \mathcal{P}_{\mathcal{R}(P)} = (A^\top PA + \delta G)^\dagger A^\top P$ under the canonical bases. Using Theorem 3.6 and (2.5), we obtain

$$\lim_{\delta \searrow 0} (A^\top PA + \delta G)^\dagger A^\top P = \left(\lim_{\delta \searrow 0} (\mathcal{A}^* \mathcal{A} + \delta I)^{-1} \mathcal{A}^* \right) \mathcal{P}_{\mathcal{R}(P)} = \mathcal{A}^\dagger \mathcal{P}_{\mathcal{R}(P)} = A_{ML}^\dagger.$$

This completes the proof. □

In [8, Theorem 2.4], the author gave a similar limit property,

$$(3.16) \quad \lim_{\delta \searrow 0} (A^\top PA + \delta Q)^\dagger A^\top P = A_{ML}^\dagger,$$

but did not include a proof. Notice that $A^\top PA + \delta G = (1 + \delta)(A^\top PA + \frac{\delta}{1+\delta}Q)$. Therefore, (3.14) and (3.16) are equivalent.

In many scenarios, researchers are more interested in (3.8) with $M = I_m$. In this case, we have $A_{IL} = \mathcal{A}^\dagger$, which means that the I, L -weighted pseudoinverse of A is nothing but the pseudoinverse of the linear operator \mathcal{A} . Moreover, it has a direct relation with the GSVD of the matrix pair $\{A, L\}$. Let us review the GSVD proposed in [27].

THEOREM 3.9 (GSVD). *Let $A \in \mathbb{R}^{m \times n}$ and $L \in \mathbb{R}^{p \times n}$. There exist orthogonal matrices $U_A \in \mathbb{R}^{m \times m}$, $U_L \in \mathbb{R}^{p \times p}$ and invertible matrix $X \in \mathbb{R}^{n \times n}$, such that the GSVD of $\{A, L\}$ has the form*

$$(3.17a) \quad A = U_A \Sigma_A X^{-1}, \quad L = U_L \Sigma_L X^{-1},$$

with

$$(3.17b) \quad \Sigma_A = \begin{pmatrix} C_A & \mathbf{0} \\ r & n-r \end{pmatrix} m, \quad \Sigma_L = \begin{pmatrix} S_L & \mathbf{0} \\ r & n-r \end{pmatrix} p$$

and

$$(3.17c) \quad C_A = \begin{pmatrix} I_{q_1} & & \\ & C_{q_2} & \\ & & \mathbf{0} \end{pmatrix} \begin{matrix} q_1 \\ q_2 \\ m - q_1 - q_2 \end{matrix}, \quad S_L = \begin{pmatrix} \mathbf{0} & & \\ & S_{q_2} & \\ & & I_{q_3} \end{pmatrix} \begin{matrix} p - r + q_1 \\ q_2 \\ q_3 \end{matrix},$$

where $q_1 + q_2 + q_3 = r = \text{rank}((A^\top, L^\top)^\top)$ and $C_A^\top C_A + S_L^\top S_L = I_r$.

In [8], the author shows that if $\mathcal{N}(A) \cap \mathcal{N}(L) = \{\mathbf{0}\}$, then $A_{IL}^\dagger = X \Sigma_A^\dagger U_A^\top$. In the following result, we give a similar expression for A_{IL}^\dagger , which is applicable regardless of whether $\mathcal{N}(A) \cap \mathcal{N}(L) = \{\mathbf{0}\}$ or not.

THEOREM 3.10. *For any two matrices $A \in \mathbb{R}^{m \times n}$ and $L \in \mathbb{R}^{p \times n}$ with the GSVD as described in Theorem 3.9, let $G = A^\top A + L^\top L$. Then*

$$(3.18) \quad A_{IL}^\dagger = \mathcal{P}_{\mathcal{R}(G)} X \Sigma_A^\dagger U_A^\top.$$

Proof. First we prove that $x := X \Sigma_A^\dagger U_A^\top b$ is a solution of (3.8). We only need to verify that the two conditions in (3.2) are satisfied. Since $AX = U_A \Sigma_A$, we have

$$A^\top (Ax - b) = X^{-\top} \Sigma_A^\top U_A^\top \left(U_A \Sigma_A \Sigma_A^\dagger U_A^\top b - b \right) = X^{-\top} \left(\Sigma_A^\top \Sigma_A \Sigma_A^\dagger - \Sigma_A^\top \right) U_A^\top b = \mathbf{0},$$

since we can easily verify that $\Sigma_A^\top \Sigma_A \Sigma_A^\dagger = \Sigma_A^\top$. For the second condition in (3.2), if we partition X as

$$(3.19) \quad X = \begin{pmatrix} X_1 & X_2 & X_3 & X_4 \\ q_1 & q_2 & q_3 & n-r \end{pmatrix} n$$

we can verify that $\mathcal{N}(A^\top A) = \mathcal{N}(A) = \mathcal{R}((X_3 \ X_4))$. On the other hand, it holds that $x \in \mathcal{R}(X \Sigma_A^\dagger)$ and

$$(3.20) \quad X \Sigma_A^\dagger = (X_1 \ X_2 \ X_3) C_A^\dagger = (X_1 \ X_2 C_{q_2}^{-1} \ \mathbf{0}),$$

which means that $x \in \mathcal{R}((X_1 \ X_2))$. Notice that

$$G = X^{-\top} \left(\begin{pmatrix} C_A^\top C_A & \\ & \mathbf{0} \end{pmatrix} + \begin{pmatrix} S_L^\top S_L & \\ & \mathbf{0} \end{pmatrix} \right) X^{-1} = X^{-\top} \begin{pmatrix} I_r & \\ & \mathbf{0} \end{pmatrix} X^{-1},$$

leading to $X^\top G X = \begin{pmatrix} I_r & \\ & \mathbf{0} \end{pmatrix}$. Thus, $\mathcal{R}(X_4) = \mathcal{N}(G)$ and the columns of $(X_1 \ X_2 \ X_3)$ are mutually G -orthonormal. It follows that $x^\top G z = 0$ for any $z \in \mathcal{N}(A^\top A) = \mathcal{R}((X_3 \ X_4))$.

Note that $\mathcal{P}_{\mathcal{R}(G)} x = \mathcal{P}_{\mathcal{R}(G)} X \Sigma_A^\dagger U_A^\top b \in \mathcal{R}(G)$, which is the minimum 2-norm solution of (3.8) for an arbitrary $b \in \mathbb{R}^n$. We immediately obtain (3.18). \square

This theorem provides a direct computational approach for A_{IL}^\dagger using the GSVD. However, if the matrices are very large, computing the GSVD is extremely expensive. In this case, we need an iterative approach to approximate $A_{ML}^\dagger b$ for any given b .

4. Iterative method for computing weighted pseudoinverse. By Theorem 3.5 and (3.10), computing $A_{ML}^\dagger b$ is equivalent to computing the minimum 2-norm solution of the GLS problem (3.1). We aim to approximate the solution of the GLS problem through an iterative process. The starting point comes from Theorem 3.4. To solve the LS problem (3.6), we apply the GKB process to the operator \mathcal{A} between the two Hilbert spaces \mathcal{X} and \mathcal{Y} ; see [4] for the GKB for LS problems in Hilbert spaces. Starting from the initial vector $\mathcal{P}_{\mathcal{R}(P)} b$, the recursive relations of GKB can be expressed as follows:

$$(4.1) \quad \begin{cases} \beta_1 u_1 = \mathcal{P}_{\mathcal{R}(P)} b, \\ \alpha_i v_i = \mathcal{A}^* u_i - \beta_i v_{i-1}, \\ \beta_{i+1} u_{i+1} = \mathcal{A} v_i - \alpha_i u_i, \end{cases}$$

where $u_i \in \mathcal{Y}$ and $v_i \in \mathcal{X}$, and α_i and β_i are positive scalars such that $\|v_i\|_{\mathcal{X}} = \|u_i\|_{\mathcal{Y}} = 1$. Note that $v_0 := \mathbf{0}$ for the initial step. We remark that it is assumed that

$\mathcal{P}_{\mathcal{R}(P)}b \neq \mathbf{0}$; otherwise, the case is trivial because $A_{ML}^\dagger b = \mathbf{0}$. Using Lemma 3.3, we present the matrix form of the above recursive relations:

$$(4.2) \quad \begin{cases} \beta_1 u_1 = PP^\dagger b, \\ \alpha_i v_i = G^\dagger A^\top P u_i - \beta_i v_{i-1}, \\ \beta_{i+1} u_{i+1} = PP^\dagger A v_i - \alpha_i u_i. \end{cases}$$

Note that if $G = I_n$ and $P = I_m$, then the above recursive relations correspond to the standard GKB process of the matrix A . We name the iterative process corresponding to (4.2) the *generalized Golub–Kahan bidiagonalization* (gGKB). Before giving the practical computation procedure, let us explore how to further reduce the computational cost. In fact, computations involving P^\dagger can be avoided, as demonstrated by the following result.

LEMMA 4.1. *For the recursive relations*

$$(4.3) \quad \begin{cases} \beta_1 \tilde{u}_1 = b, \\ \alpha_i \tilde{v}_i = G^\dagger A^\top P \tilde{u}_i - \beta_i \tilde{v}_{i-1}, \\ \beta_{i+1} \tilde{u}_{i+1} = A \tilde{v}_i - \alpha_i \tilde{u}_i, \end{cases}$$

where $\tilde{v}_0 := \mathbf{0}$ and the values of α_i and β_i are the same as in (4.2), it follows that $v_i = \tilde{v}_i$ and $u_i = PP^\dagger \tilde{u}_i$.

Proof. We prove it by mathematical induction. For $i = 1$, we have $\beta_1 PP^\dagger \tilde{u}_1 = PP^\dagger b = \beta_1 u_1$, implying $u_1 = PP^\dagger \tilde{u}_1$. Note that $\tilde{v}_0 = v_0$ and $P u_1 = PPP^\dagger \tilde{u}_1 = P \tilde{u}_1$. It follows that $\alpha_1 v_1 = \alpha_1 \tilde{v}_1$, meaning $v_1 = \tilde{v}_1$. Now assume $v_i = \tilde{v}_i$ and $u_i = PP^\dagger \tilde{u}_i$ for $i \geq 1$. Then

$$PP^\dagger A v_i - \alpha_i u_i = PP^\dagger A \tilde{v}_i - \alpha_i PP^\dagger \tilde{u}_i = PP^\dagger (A \tilde{v}_i - \alpha_i \tilde{u}_i) = \beta_{i+1} PP^\dagger \tilde{u}_{i+1}.$$

Thus, we have $\beta_{i+1} u_{i+1} = \beta_{i+1} PP^\dagger \tilde{u}_{i+1}$, meaning $u_{i+1} = PP^\dagger \tilde{u}_{i+1}$. Similar to the proof for $i = 1$, using $P u_{i+1} = PPP^\dagger \tilde{u}_{i+1} = P \tilde{u}_{i+1}$, we can also prove $v_{i+1} = \tilde{v}_{i+1}$. \square

Using the above result, we can simplify the computation if we only need to generate v_i but not u_i . At the initial step, we have

$$\beta_1 = \|PP^\dagger b\|_P = [(PP^\dagger b)^\top PPP^\dagger b]^{1/2} = (b^\top P b)^{1/2}.$$

At the i th step, to compute β_{i+1} , let $r_i = A v_i - \alpha_i \tilde{u}_i$. Then we have $\beta_{i+1} u_{i+1} = PP^\dagger r_i$. Thus, it follows that

$$\beta_{i+1} = \|PP^\dagger r_i\|_P = [(PP^\dagger r_i)^\top PPP^\dagger r_i]^{1/2} = (r_i^\top P r_i)^{1/2}.$$

Now we can give the whole iterative procedure of the gGKB process, as shown in Algorithm 4.1.

Note that at each iteration of gGKB we need to compute $G^\dagger \bar{s}$. For large-scale matrices, it is generally impractical to obtain G^\dagger directly. If G is sparse and positive definite, we can first apply the sparse Cholesky factorization to G and then compute G^\dagger . Otherwise, using the fact that G^\dagger is the minimum 2-norm solution of $\min_{s \in \mathbb{R}^n} \|Gs - \bar{s}\|_2$, we can apply the iterative solver LSQR to $\min_{s \in \mathbb{R}^n} \|Gs - \bar{s}\|_2$ to approximate $G^\dagger \bar{s}$ [28].

Algorithm 4.1. Generalized Golub–Kahan bidiagonalization (gGKB).

Input: $A \in \mathbb{R}^{m \times n}$, $P \in \mathbb{R}^{m \times m}$, $Q \in \mathbb{R}^{n \times n}$, $b \in \mathbb{R}^m$

- 1: Form $G = A^\top P A + Q$
- 2: Compute $\beta_1 = (b^\top P b)^{1/2}$, $\tilde{u}_1 = b/\beta_1$
- 3: Compute $\bar{s} = A^\top P u_1$, $s = G^\dagger \bar{s}$
- 4: $\alpha_1 = (s^\top G s)^{1/2}$, $v_1 = s/\alpha_1$
- 5: **for** $i = 1, 2, \dots, k$, **do**
- 6: $r = A v_i - \alpha_i \tilde{u}_i$
- 7: $\beta_{i+1} = (r^\top P r)^{1/2}$, $\tilde{u}_{i+1} = r/\beta_{i+1}$
- 8: $\bar{s} = A^\top P \tilde{u}_{i+1}$, $s = G^\dagger \bar{s} - \beta_{i+1} v_i$
- 9: $\alpha_{i+1} = (s^\top G s)^{1/2}$, $v_{i+1} = s/\alpha_{i+1}$
- 10: **end for**

Output: $\{\alpha_i, \beta_i\}_{i=1}^{k+1}$, $\{\tilde{u}_i, v_i\}_{i=1}^{k+1}$ $\triangleright u_i = P P^\dagger \tilde{u}_i$

We remark that in [20] the author generalized the GKB process for computing nontrivial GSVD components of $\{A, L\}$. Here the proposed gGKB process is used to iteratively solve the GLS problem and is more versatile, as it can handle cases where P is noninvertible.

The following result describes the subspaces generated by gGKB.

PROPOSITION 4.2. *The gGKB process generates vectors $v_i \in \mathcal{R}(G)$ and $u_i \in \mathcal{R}(P)$, and $\{v_i\}_{i=1}^k$ is a G -orthonormal basis of the Krylov subspace*

$$(4.4) \quad \mathcal{K}_k(G^\dagger A^\top P A, G^\dagger A^\top P b) = \text{span}\{(G^\dagger A^\top P A)^i G^\dagger A^\top P b\}_{i=0}^{k-1},$$

and $\{u_i\}_{i=1}^k$ is a P -orthonormal basis of the Krylov subspace

$$(4.5) \quad \mathcal{K}_k(P P^\dagger A G^\dagger A^\top P, P P^\dagger b) = P P^\dagger \text{span}\{(A G^\dagger A^\top P)^i b\}_{i=0}^{k-1}.$$

Proof. The proof is based on the property of GKB for linear compact operators. As demonstrated above, the gGKB of A is essentially the GKB of \mathcal{A} between the two Hilbert spaces \mathcal{X} and \mathcal{Y} . Therefore, the generated vectors satisfy $v_i \in \mathcal{R}(G)$ and $u_i \in \mathcal{R}(P)$, and $\{v_i\}_{i=1}^k$ and $\{u_i\}_{i=1}^k$ are G - and P -orthonormal bases of the Krylov subspaces $\mathcal{K}_k(\mathcal{A}^* \mathcal{A}, \mathcal{A}^* \mathcal{P}_{\mathcal{R}(P)} b)$ and $\mathcal{K}_k(\mathcal{A} \mathcal{A}^*, \mathcal{P}_{\mathcal{R}(P)} b)$, respectively. By Lemma 3.3, we have

$$(\mathcal{A}^* \mathcal{A})^i \mathcal{A}^* \mathcal{P}_{\mathcal{R}(P)} b = (G^\dagger A^\top P P P^\dagger A)^i G^\dagger A^\top P P P^\dagger b = (G^\dagger A^\top P A)^i G^\dagger A^\top P b$$

and

$$(\mathcal{A} \mathcal{A}^*)^i \mathcal{P}_{\mathcal{R}(P)} b = (P P^\dagger A G^\dagger A^\top P)^i P P^\dagger b = P P^\dagger (A G^\dagger A^\top P)^i b.$$

The desired result immediately follows. □

Using (4.3), one can also verify that $\text{span}\{\tilde{u}_i\}_{i=1}^k = \mathcal{K}_k(A G^\dagger A^\top P, b)$. Since the dimensions of \mathcal{X} and \mathcal{Y} are $r_G = \text{rank}(G)$ and $r_P = \text{rank}(P)$, respectively, by Proposition 4.2 the gGKB process will eventually terminate in at most $\min\{r_G, r_P\}$ steps. Here “terminate” means that α_i or β_i equals zero at the current step, thereby the Krylov subspaces cannot expand any longer. The “terminate step” can be defined as

$$(4.6) \quad k_t = \min\{k : \alpha_{k+1} \beta_{k+1} = 0\}.$$

Suppose gGKB does not terminate before the k th iteration, i.e., $\alpha_i \beta_i \neq 0$ for $1 \leq i \leq k$. Then the k -step gGKB process generates a G -orthonormal matrix $V_k = (v_1, \dots, v_k) \in \mathbb{R}^{n \times k}$ and a P -orthonormal matrix $U_k = (u_1, \dots, u_k) \in \mathbb{R}^{m \times k}$, which satisfy the relations

$$(4.7) \quad \begin{cases} \beta_1 U_{k+1} e_1 = PP^\dagger b, \\ PP^\dagger AV_k = U_{k+1} B_k, \\ G^\dagger A^\top P U_{k+1} = V_k B_k^\top + \alpha_{k+1} v_{k+1} e_{k+1}^\top, \end{cases}$$

where e_1 and e_{k+1} are the first and $(k+1)$ th columns of I_{k+1} , and

$$(4.8) \quad B_k = \begin{pmatrix} \alpha_1 & & & & \\ \beta_2 & \alpha_2 & & & \\ & \beta_3 & \ddots & & \\ & & \ddots & \alpha_k & \\ & & & \beta_{k+1} & \end{pmatrix} \in \mathbb{R}^{(k+1) \times k}$$

has full column rank. Note that it may happens that $\beta_{k+1} = 0$, which means that gGKB terminates just at the k th step and $u_{k+1} = \mathbf{0}$.

Based on gGKB, we can design an iterative approach for solving (3.6), which will also solve (3.1). Note that under the canonical bases, we can rewrite (3.6) as

$$(4.9) \quad \min_{x \in \mathcal{R}(G)} \|PP^\dagger A - PP^\dagger b\|_P.$$

From $k = 1$ onwards, we seek an approximate solution to (4.9) in the subspace $\text{span}\{V_k\}$. By letting $x = V_k y$ with $y \in \mathbb{R}^k$, we obtain from (4.7) that

$$\begin{aligned} \min_{x \in \text{span}\{V_k\}} \|PP^\dagger Ax - PP^\dagger b\|_P &= \min_{y \in \mathbb{R}^k} \|PP^\dagger AV_k y - \beta_1 U_{k+1} e_1\|_P \\ &= \min_{y \in \mathbb{R}^k} \|U_{k+1} (B_k y - \beta_1 e_1)\|_P = \min_{y \in \mathbb{R}^k} \|B_k y - \beta_1 e_1\|_2, \end{aligned}$$

where we have used that $\{u_i\}$ are P -orthonormal. Note that

$$\operatorname{argmin}_{y \in \mathbb{R}^k} \|B_k y - \beta_1 e_1\|_2 = B_k^\dagger \beta_1 e_1 =: y_k$$

since B_k has full column rank. Therefore, at the k th iteration, the iterative approximation to (4.9) is given by

$$(4.10) \quad x_k = V_k y_k = V_k B_k^\dagger \beta_1 e_1.$$

The above approach is very similar to the LSQR algorithm for the standard LS problem [28]. Moreover, the bidiagonal structure of B_k enables the design of a recursive procedure to update x_k step by step, without explicitly computing $B_k^\dagger \beta_1 e_1$ at each iteration. This procedure is based on the Givens QR factorization of B_k ; see [28, section 4.1] for details. Note that u_i is not required for computing x_k , so there is no need to compute PP^\dagger in gGKB. We summarize the iterative algorithm for solving (4.9) in Algorithm 4.2, which is named the *generalized LSQR* (gLSQR) algorithm.

As the iteration proceeds, the k th solution x_k gradually approximates the true solution of (4.9), and consequently of (3.1). We now state this property precisely.

Algorithm 4.2. Generalized LSQR (gLSQR).

Input: $A \in \mathbb{R}^{m \times n}$, $P \in \mathbb{R}^{m \times m}$, $Q \in \mathbb{R}^{n \times n}$, $b \in \mathbb{R}^m$

- 1: **(Initialization)**
- 2: Compute $\beta_1 \tilde{u}_1 = b$, $\alpha_1 v_1 = G^\dagger A^\top P \tilde{u}_1$
- 3: Set $x_0 = \mathbf{0}$, $w_1 = v_1$, $\bar{\phi}_1 = \beta_1$, $\bar{\rho}_1 = \alpha_1$
- 4: **for** $i = 1, 2, \dots$ until convergence, **do**
- 5: **(Applying the gGKB process)**
- 6: $\beta_{i+1} \tilde{u}_{i+1} = A v_i - \alpha_i \tilde{u}_i$
- 7: $\alpha_{i+1} v_{i+1} = G^\dagger A^\top P \tilde{u}_{i+1} - \beta_{i+1} v_i$
- 8: **(Applying the Givens QR factorization to B_k)**
- 9: $\rho_i = (\bar{\rho}_i^2 + \beta_{i+1}^2)^{1/2}$
- 10: $c_i = \bar{\rho}_i / \rho_i$
- 11: $s_i = \beta_{i+1} / \rho_i$
- 12: $\theta_{i+1} = s_i \alpha_{i+1}$
- 13: $\bar{\rho}_{i+1} = -c_i \alpha_{i+1}$
- 14: $\bar{\phi}_i = c_i \bar{\phi}_i$
- 15: $\bar{\phi}_{i+1} = s_i \bar{\phi}_i$
- 16: **(Updating the solution)**
- 17: $x_i = x_{i-1} + (\bar{\phi}_i / \rho_i) w_i$
- 18: $w_{i+1} = v_{i+1} - (\theta_{i+1} / \rho_i) w_i$
- 19: **end for**

Output: Approximate minimum 2-norm solution of (3.1): x_k

THEOREM 4.3. *Suppose the gGKB process terminates at step k_t . Then x_{k_t} obtained by gLSQR is the exact minimum 2-norm solution of (3.1).*

Proof. Since $x_{k_t} \in \text{span}\{V_k\} \subseteq \mathcal{R}(G)$, by Theorem 3.1, we only need to verify that x_{k_t} satisfies the two conditions in (3.2).

Step 1: Prove $A^\top P(Ax_{k_t} - b) = \mathbf{0}$. By writing x_{k_t} as $x_{k_t} = V_{k_t} y_{k_t}$, we obtain from (4.7) that

$$P(Ax_{k_t} - b) = P\mathcal{P}_{\mathcal{R}(P)}(Ax_{k_t} - b) = PU_{k_t+1}(B_{k_t} y_t - \beta_1 e_1).$$

Using (4.7) again, we get

$$\begin{aligned} G^\dagger A^\top P(Ax_{k_t} - b) &= G^\dagger A^\top PU_{k_t+1}(B_{k_t} y_{k_t} - \beta_1 e_1) \\ &= (V_{k_t} B_{k_t}^\top + \alpha_{k_t+1} v_{k_t+1} e_{k_t+1}^\top) (B_{k_t} y_{k_t} - \beta_1 e_1) \\ &= V_{k_t} (B_{k_t}^\top B_{k_t} y_{k_t} - B_{k_t}^\top \beta_1 e_1) + \alpha_{k_t+1} \beta_{k_t+1} v_{k_t+1} e_{k_t+1}^\top y_{k_t} \\ &= \alpha_{k_t+1} \beta_{k_t+1} v_{k_t+1} e_{k_t+1}^\top y_{k_t} \\ &= \mathbf{0}, \end{aligned}$$

since $\alpha_{k_t+1} \beta_{k_t+1} = 0$ and $B_{k_t}^\top B_{k_t} y_{k_t} = B_{k_t}^\top \beta_1 e_1$ due to $y_{k_t} = \text{argmin}_y \|B_{k_t} y - \beta_1 e_1\|_2$. Note that $x_{k_t} \in \mathcal{R}(G)$. Using the same approach as in the proof (Step 1) of Theorem 3.4, we obtain $A^\top P(Ax_{k_t} - b) = \mathbf{0}$.

Step 2: Prove $x_{k_t}^\top Gz = 0$ for any $z \in \mathcal{N}(A^\top PA)$. By Proposition 4.2, we have $x_{k_t} \in \mathcal{R}(V_{k_t}) \subseteq \mathcal{R}(G^\dagger A^\top P)$. Let $x_{k_t} = G^\dagger A^\top Pw$. Then for any $z \in \mathcal{N}(A^\top PA)$ we have

$$x_{k_t}^\top Gz = (G^\dagger A^\top Pw)^\top Gz = w^\top PAG^\dagger Gz.$$

Recall that $\mathcal{P}_{\mathcal{R}(G)}(\mathcal{N}(A^\top PA)) \subseteq \mathcal{N}(A^\top PA) = \mathcal{N}(L_P A)$, which has been proved in the proof of Theorem 3.1. Thus, $PAG^\dagger Gz = L_P^\top L_P A \mathcal{P}_{\mathcal{R}(G)} z = \mathbf{0}$, leading to $x_{k_t}^\top Gz = \mathbf{0}$. \square

An iterative algorithm should include a stopping criterion to decide whether the current iteration can be stopped to obtain a solution with acceptable approximation accuracy. Notice from Theorem 3.4 that $\mathcal{A}^*(\mathcal{A}x_k - \mathcal{P}_{\mathcal{R}(P)}b) =: \mathcal{A}^*r_k$ would be zero at the iteration where the accurate solution is computed. The scaling invariant quantity $\|\mathcal{A}^*r_k\|_G / (\|\mathcal{A}\| \|\mathcal{P}_{\mathcal{R}(P)}b\|_P)$ can be used to measure the accuracy of the iterative solution, where $\|\mathcal{A}\|$ is the operator norm defined as $\|\mathcal{A}\| := \max_{\substack{v \in \mathcal{R}(G) \\ v \neq \mathbf{0}}} \frac{\|\mathcal{A}v\|_P}{\|v\|_G}$. Thus, we can use

$$(4.11) \quad \frac{\|\mathcal{A}^*r_k\|_G}{\|\mathcal{A}\| \|\mathcal{P}_{\mathcal{R}(P)}b\|_P} \leq \text{tol}$$

as a stopping criterion for gLSQR. We note that (4.11) is analogous to the stopping criterion of LSQR for the standard LS problem $\min_x \|Ax - b\|_2$, which is expressed as $\frac{\|A^\top r_k\|_2}{\|A\|_2 \|r_k\|_2} \leq \text{tol}$; see [28, section 6]. To ensure computational practicality, we discuss how to compute the quantities in (4.11). It is obvious that $\|\mathcal{P}_{\mathcal{R}(P)}b\|_P = (b^\top P b)^{1/2}$. Using Lemma 3.3 and the procedure in the proof of Theorem 4.3, we get

$$\|\mathcal{A}^*r_k\|_G = \|G^\dagger A^\top P \mathcal{P}_{\mathcal{R}(P)}(Ax_k - b)\|_G = \|\alpha_{k+1} \beta_{k+1} v_{k+1} e_k^\top y_k\|_G = \alpha_{k+1} \beta_{k+1} |e_k^\top y_k|,$$

where we have used $\|v_i\|_G = 1$. Therefore, $\|\mathcal{A}^*r_k\|_G$ can be computed quickly with very little additional cost.

To obtain an accurate estimate of $\|\mathcal{A}\|$, here we consider the GLS problem (3.8) with $M = I_m$ for simplicity. Note that any GLS problem can be reduced to this form by substituting $A \leftarrow MA$ and $b \leftarrow Mb$. We give a matrix expression for $\|\mathcal{A}\|$ using the GSVD of $\{A, L\}$.

PROPOSITION 4.4. *Suppose the GSVD of $\{A, L\}$ has the form (3.17). Then we have*

$$(4.12) \quad \|\mathcal{A}\| = \sigma_{\max}(C_A),$$

which is the largest singular value of C_A .

Proof. Using the expression of \mathcal{A} under the canonical bases, we have

$$\|\mathcal{A}\| = \max_{\substack{v \in \mathcal{R}(G) \\ v \neq \mathbf{0}}} \frac{\|\mathcal{A}v\|_2}{\|v\|_G} = \max_{\substack{v \in \mathcal{R}(G) \\ v \neq \mathbf{0}}} \frac{v^\top A^\top A v}{v^\top G v}.$$

In the GSVD of $\{A, L\}$, we use the partition of X as described in (3.19) and denote $\tilde{X}_1 = (X_1 \ X_2 \ X_3)$. In the proof of Theorem 3.10 we have shown that $\mathcal{N}(G) = \mathcal{R}(X_4)$. Now we show $\mathcal{R}(G) = \mathcal{P}_{\mathcal{R}(G)} \mathcal{R}(\tilde{X}_1)$. We only need to show $\dim(\mathcal{P}_{\mathcal{R}(G)} \mathcal{R}(\tilde{X}_1)) = r = \text{rank}(G)$, which is equivalent to $\mathcal{P}_{\mathcal{R}(G)} \tilde{X}_1$ having full column rank. Suppose $\mathcal{P}_{\mathcal{R}(G)} \tilde{X}_1 w = \mathbf{0}$ for a $w \in \mathbb{R}^r$. Then $z := \tilde{X}_1 w \in \mathcal{N}(G)$. Thus, we have $0 = z^\top G z = w^\top \tilde{X}_1^\top G \tilde{X}_1 w = w^\top w$, leading to $w = \mathbf{0}$ and $z = \mathbf{0}$. This proves that $\mathcal{P}_{\mathcal{R}(G)} \tilde{X}_1$ has full column rank.

For any $v \in \mathcal{R}(G)$, write $v = \mathcal{P}_{\mathcal{R}(G)} \tilde{X}_1 y$ with $y \in \mathbb{R}^r$. Then we have

$$v^\top G v = y^\top \tilde{X}_1^\top G G^\dagger G G^\dagger \tilde{X}_1 y = y^\top \tilde{X}_1^\top G \tilde{X}_1 y = \|y\|_2,$$

and

$$Av = A \mathcal{P}_{\mathcal{R}(G)} \tilde{X}_1 y = A \tilde{X}_1 y - A \mathcal{P}_{\mathcal{N}(G)} \tilde{X}_1 y = A \tilde{X}_1 y = U_A C_A y,$$

where we have used $\mathcal{N}(G) \subseteq \mathcal{N}(A)$. It follows that $v^\top A^\top Av = \|U_A C_A y\|_2 = \|C_A y\|_2$. Therefore, we obtain

$$\|\mathcal{A}\| = \max_{\substack{y \in \mathbb{R}^n \\ y \neq 0}} \frac{\|C_A y\|_2}{\|y\|_2} = \|C_A\|_2 = \sigma_{\max}(C_A).$$

The proof is completed. □

Notice that C_A is a diagonal matrix (not necessarily square). Thus, $\sigma_{\max}(C_A)$ is the maximum value of the diagonals of C_A . For a regular matrix pair $\{A, L\}$, i.e., G is nonsingular, several iterative algorithms exist that can rapidly compute the largest generalized singular values of $\{A, L\}$ [18, 36], thereby providing an accurate estimate of $\|\mathcal{A}\|$. For a nonregular $\{A, L\}$, the method proposed in [20] can accomplish the same task.

To the best of our knowledge, gLSQR is the first iterative method designed to solve (3.8) and, consequently, to approximate $A_{ML}^\dagger b$ for any given $b \in \mathbb{R}^n$. A direct method for computing A_{ML}^\dagger is proposed in [8], which relies on the QR factorization and requires a basis for $\mathcal{N}(L)$. This limits its applicability to large-scale problems. In contrast, gLSQR does not require any matrix factorizations. Instead, the main computational bottleneck is the need to approximate $G^\dagger s$ at each iteration. As will be shown in the experiments, the final accuracy of $A_{ML}^\dagger b$ is influenced by the accuracy for approximating $G^\dagger s$.

At the end of this section, we present a diagram in Figure 4.1 to summarize the main ideas and findings of this paper. It illustrates how various concepts related to the LS problem and pseudoinverse have been generalized. Our results reveal the close connections between the GLS problem, weighted pseudoinverse, GSVD, and gLSQR. These insights improve theoretical understanding and offer tools for developing more effective computational methods for related applications.

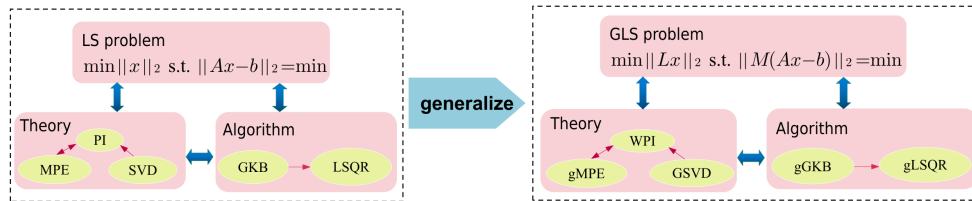


FIG. 4.1. The generalization of several concepts is illustrated as follows: PI and MPE stand for the Moore–Penrose pseudoinverse and Moore–Penrose equations, respectively. WPI and gMPE refer to the weighted pseudoinverse and generalized Moore–Penrose equations, respectively.

5. Numerical experiments. We use several numerical examples to demonstrate the performance of gLSQR for solving GLS problems. All the experiments are performed in MATLAB R2023b using double precision. We note that much of the existing literature on GLS problems is lacking in numerical results, partly because constructing nontrivial test problems, especially for large-scale matrices, is challenging. Based on Theorem 3.1, we construct a test GLS problem using the following steps:

- (1) Choose two matrices $A \in \mathbb{R}^{m \times n}$ and $L \in \mathbb{R}^{p \times n}$, where $m \leq n$. Compute $G = A^\top A + L^\top L$.
- (2) Construct a vector $w \in \mathcal{R}(G)$. Compute a matrix B with columns that form a basis for $\mathcal{N}(A)$. The true solution is constructed as

$$(5.1) \quad x^\dagger = w - B(B^\top GB)^{-1} B^\top Gw.$$

(3) Choose a vector $z \in \mathcal{R}(A)^\perp$. Let the right-hand side vector be $b = Ax^\dagger + z$. Note that (5.1) ensures that x^\dagger satisfies (3.2). According to Theorem 3.1, x^\dagger is the unique minimum 2-norm solution of (3.8) with $M = I_m$. For large-scale matrices, computing (5.1) can be very challenging. Therefore, in our experiments, we only test small- and medium-sized problems.

Experiment 1. The matrix $A \in \mathbb{R}^{2324 \times 4486}$, named `lp_bnl2`, comes from linear programming problems and is sourced from the SuiteSparse Matrix Collection [7]. The matrix $L = L_1$ is defined as the scaled discretization of the first-order differential operator:

$$L_1 = \begin{pmatrix} 1 & -1 & & & \\ & \ddots & \ddots & & \\ & & & 1 & -1 \end{pmatrix} \in \mathbb{R}^{(n-1) \times n}.$$

In this setup, G is positive definite. We construct $w \in \mathbb{R}^n$ by evaluating the function $f(t) = t$ on a uniform grid over the interval $[0, 1]$, i.e., $w(k) = \frac{k-1}{n-1}$ for $k = 1, \dots, n$. To obtain the vector z , we compute the projection of the random vector `randn(m, 1)` onto $\mathcal{R}(A)^\perp$. In this experiment, we directly compute $G^\dagger \bar{s}$ at each iteration of `gGKB` to simulate the exact computation.

The computational results obtained by `gLSQR` are displayed in Figures 5.1 and 5.2. For the residual norm, we use the directly computed quantity, i.e., the left-hand side of (4.11), as the true value and $\alpha_{k+1}\beta_{k+1}|e_k^\top y_k|/(\sigma_{\max}(C_A)\|b\|_2)$ for its estimation.

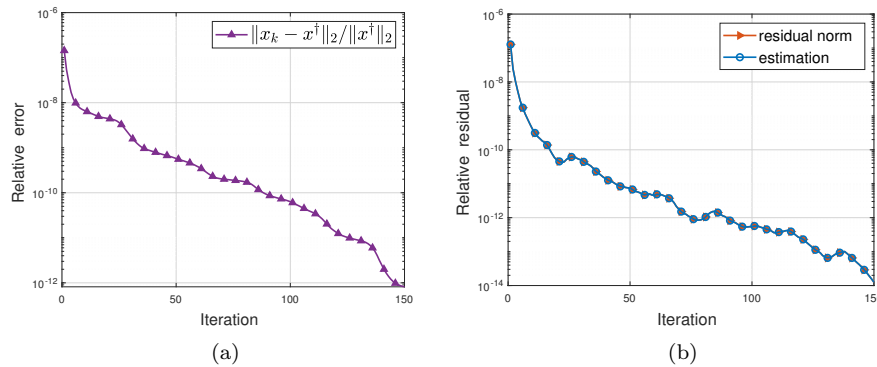


FIG. 5.1. Convergence history of `gLSQR` for the GLS problem with matrices $\{\text{lp_bnl2}, L_1\}$: (a) relative error of iterative solutions; (b) directly computed relative residual norm and its estimation.

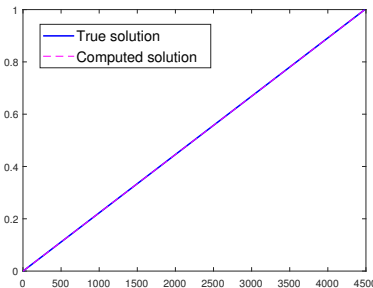


FIG. 5.2. True and computed solutions of the GLS problem with matrices $\{\text{lp_bnl2}, L_1\}$.

These two quantities should be the same if all computations are performed accurately. This can be observed from Figure 5.1(b), where the value gradually decreases at a very low level. The relative error curve shows that x^\dagger is gradually approximated by x_k . We plot the curve corresponding to x_k at the final iteration alongside x^\dagger , which shows that the two solutions match very closely.

Experiment 2. The matrix $A \in \mathbb{R}^{6334 \times 7742}$ named TF15 arising from linear programming problems is taken from [7]. The matrix $L = L_2$ is defined as the scaled discretization of the second-order differential operator:

$$L_2 = \begin{pmatrix} -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \end{pmatrix} \in \mathbb{R}^{(n-2) \times n}.$$

In this set, G is positive definite. We construct $w \in \mathbb{R}^n$ by evaluating the function $f(t) = t^3 - t^2$ on a uniform grid over the interval $[-1, 1]$, i.e., $w(k) = (\frac{2(k-1)}{n-1} - 1)^3 - (\frac{2(k-1)}{n-1} - 1)^2$ for $k = 1, \dots, n$. In this experiment, we directly compute $G^\dagger \bar{s}$ at each iteration of gGKB to simulate the exact computation.

The computational results obtained with gLSQR are presented in Figures 5.3 and 5.4, which are very similar to the first experiment. The results demonstrate the effectiveness of gLSQR in iteratively solving the GLS problem.

Experiment 3. The matrix $A \in \mathbb{R}^{3700 \times 8291}$, named ch and originating from linear programming problems, is taken from [7]. Here, L is set as $L = L_1$. In this set, G

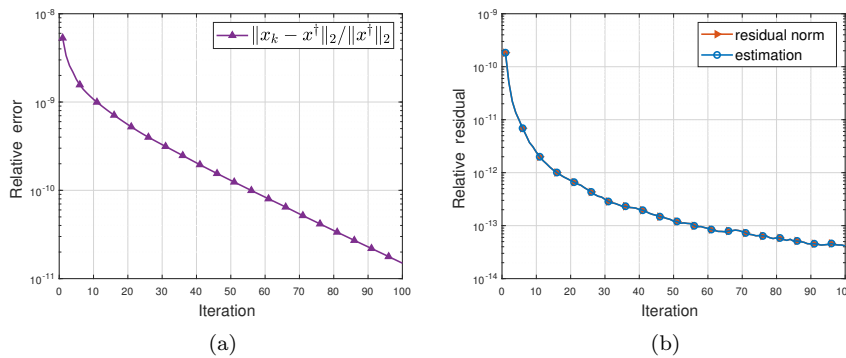


FIG. 5.3. Convergence history of gLSQR for the GLS problem with matrices {TF15, L_2 }: (a) relative error of iterative solutions; (b) directly computed relative residual norm and its estimation.

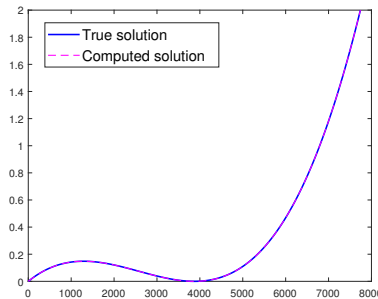


FIG. 5.4. True and computed solutions of the GLS problem with matrices {TF15, L_2 }.

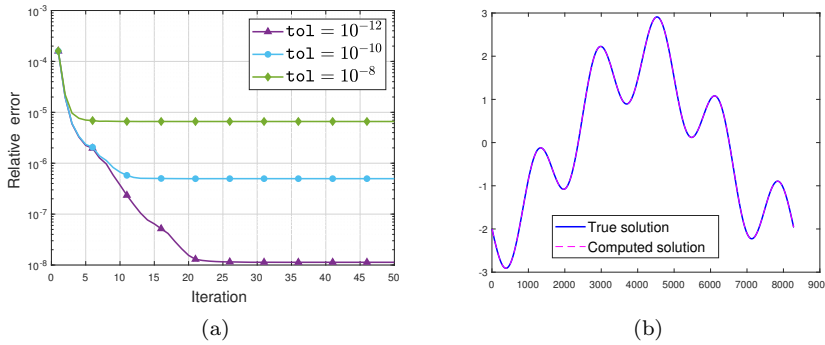


FIG. 5.5. Computed results of the GLS problem with matrices $\{\text{ch}, L_1\}$ by **gLSQR**, where at each iteration we use **LSQR** to approximate $s = G^\dagger \bar{s}$ by solving $\min_s \|Gs - \bar{s}\|_2$ with different stopping tolerance value τ : (a) relative error of iterative solutions with different τ ; (b) true and computed solutions with $\tau = 10^{-8}$.

is positive definite. We construct $w \in \mathbb{R}^n$ by evaluating the function $f(t) = \sin(5t) - 2\cos(t)$ on a uniform grid over the interval $[-\pi, \pi]$, i.e., $w(k) = \sin(\frac{10\pi(k-1)}{n-1} - 5\pi) - 2\cos(\frac{2\pi(k-1)}{n-1} - \pi)$ for $k = 1, \dots, n$.

We use this example to examine how the inaccurate computation of $s = G^\dagger \bar{s}$ affects the numerical behavior of **gLSQR**. At each iteration, we approximately compute $s = G^\dagger \bar{s}$ using the MATLAB built-in function `lsqr.m` to iteratively solve $\min_s \|Gs - \bar{s}\|_2$. The stopping tolerance value τ for `lsqr.m` is set to three different values. From Figure 5.5(a), we observe that the value of τ significantly impacts the final accuracy of x_k , with the accuracy being approximately on the order of $\mathcal{O}(\tau)$. We suspect that the final accuracy may be influenced by both the value of τ and the condition number of G . This should be explored further in future work.

Although constructing a very large-scale test example is difficult, we note that the main computational bottleneck of **gLSQR** is the computation of $G^\dagger \bar{s}$. If a sparse Cholesky factorization of G is available, it can be computed using a direct solver. Otherwise, an iterative solver such as LSQR or conjugate gradient (CG) is the only option. In this case, employing an effective preconditioner is crucial to accelerate the convergence for solving $\min_s \|Gs - \bar{s}\|_2$. Future work will involve constructing more large-scale test problems to evaluate the algorithm and exploring additional theoretical and computational aspects to enhance the performance of **gLSQR**.

6. Conclusion. In this paper, we provide a new interpretation of the weighted pseudoinverse arising from the GLS problem $\min \|Lx\|_2$ such that $\|M(Ax - b)\|_2 = \min$. By introducing the operator $\mathcal{A} : \mathcal{X} = (\mathcal{R}(G), \langle \cdot, \cdot \rangle_G) \rightarrow (\mathcal{R}(P), \langle \cdot, \cdot \rangle_P)$, $v \mapsto \mathcal{P}_{\mathcal{R}(P)} Av$ with $P = M^\top M$ and $G = A^\top PA + L^\top L$, we have shown that the minimum 2-norm solution of the GLS problem is the minimum \mathcal{X} -norm solution of the LS problem $\min_{x \in \mathcal{X}} \|Ax - \mathcal{P}_{\mathcal{R}(P)} b\|_P$. Consequently, the weighted pseudoinverse A_{ML}^\dagger is shown to be equivalent to $A^\dagger \mathcal{P}_{\mathcal{R}(P)}$ under the canonical bases. With this new interpretation of A_{ML}^\dagger , we have derived a set of generalized Moore–Penrose equations that completely characterize the weighted pseudoinverse, and provided a closed-form expression for A_{IL}^\dagger using the GSVD of $\{A, L\}$. We have generalized the GKB process and proposed the **gLSQR** algorithm for iteratively computing $A_{ML}^\dagger b$ for any given $b \in \mathbb{R}^m$, which allows us to compute the minimum 2-norm solution of the GLS problem. Several numerical examples have been used to test the **gLSQR** algorithm, demonstrating that it can compute the solution of a GLS problem with satisfactory accuracy.

The results in this paper suggest that the closely related concepts—GLS, weighted pseudoinverse, GSVD, gGKB, and gLSQR—are appropriate generalizations of the classical concepts LS, pseudoinverse, SVD, GKB, and LSQR. This provides new tools for both analysis and computation of related applications.

Acknowledgments. The author thanks the two anonymous referees and Editor Prof. Miroslav Tůma for their detailed reading of the manuscript and many invaluable comments.

REFERENCES

- [1] A. BEN-ISRAEL AND T. N. GREVILLE, *Generalized Inverses: Theory and Applications*, Springer Science & Business Media, 2006.
- [2] G. CALLON AND C. GROETSCH, *The method of weighting and approximation of restricted pseudosolutions*, *J. Approx. Theory*, 51 (1987), pp. 11–18, [https://doi.org/10.1016/0021-9045\(87\)90090-6](https://doi.org/10.1016/0021-9045(87)90090-6).
- [3] S. L. CAMPBELL AND C. D. MEYER, *Generalized Inverses of Linear Transformations*, *Classics Appl. Math.* 56, SIAM, Philadelphia, 2009, <https://doi.org/10.1137/1.9780898719048>.
- [4] N. A. CARUSO AND P. NOVATI, *Convergence analysis of LSQR for compact operator equations*, *Linear Algebra Appl.*, 583 (2019), pp. 146–164, <https://doi.org/10.1016/j.laa.2019.08.024>.
- [5] J. S. CHIPMAN, *On least squares with insufficient observations*, *J. Amer. Statist. Assoc.*, 59 (1964), pp. 1078–1111, <https://doi.org/10.1080/01621459.1964.10480751>.
- [6] R. E. CLINE AND T. N. E. GREVILLE, *An extension of the generalized inverse of a matrix*, *SIAM J. Appl. Math.*, 19 (1970), pp. 682–688, <https://doi.org/10.1137/0119069>.
- [7] T. A. DAVIS AND Y. HU, *The University of Florida Sparse Matrix Collection*, *ACM Trans. Math. Software*, 38 (2011), 1, <https://doi.org/10.1145/2049662.2049663>.
- [8] L. ELDÉN, *A weighted pseudoinverse, generalized singular values, and constrained least squares problems*, *BIT*, 22 (1982), pp. 487–502, <https://doi.org/10.1007/BF01934412>.
- [9] H. W. ENGL, M. HANKE, AND A. NEUBAUER, *Regularization of Inverse Problems*, *Math. Appl.* 375, Kluwer Academic Publishers, Dordrecht, 1996.
- [10] G. GOLUB AND W. KAHAN, *Calculating the singular values and pseudo-inverse of a matrix*, *J. Soc. Indust. Appl. Math. Ser. B Numer. Anal.*, 2 (1965), pp. 205–224, <https://doi.org/10.1137/0702016>.
- [11] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, 4th ed., The Johns Hopkins University Press, Baltimore, 2013.
- [12] C. W. GROETSCH, *Generalized Inverses of Linear Operators: Representation and Approximation*, *Mongr. Textbooks Pure Appl. Math.* 37, Marcel Dekker, New York, Basel, 1977.
- [13] P. HANSEN AND M. HANKE, *A Lanczos algorithm for computing the largest quotient singular values in regularization problems*, in *SVD and Signal Processing III*, Elsevier, 1995, pp. 131–138.
- [14] P. C. HANSEN, *Rank-deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion*, *SIAM Mongr. Math. Model. Comput.* 4, SIAM, Philadelphia, 1998, <https://doi.org/10.1137/1.9780898719697>.
- [15] P. C. HANSEN, *Discrete Inverse Problems: Insight and Algorithms*, *Fundam. Algorithms* 7, SIAM, Philadelphia, 2010, <https://doi.org/10.1137/1.9780898718836>.
- [16] J. HARTUNG, *A note on restricted pseudoinverses*, *SIAM J. Math. Anal.*, 10 (1979), pp. 266–273, <https://doi.org/10.1137/0510025>.
- [17] S. G. JAMES RAMSAY AND G. HOOKER, *Functional Data Analysis with R and MATLAB*, Springer, New York, 2000.
- [18] Z. JIA AND H. LI, *The joint bidiagonalization method for large GSVD computations in finite precision*, *SIAM J. Matrix Anal. Appl.*, 44 (2023), pp. 382–407, <https://doi.org/10.1137/22M1483608>.
- [19] R. KRESS, *Linear Integral Equations*, Springer, New York, 2013, <https://doi.org/10.1007/978-1-4614-9593-2>.
- [20] H. LI, *Characterizing GSVD by singular value expansion of linear operators and its computation*, *SIAM J. Matrix Anal. Appl.*, 46 (2025), pp. 439–465, <https://doi.org/10.1137/24M1651150>.
- [21] D. W. MARQUARDT, *Generalized inverses, ridge regression, biased linear estimation, and nonlinear estimation*, *Technometrics*, 12 (1970), pp. 591–612, <https://doi.org/10.1080/00401706.1970.10488699>.

- [22] R. D. MILNE, *An oblique matrix pseudoinverse*, SIAM J. Appl. Math., 16 (1968), pp. 931–944, <https://doi.org/10.1137/0116075>.
- [23] N. MINAMIDE AND K. NAKAMURA, *A restricted pseudoinverse and its application to constrained minima*, SIAM J. Appl. Math., 19 (1970), pp. 167–177, <https://doi.org/10.1137/0119015>.
- [24] S. K. MITRA AND C. R. RAO, *Projections under seminorms and generalized Moore Penrose inverses*, Linear Algebra Appl., 9 (1974), pp. 155–167, [https://doi.org/10.1016/0024-3795\(74\)90034-2](https://doi.org/10.1016/0024-3795(74)90034-2).
- [25] E. H. MOORE, *On the reciprocal of the general algebraic matrix*, Bull. Amer. Math. Soc., 26 (1920), pp. 294–295.
- [26] E. H. MOORE, *General Analysis*, The American Philosophical Society, Philadelphia, 1935.
- [27] C. C. PAIGE AND M. A. SAUNDERS, *Towards a generalized singular value decomposition*, SIAM J. Numer. Anal., 18 (1981), pp. 398–405, <https://doi.org/10.1137/0718026>.
- [28] C. C. PAIGE AND M. A. SAUNDERS, *LSQR: An algorithm for sparse linear equations and sparse least squares*, ACM Trans. Math. Software, 8 (1982), pp. 43–71, <https://doi.org/10.1145/355984.355989>.
- [29] R. PENROSE, *A generalized inverse for matrices*, Math. Proc. Cambridge Philos. Soc., 51 (1955), pp. 406–413.
- [30] R. PENROSE, *On best approximate solutions of linear matrix equations*, Math. Proc. Cambridge Philos. Soc., 52 (1956), pp. 17–19.
- [31] C. M. PRICE, *The matrix pseudoinverse and minimal variance estimates*, SIAM Rev., 6 (1964), pp. 115–120, <https://doi.org/10.1137/1006029>.
- [32] J. F. WARD, T. L. BOULLION, AND T. O. LEWIS, *Weighted pseudoinverses with singular weights*, SIAM J. Appl. Math., 21 (1971), pp. 480–482, <https://doi.org/10.1137/0121051>.
- [33] J. F. WARD, JR., *On a limit formula for weighted pseudoinverses*, SIAM J. Appl. Math., 33 (1977), pp. 34–38, <https://doi.org/10.1137/0133003>.
- [34] V. WEISS, L. ANDOR, G. RENNER, AND T. VÁRADY, *Advanced surface fitting techniques*, Comput. Aided Geom. Design, 19 (2002), pp. 19–42, [https://doi.org/10.1016/S0167-8396\(01\)00086-3](https://doi.org/10.1016/S0167-8396(01)00086-3).
- [35] H. WENDLAND, *Scattered Data Approximation*, Cambridge Monogr. Appl. Comput. Math. 17, Cambridge University Press, 2004.
- [36] H. ZHA, *Computing the generalized singular values/vectors of large sparse or structured matrix pairs*, Numer. Math., 72 (1996), pp. 391–417, <https://doi.org/10.1007/s002110050175>.