mathematics

MDPI

Article

Conditioning Theory for Generalized Inverse $C_A^{\ddagger}$ and Their Estimations

Mahvish Samar [1,*], Xinzhong Zhu [1] and Abdul Shakoor [2]

1 College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua 321004, China;
2 Department of Mathematics, The Islamia University of Bahawalpur, Bahawalpur 63100, Pakistan;
* Correspondence: mahvishsamar@hotmail.com

Abstract: The conditioning theory of the generalized inverse $C_A^{\ddagger}$ is considered in this article. First, we introduce three kinds of condition numbers for the generalized inverse $C_A^{\ddagger}$, i.e., normwise, mixed and componentwise ones, and present their explicit expressions. Then, using the intermediate result, which is the derivative of $C_A^{\ddagger}$, we can recover the explicit condition number expressions for the solution of the equality constrained indefinite least squares problem. Furthermore, using the augment system, we investigate the componentwise perturbation analysis of the solution and residual of the equality constrained indefinite least squares problem. To estimate these condition numbers with high reliability, we choose the probabilistic spectral norm estimator to devise the first algorithm and the small-sample statistical condition estimation method for the other two algorithms. In the end, the numerical examples illuminate the obtained results.

Keywords: generalized inverse $C_A^{\ddagger}$; normwise condition number; mixed and componentwise condition numbers; EILS problem; probabilistic spectral norm estimator; small-sample statistical condition estimation

MSC: 65F20; 65F35; 65F30; 15A12; 15A60

Citation: Samar, M.; Zhu, X.; Shakoor, A. Conditioning Theory for Generalized Inverse $C_A^{\ddagger}$ and Their Estimations. *Mathematics* **2023**, *11*, 0. https://doi.org/

Academic Editor: Adolfo Ballester-Bolinches

Received: 24 March 2023
Revised: 25 April 2023
Accepted: 25 April 2023
Published:

1. Introduction

Throughout this paper, $\mathbb{R}^{m\times n}$ denotes the set of real $m \times n$ matrices. For a matrix $A \in \mathbb{R}^{m\times n}$, A^T is the transpose of A, $\text{rank}(A)$ denotes the rank of A, $\|A\|_2$ is the spectral norm of A, and $\|A\|_F$ is the Frobenius norm of A. For a vector a ,$\|a\|_\infty$ is its ∞-norm, and $\|a\|_2$ the 2-norm. The notation $|A|$ is a matrix whose components are the absolute values of the corresponding components of A. For any matrix A, the following four equations uniquely define the Moore–Penrose inverse $A^\dagger$ of A [1]:

$$AA^\dagger A = A,\ A^\dagger AA^\dagger = A^\dagger,\ (AA^\dagger)^T = AA^\dagger,\ (A^\dagger A)^T = A^\dagger A. \tag{1}$$

The generalized inverse $C_A^{\ddagger}$ is defined by

$$C_A^{\ddagger} = (I - (PQP)^\dagger Q)C^\dagger, \tag{2}$$

where $Q = A^T JA$, $A \in \mathbb{R}^{(p+q)\times n}$ denotes weight matrix and $P = I - C^\dagger C$ is the orthogonal projection onto the null space of C and $C \in \mathbb{R}^{s\times n}$ may not have full rank and J is a signature matrix defined by

$$J = \begin{bmatrix} I_p & 0 \\ 0 & -I_q \end{bmatrix}, \quad p + q = m.$$

The generalized inverse $C_A^{\ddagger}$ originated from the equality constrained indefinite least squares problem (EILS), which is stated as follows [2–5]:

$$\text{EILS}: \quad \min_{\|Cx-h\|_2} (g - Ax)^T J(g - Ax), \tag{3}$$

where $g \in \mathbb{R}^m$ and $h \in \mathbb{R}^s$. The EILS problem has a unique solution:

$$x = C_A^{\ddagger} h + (PQP)^{\dagger} A^T J g \tag{4}$$

under the following condition:

$$\text{rank}(C) = s, \quad x^T Q x > 0 \quad \text{for all nonzero} \quad x \in \text{null}(C).$$

The above condition implies

$$p \geq n - s, \quad \text{rank}\begin{pmatrix} A \\ C \end{pmatrix} = n, \tag{5}$$

then (5) ensures the existence and uniqueness of generalized inverse $C_A^{\ddagger}$ (see [2,6]). The generalized inverse $C_A^{\ddagger}$ has significant applications in the study of EILS algorithms, the analysis of large-scale structure, error analysis, perturbation theory, and the solution of the EILS problem [2–5,7–10]. The EILS problem was first demonstrated by Bojanczyk et al. [5]. Additionally, we reveal some detailed work on the perturbation analysis of this problem. The perturbation theory of the EILS problem was discussed by Wang [11] and extended by Shi and Liu [8] based on the hyperbolic MGS elimination method. Diao and Zhou [12] recovered the linearized estimate of the backward error of this problem. Later, Li et al. [13] investigated the componentwise condition numbers for the EILS problem. Recently, Wang and Meng [14] studied the condition numbers and normwise perturbation analysis of the EILS problem.

Componentwise perturbation analysis has received significant attention in recent years; for references, see [15–19]. The motivation for studying componentwise perturbation analysis is reasonable for research because, if the perturbation in the input data is measured componentwise rather than by norm, it may help us to measure the sensitivity of a function more accurately [15], and improve the exactness and effectiveness of the EILS solution computation. It has attracted many authors' attention to consider the componentwise perturbation analysis in which the least squares problem [16] and the weighted least squares problem [17] are included. In this article, we continue the research on componentwise perturbation analysis of the EILS problem. We can recover the componentwise perturbation bounds of the indefinite least squares problem with the intermediate result.

The generalized inverse $C_A^{\ddagger}$ reduce to K-weighted pseudoinverse $L_K^{\dagger}$ when $q = 0$ and K has a full row rank. This pseudoinverse was expanded to the MK-weighted pseudoinverse $L_{MK}^{\dagger}$ by Wei and Zhang [6], which describes its structure and uniqueness. Its algorithm was developed by Elden [20]. According to Wei [21], the expression of $L_K^{\dagger}$ based on GSVD was investigated. A perturbation equation for $L_K^{\dagger}$ was given by Gulliksson et al. [22]. The condition numbers for the K-weighted pseudoinverse $L_K^{\dagger}$ and their statistical estimate were recently provided by Mahvish et al. [23].

The condition number is a well-known research topic in numerical analysis that estimates the worst-case sensitivity of input data to small perturbations on it (see [24–26] and references therein). The normwise condition number [25] has the disadvantage of disregarding the scaling structure of both input and output data. To address this issue, the terms mixed and componentwise condition numbers are introduced [26]. Mixed condition numbers employ componentwise error analysis for input data and normwise error analysis for output data. On the other hand, the componentwise condition numbers employ componentwise error analysis for input and output data. In fact, due to rounding errors and data storage difficulties, it is more practical to estimate input errors componentwise rather than normwise. However, the condition numbers of the generalized inverse $C_A^{\ddagger}$ have not been discussed until now. Inspired by this, we attempt to present the explicit expressions of normwise, mixed and componentwise condition numbers for the generalized inverse $C_A^{\ddagger}$, as well as their statistical estimation due to their importance in EILS research.

The rest of this manuscript is organized as follows: Section 2 provides some preliminaries that will be helpful for the upcoming discussions. With the intermediate result, i.e., the derivative of $C_A^{\ddagger}$, we can recover the explicit expression of condition numbers for the solution of the EILS problem in Section 3. Section 4 will present the componentwise perturbation analysis for the EILS problem. In Section 5, we propose the first two algorithms for the normwise condition number by using the probabilistic spectral norm estimator [27] and the small-sample statistical condition estimation [28] method. Additionally, we construct the third algorithm for the mixed and componentwise condition numbers by using the small-sample statistical condition estimation [28] method. To check the efficiency of these algorithms, we demonstrate them through numerical experiments in Section 6.

2. Preliminaries

In this part, we introduce some definitions and important results, which will be used in the upcoming sections.

Firstly, we define the entrywise division between two vectors $v = [v_1, \ldots, v_p]^{\mathrm{T}} \in \mathbb{R}^p$ and $w = [w_1, \ldots, w_p]^{\mathrm{T}} \in \mathbb{R}^p$ by $\frac{v}{w} = [\eta_1, \ldots, \eta_p]^{\mathrm{T}}$ with

$$\eta_i = \begin{cases} \frac{v_i}{w_i}, & \text{if } w_i \neq 0 \\ v_i, & \text{if } w_i = 0 \end{cases}.$$

Following [1,26,29], the componentwise distance between v and w is defined by

$$d(v, w) = \left\| \frac{v - w}{w} \right\|_\infty = \max_{i=1,\ldots,p} \left\{ \frac{|v_i - w_i|}{|w_i|} \right\} = \begin{cases} \frac{|v_{i*} - w_{i*}|}{|w_{i*}|}, & \text{if } w_{i*} \neq 0 \\ |v_{i*}|, & \text{if } w_{i*} = 0 \end{cases}$$

Note that when $w_{i*} \neq 0, \forall\ i = 1, \ldots, p$, $d(v, w)$ gives the relative distance from v to w with respect to w, while the absolute distance for $w_{i*} = 0$. We describe the distance between the matrices $V, W \in \mathbb{R}^{n \times n}$ as follows:

$$d(V, W) = d(\mathrm{vec}(V), \mathrm{vec}(W)).$$

In order to define the mixed and componentwise condition numbers, we also need to define the set $B^\circ(v, \varepsilon) = \{u \in \mathbb{R}^p |\ |u_i - v_i| \leq \epsilon |v_i|,\quad i = 1, \cdots, p\}$ and $B(v, \varepsilon) = \{u \in \mathbb{R}^p \mid \|u - v\|_2 \leqslant \varepsilon \|v\|_2\}$ for given $\varepsilon > 0$.

Definition 1 ([29])**.** *Let $\aleph : \mathbb{R}^p \to \mathbb{R}^q$ be a continuous mapping defined on an open set* $\mathrm{Dom}(\aleph) \subset \mathbb{R}^p$*, and $v \in \mathrm{Dom}(\aleph), v \neq 0$ such that $\aleph(v) \neq 0$.*

(i) The normwise condition number of $\aleph$ at v is given by

$$n(\aleph, v) \quad = \quad \lim_{\varepsilon \to 0} \sup_{\substack{u \in B(v,\varepsilon) \\ u \neq v}} \left(\frac{\|\aleph(u) - \aleph(v)\|_2}{\|\aleph(v)\|_2} \Big/ \frac{\|u - v\|_2}{\|v\|_2} \right).$$

(ii) The mixed condition number of $\aleph$ at v is given by

$$m(\aleph, v) \quad = \quad \lim_{\varepsilon \to 0} \sup_{\substack{u \in B^o(v,\varepsilon) \\ u \neq v}} \frac{\|\aleph(u) - \aleph(v)\|_\infty}{\|\aleph(v)\|_\infty} \frac{1}{d(u, v)}.$$

(iii) *The componentwise condition number of* $\aleph$ *at* v *is given by*

$$c(\aleph, v) = \lim_{\varepsilon \to 0} \sup_{\substack{u \in B^o(v,\varepsilon) \\ u \neq v}} \frac{d(\aleph(u), \aleph(v))}{d(u,v)}.$$

When the map $\aleph$ in Definition 1 is Fréchet differentiable, the following lemma given in [29] makes the computation of condition numbers easier.

Lemma 1 ([29])**.** *Under the assumptions of Definition 1, and supposing* $\aleph$ *is Fréchet differentiable at* v*, we have*

$$n(\aleph, v) = \frac{\|\mathrm{d}\aleph(v)\|_2 \|v\|_2}{\|\aleph(v)\|_2},\ m(\aleph, v) = \frac{\||\mathrm{d}\aleph(v)||v|\|_\infty}{\|\aleph(v)\|_\infty},\ c(\aleph, v) = \left\| \frac{|\mathrm{d}\aleph(v)||v|}{|\aleph(v)|} \right\|_\infty,$$

where $\mathrm{d}\aleph(v)$ *stands for the Fréchet derivative of* $\aleph$ *at* v.

To obtain the explicit expressions of the above condition numbers, we need some properties of the Kronecker product [30] between X and Y:

$$\mathrm{vec}(YZX) = \left(X^T \otimes Y\right)\mathrm{vec}(Z), \tag{6}$$

$$\mathrm{vec}\left(Y^T\right) = \Pi_{mn}\mathrm{vec}(Y), \tag{7}$$

$$\|Y \otimes X\|_2 = \|Y\|_2 \|X\|_2, \tag{8}$$

where the matrix Z has a suitable dimension, and $\Pi_{mn} \in \mathbb{R}^{mn \times mn}$ is the vec-permutation matrix, which depends only on the dimensions m and n.

Now, we present the following two lemmas, which will be helpful for obtaining condition numbers and their upper bounds.

Lemma 2 ([31], p. 174, Theorem 5)**.** *Let* S *be an open subset of* $\mathbb{R}^{n \times q}$*, and let* $\aleph : S \longrightarrow \mathbb{R}^{m \times p}$ *be a matrix function defined and* $k \geq 1$ *times (continuously) differentiable on* S*. If* $\mathrm{rank}(\aleph(X))$ *is constant on* S*, then* $\aleph^\dagger : S \longrightarrow \mathbb{R}^{p \times m}$ *is* k *times (continuously) differentiable on* S*, and*

$$\mathrm{d}\aleph^\dagger = -\aleph^\dagger \mathrm{d}\aleph \aleph^\dagger + \aleph^\dagger \aleph^{\dagger^T} \mathrm{d}\aleph^T (I_m - \aleph\aleph^\dagger) + (I_p - \aleph^\dagger\aleph)\mathrm{d}\aleph^T \aleph^{\dagger^T} \aleph^\dagger. \tag{9}$$

Lemma 3 ([1])**.** *For any matrices* E*,* F*,* G*,* H*,* U *and* V *with dimensions making the following well defined*

$$[E \otimes F + (G \otimes H)\Pi]\mathrm{vec}(U),$$
$$\frac{[E \otimes F + (G \otimes H)\Pi]\mathrm{vec}(U)}{V},$$
$$FUE^T \text{and } HU^T G^T,$$

we have

$$\||[E \otimes F + (G \otimes H)\Pi]|\mathrm{vec}(|U|)\|_\infty \leq \left\|\mathrm{vec}(|F||U||E|^T + |H||U|^T|G|^T)\right\|_\infty$$

and

$$\left\| \frac{|[E \otimes F + (G \otimes H)\Pi]|\mathrm{vec}(|U|)}{|V|} \right\|_\infty \leq \left\| \frac{\mathrm{vec}(|F||U||E|^T + |H||U|^T|G|^T)}{|V|} \right\|_\infty.$$

3. Condition Numbers

First, we define a mapping $\phi(u) : \mathbb{R}^{mn+sn} \to \mathbb{R}^{ns}$ by

$$\phi(u) = \mathrm{vec}(C_A^\ddagger). \tag{10}$$

Here, $u = (\mathrm{vec}(A)^T, \mathrm{vec}(C)^T)^T$, $\Delta u = (\mathrm{vec}(\Delta A)^T, \mathrm{vec}(\Delta C)^T)^T$, and for matrix $X = (x_{ij})$, $\|X\|_F = \|\mathrm{vec}(X)\|_2$ and $\|X\|_{\max} = \|\mathrm{vec}(X)\|_\infty = \max_{i,j} |x_{ij}|$.

Then, using Definition 1, we present the definitions of the normwise, mixed, and componentwise condition numbers for generalized inverse $C_A^\ddagger$ as given in [32]:

$$n^\ddagger(A,C) = n(\phi,u) := \lim_{\varepsilon\to 0} \sup_{\|[\Delta A,\, \Delta C]\|_F \le \varepsilon \|[A,\, C]\|_F} \frac{\|(C+\Delta C)_A^\ddagger - C_A^\ddagger\|_F / \|C_A^\ddagger\|_F}{\|[\Delta A,\, \Delta C]\|_F / \|[A,\, C]\|_F}, \tag{11}$$

$$m^\ddagger(A,C) = m(\phi,u) := \lim_{\varepsilon\to 0} \sup_{\substack{\|\Delta A/A\|_{\max}\le\varepsilon \\ \|\Delta C/C\|_{\max}\le\varepsilon}} \frac{\|(C+\Delta C)_A^\ddagger - C_A^\ddagger\|_{\max}}{\|C_A^\ddagger\|_{\max}} \frac{1}{d(u+\Delta u, u)}, \tag{12}$$

$$c^\ddagger(A,C) = c(\phi,u) := \lim_{\varepsilon\to 0} \sup_{\substack{\|\Delta A/A\|_{\max}\le\varepsilon \\ \|\Delta C/C\|_{\max}\le\varepsilon}} \frac{1}{d(u+\Delta u, u)} \left\| \frac{(C+\Delta C)_A^\ddagger - C_A^\ddagger}{C_A^\ddagger} \right\|_{\max}. \tag{13}$$

With the help of the vec operator, Frobenius, spectral, and Max norms, we can rewrite the definitions of normwise, mixed and componentwise condition numbers as follows:

$$n^\ddagger(A,C) = n(\phi,u) := \lim_{\varepsilon\to 0} \sup_{\left\| \begin{bmatrix} \mathrm{vec}(\Delta A) \\ \mathrm{vec}(\Delta C) \end{bmatrix} \right\|_2 \le \varepsilon \left\| \begin{bmatrix} \mathrm{vec}(A) \\ \mathrm{vec}(C) \end{bmatrix} \right\|_2} \frac{\|\mathrm{vec}((C+\Delta C)_A^\ddagger - C_A^\ddagger)\|_2}{\|\mathrm{vec}(C_A^\ddagger)\|_2} \Bigg/ \frac{\left\| \begin{bmatrix} \mathrm{vec}(\Delta A) \\ \mathrm{vec}(\Delta C) \end{bmatrix} \right\|_2}{\left\| \begin{bmatrix} \mathrm{vec}(A) \\ \mathrm{vec}(C) \end{bmatrix} \right\|_2}, \tag{14}$$

$$m^\ddagger(A,C) = m(\phi,u) := \lim_{\varepsilon\to 0} \sup_{\substack{\|\mathrm{vec}(\Delta A)/\mathrm{vec}(A)\|_\infty\le\varepsilon \\ \|\mathrm{vec}(\Delta C)/\mathrm{vec}(C)\|_\infty\le\varepsilon}} \frac{\|\mathrm{vec}((C+\Delta C)_A^\ddagger - C_A^\ddagger)\|_\infty}{\|\mathrm{vec}(C_A^\ddagger)\|_\infty} \frac{1}{d(u+\Delta u, u)}, \tag{15}$$

$$c^\ddagger(A,C) = c(\phi,u) := \lim_{\varepsilon\to 0} \sup_{\substack{\|\mathrm{vec}(\Delta A)/\mathrm{vec}(A)\|_\infty\le\varepsilon \\ \|\mathrm{vec}(\Delta C)/\mathrm{vec}(C)\|_\infty\le\varepsilon}} \frac{1}{d(u+\Delta u, u)} \left\| \frac{\mathrm{vec}((C+\Delta C)_A^\ddagger - C_A^\ddagger)}{\mathrm{vec}(C_A^\ddagger)} \right\|_\infty. \tag{16}$$

In the following, we find the expression of the Fréchet derivative of ϕ at u.

Lemma 4. *Let the mapping ϕ be continuous. Then, the Fréchet differential at u is:*

$$\phi'(u) = [W(A), W(C)], \tag{17}$$

where

$$\begin{aligned} W(A) &= -[(C_A^{\ddagger^T} \otimes (PQP)^\dagger A^T J) + ((JAC_A^\ddagger)^T \otimes (PQP)^\dagger)\Pi_{mn}], \\ W(C) &= -[(C_A^{\ddagger^T} \otimes C_A^\ddagger) - ((I - CC^\dagger)^T \otimes C_A^\ddagger C^{\dagger^T})\Pi_{sn} - (C^{\dagger^T} Q C_A^\ddagger)^T \otimes (PQP)^\dagger)\Pi_{sn}]. \end{aligned} \tag{18}$$

Proof. Differentiating both sides of (2), we obtain

$$\mathrm{d}(C_A^\ddagger) \;=\; \mathrm{d}[(I - (PQP)^\dagger Q)C^\dagger]. \tag{19}$$

From ([3], Theorem 2.2), we obtain

$$(PQP)^\dagger = P(PQP)^\dagger = (PQP)^\dagger P = P(PQP)^\dagger P, \tag{20}$$

$$P(I - (PQP)^\dagger QP) = 0, \;\; (PQP)^\dagger QP = P. \tag{21}$$

Thus, substituting (20) into (19) and differentiating both sides of the equation, we can deduce

$$\begin{aligned}\mathrm{d}(C_A^{\ddagger}) &= \mathrm{d}[(I-P(PQP)^{\dagger}Q)C^{\dagger}] = \mathrm{d}C^{\dagger} - \mathrm{d}(P(PQP)^{\dagger}QC^{\dagger}) \\ &= (I-P(PQP)^{\dagger}Q)\mathrm{d}C^{\dagger} - \mathrm{d}P(PQP)^{\dagger}QC^{\dagger} - P\mathrm{d}(PQP)^{\dagger}QC^{\dagger} - P(PQP)^{\dagger}\mathrm{d}QC^{\dagger}.\end{aligned}$$

Further, using (9), we have

$$\begin{aligned}\mathrm{d}(C_A^{\ddagger}) = {} & (I-P(PQP)^{\dagger}Q)[-C^{\dagger}\mathrm{d}CC^{\dagger} + C^{\dagger}C^{\dagger^T}\mathrm{d}C^T(I-CC^{\dagger}) + (I-C^{\dagger}C)\mathrm{d}C^TC^{\dagger^T}C^{\dagger}] \\ & - \mathrm{d}(I-C^{\dagger}C)(PQP)^{\dagger}QC^{\dagger} + P[(PQP)^{\dagger}\mathrm{d}(PQP)(PQP)^{\dagger} \\ & - (PQP)^{\dagger}(PQP)^{\dagger^T}\mathrm{d}(PQP)^T(I-(PQP)(PQP)^{\dagger}) \\ & - (I-(PQP)^{\dagger}(PQP))\mathrm{d}(PQP)^T(PQP)^{\dagger^T}(PQP)^{\dagger}]QC^{\dagger} - P(PQP)^{\dagger}\mathrm{d}QC^{\dagger}.\end{aligned}$$

Noting (20), (2), and $(I-P(PQP)^{\dagger}Q)(I-C^{\dagger}C) = P(I-(PQP)^{\dagger}QP)$, the previous equation may be expressed as

$$\begin{aligned}\mathrm{d}(C_A^{\ddagger}) = {} & -C_A^{\ddagger}\mathrm{d}CC^{\dagger} + C_A^{\ddagger}C^{\dagger^T}\mathrm{d}C^T(I-CC^{\dagger}) + P(I-(PQP)^{\dagger}QP)\mathrm{d}C^TC^{\dagger^T}C^{\dagger} + \mathrm{d}C^{\dagger}C(PQP)^{\dagger}QC^{\dagger} \\ & + C^{\dagger}\mathrm{d}C(PQP)^{\dagger}QC^{\dagger} + (PQP)^{\dagger}\mathrm{d}(PQP)(PQP)^{\dagger}QC^{\dagger} \\ & - (PQP)^{\dagger}(PQP)^{\dagger^T}\mathrm{d}(PQP)^T(I-(PQP)(PQP)^{\dagger})QC^{\dagger} \\ & - P(I-(PQP)^{\dagger}QP)\mathrm{d}(PQP)^T(PQP)^{\dagger^T}(PQP)^{\dagger}QC^{\dagger} - (PQP)^{\dagger}\mathrm{d}QC^{\dagger}.\end{aligned}$$

Further, by the fact $PQ = (QP)^T = QP$, (21), and

$$CP(PQP)^{\dagger} = C(PQP)^{\dagger} = 0, \tag{22}$$

the above equation may be simplified as follows:

$$\begin{aligned}\mathrm{d}(C_A^{\ddagger}) = {} & -C_K^{\ddagger}\mathrm{d}CC^{\dagger} + C_A^{\ddagger}C^{\dagger^T}\mathrm{d}C^T(I-CC^{\dagger}) + C^{\dagger}\mathrm{d}C(PQP)^{\dagger}QC^{\dagger} - (PQP)^{\dagger}\mathrm{d}QC^{\dagger} \\ & + (PQP)^{\dagger}\mathrm{d}QP(PQP)^{\dagger}QC^{\dagger} + (PQP)^{\dagger}Q\mathrm{d}P(PQP)^{\dagger}QC^{\dagger} \\ & - (PQP)^{\dagger}(PQP)^{\dagger^T}\mathrm{d}Q^TP^T(I-(PQP)(PQP)^{\dagger})QC^{\dagger} \\ & - (PQP)^{\dagger}(PQP)^{\dagger^T}Q^T\mathrm{d}P^T(I-(PQP)(PQP)^{\dagger})QC^{\dagger}. \\ = {} & -C_K^{\ddagger}\mathrm{d}CC^{\dagger} + C_A^{\ddagger}C^{\dagger^T}\mathrm{d}C^T(I-CC^{\dagger}) + C^{\dagger}\mathrm{d}C(PQP)^{\dagger}QC^{\dagger} - (PQP)^{\dagger}\mathrm{d}A^TJAC^{\dagger} \\ & - (PQP)^{\dagger}A^TJ\mathrm{d}AC^{\dagger} + (PQP)^{\dagger}P\mathrm{d}A^TJA(PQP)^{\dagger}QC^{\dagger} + (PQP)^{\dagger}PA^TJ\mathrm{d}A(PQP)^{\dagger}QC^{\dagger} \\ & - (PQP)^{\dagger}(PQP)^{\dagger^T}\mathrm{d}Q^TP(I-QP(PQP)^{\dagger})QC^{\dagger} + (PQP)^{\dagger}Q\mathrm{d}P(PQP)^{\dagger}QC^{\dagger} \\ & - (PQP)^{\dagger}(PQP)(PQP)^{\dagger}\mathrm{d}P^T(I-QP(PQP)^{\dagger})QC^{\dagger}.\end{aligned} \tag{23}$$

Considering $PQ = (QP)^T = QP$, we obtain

$$P((I-QP(PQP)^{\dagger}) = 0 \quad QP(PQP)^{\dagger} = P \tag{24}$$

Substituting this fact into (23) implies

$$\begin{aligned}\mathrm{d}(C_A^{\ddagger}) = {} & -C_A^{\ddagger}\mathrm{d}CC^{\dagger} + C_A^{\ddagger}C^{\dagger^T}\mathrm{d}C^T(I-CC^{\dagger}) + C^{\dagger}\mathrm{d}C(PQP)^{\dagger}QC^{\dagger} \\ & - (PQP)^{\dagger}A^TJ\mathrm{d}A(I-P(PQP)^{\dagger}Q)C^{\dagger} - (PQP)^{\dagger}QC^{\dagger}\mathrm{d}C(PQP)^{\dagger}QC^{\dagger} \\ & - (PQP)^{\dagger}\mathrm{d}A^TJA(I-(PQP)^{\dagger}Q)C^{\dagger} + (PQP)^{\dagger}\mathrm{d}C^TC^{\dagger^T}Q(I-(PQP)^{\dagger}Q)C^{\dagger}.\end{aligned}$$

We can rewrite the above equation by using (2) and (20) as

$$
\begin{aligned}
\mathrm{d}(C_A^{\ddagger}) = & -C_A^{\ddagger}\mathrm{d}CC_A^{\ddagger} + C_A^{\ddagger}C^{\dagger^T}\mathrm{d}C^T(I - CC^{\dagger}) + (PQP)^{\dagger}\mathrm{d}C^T C^{\dagger^T} QC_A^{\ddagger} - (PQP)^{\dagger}A^T J\mathrm{d}AC_A^{\ddagger} \\
& - (PQP)^{\dagger}\mathrm{d}A^T JAC_A^{\ddagger}.
\end{aligned} \tag{25}
$$

By applying "vec" operator on (25), and using (6) and (7), we obtain

$$
\begin{aligned}
\mathrm{vec}(\mathrm{d}(C_A^{\ddagger})) = & -(C_A^{\ddagger^T} \otimes (PQP)^{\dagger}A^T J)\mathrm{vec}(\mathrm{d}A) - ((JAC_A^{\ddagger})^T \otimes (PQP)^{\dagger})\mathrm{vec}(\mathrm{d}A^T) \\
& - (C_A^{\ddagger^T} \otimes C_A^{\ddagger})\mathrm{vec}(\mathrm{d}C) + ((I - CC^{\dagger})^T \otimes C_A^{\ddagger}C^{\dagger^T})\mathrm{vec}(\mathrm{d}C^T) \\
& + ((C^{\dagger^T} QC_A^{\ddagger})^T \otimes (PQP)^{\dagger})\mathrm{vec}(\mathrm{d}C^T) \quad \text{by (6)} \\
= & -[(C_A^{\ddagger^T} \otimes (PQP)^{\dagger}A^T J) + ((JAC_A^{\ddagger})^T \otimes (PQP)^{\dagger})\Pi_{mn}]\mathrm{vec}(\mathrm{d}A) \\
& - [(C_A^{\ddagger^T} \otimes C_A^{\ddagger}) - ((I - CC^{\dagger})^T \otimes C_A^{\ddagger}C^{\dagger^T})\Pi_{sn} - (C^{\dagger^T} QC_A^{\ddagger})^T \otimes (PQP)^{\dagger})\Pi_{sn}]\mathrm{vec}(\mathrm{d}C) \\
& \quad \text{by (7)} \\
= & [-(C_A^{\ddagger^T} \otimes (PQP)^{\dagger}A^T J) - ((JAC_A^{\ddagger})^T \otimes (PQP)^{\dagger})\Pi_{mn}, \\
& - (C_A^{\ddagger^T} \otimes C_A^{\ddagger}) + ((I - CC^{\dagger})^T \otimes C_A^{\ddagger}C^{\dagger^T})\Pi_{sn} + (C^{\dagger^T} QC_A^{\ddagger})^T \otimes (PQP)^{\dagger})\Pi_{sn}] \begin{bmatrix} \mathrm{vec}(\mathrm{d}A) \\ \mathrm{vec}(\mathrm{d}C) \end{bmatrix}.
\end{aligned}
$$

That is,

$$
\mathrm{d}(\mathrm{vec}(C_A^{\ddagger})) = [W(A), W(C)]\mathrm{d}v.
$$

Thus, we have obtained the required result by using the definition of Fréchet derivative. □

Remark 1. *Setting $C = L$, $K = A$, $q = 0$ and C as full row rank, we have $C_A^{\ddagger} = L_K^{\dagger}$ and*

$$
\begin{aligned}
\widetilde{W}(A) &= -[(C_A^{\dagger^T} \otimes (AP)^{\dagger}) + ((AC_A^{\dagger})^T \otimes (AP)^{\dagger}(AP)^{\dagger^T})\Pi_{mn}], \\
\widetilde{W}(C) &= -[(C_A^{\dagger^T} \otimes C_A^{\dagger}) - ((AC^{\dagger})^T AC_A^{\dagger})^T \otimes (AP)^{\dagger}(AP)^{\dagger^T})\Pi_{sn}],
\end{aligned}
$$

where the latter is just the result of ([23], Lemma 3.1), with which we can recover the condition numbers for K-weighted pseudoinverse $L_K^{\dagger}$ [23].

Using the straightforward results of Lemma 1 and Lemma 4, we derive the following condition numbers for $C_A^{\ddagger}$.

Theorem 1. *The normwise, mixed and componentwise condition numbers for $C_A^{\ddagger}$ defined in (11)–(13) are*

$$
n^{\ddagger}(A, C) = \frac{\|[W(A), W(C)]\|_2 \left\| \begin{bmatrix} \mathrm{vec}(A) \\ \mathrm{vec}(C) \end{bmatrix} \right\|_2}{\|\mathrm{vec}(C_A^{\ddagger})\|_2}, \tag{26}
$$

$$
m^{\ddagger}(A, C) = \frac{\| |W(A)|\mathrm{vec}(|A|) + |W(C)|\mathrm{vec}(|C|) \|_\infty}{\|\mathrm{vec}(C_A^{\ddagger})\|_\infty}, \tag{27}
$$

$$
c^{\ddagger}(A, C) = \left\| \frac{|W(A)|\mathrm{vec}(|A|) + |W(C)|\mathrm{vec}(|C|)}{\mathrm{vec}(C_A^{\ddagger})} \right\|_\infty. \tag{28}
$$

Next, we provide easier computable upper bounds by minimizing the cost of computing the above condition numbers. The estimation of the upper bounds will be demonstrated by numerical experiments in Section 6.

Corollary 1. *The upper bounds of normwise, mixed and componentwise condition numbers for $C_A^{\ddagger}$ are*

$$
\begin{aligned}
n^{\ddagger}(A,C) &\le n^{upper}(A,C) \\
&= [\|C_A^{\ddagger}\|_2\|(PQP)^{\dagger}A^TJ\|_2 + \|JC_A^{\ddagger}\|_2\|(PQP)^{\dagger}\|_2 + \|C_A^{\ddagger}\|_2\|C_A^{\ddagger}\|_2 + \|(I-CC^{\dagger})\|_2\|C_A^{\ddagger}C^{\dagger^T}\|_2 \\
&\quad + \|C^{\dagger^T}QC_A^{\ddagger}\|_2\|(PQP)^{\dagger}\|_2]\frac{\|[A,\ C]\|_F}{\|C_A^{\ddagger}\|_F}, \\
m^{\ddagger}(A,C) &\le m^{upper}(A,C) \\
&= \|[|(PQP)^{\dagger}A^TJ||A||C_A^{\ddagger^T}| + |(PQP)^{\dagger}||A^T||JC_A^{\ddagger}| + |C_A^{\ddagger}||C||C_A^{\ddagger^T}| + |C_A^{\ddagger}C^{\dagger^T}||C^T||(I-CC^{\dagger})| \\
&\quad + |(PQP)^{\dagger}||C^T||C^{\dagger^T}QC_A^{\ddagger}|]\|_{\max}/\|C_A^{\ddagger}\|_{\max}, \\
c^{\ddagger}(A,C) &\le c^{upper}(A,C) \\
&= \|[|(PQP)^{\dagger}A^TJ||A||C_A^{\ddagger^T}| + |(PQP)^{\dagger}||A^T||JC_A^{\ddagger}| + |C_A^{\ddagger}||C||C_A^{\ddagger^T}| + |C_A^{\ddagger}C^{\dagger^T}||C^T||(I-CC^{\dagger})| \\
&\quad + |(PQP)^{\dagger}||C^T||C^{\dagger^T}QC_A^{\ddagger}|]/C_A^{\ddagger}\|_{\max}.
\end{aligned}
$$

Proof. For any two matrices X and Y, it is well-known that $\|[X,\ Y]\|_2 \le \|X\|_2 + \|Y\|_2$. With the help of Theorem 1, and (8), we obtain

$$
\begin{aligned}
n^{\ddagger}(A,C) &\le [\| -(C_A^{\ddagger^T} \otimes (PQP)^{\dagger}A^TJ) - ((JC_A^{\ddagger})^T \otimes (PQP)^{\dagger})\Pi_{mn}\|_2 \\
&\quad + \| -(C_A^{\ddagger^T} \otimes C_A^{\ddagger}) + ((I-CC^{\dagger})^T \otimes C_A^{\ddagger}C^{\dagger^T})\Pi_{sn} + (C^{\dagger^T}QC_A^{\ddagger})^T \otimes (PQP)^{\dagger})\Pi_{sn}\|_2] \\
&\quad \times \frac{\|[A,\ C]\|_F}{\|C_A^{\ddagger}\|_F} \\
&\le [\|C_A^{\ddagger^T} \otimes (PQP)^{\dagger}A^TJ\|_2 + \|(JC_A^{\ddagger})^T \otimes (PQP)^{\dagger}\|_2 + \|C_A^{\ddagger^T} \otimes C_A^{\ddagger}\|_2 \\
&\quad + \|(I-CC^{\dagger})^T \otimes C_A^{\ddagger}C^{\dagger^T}\|_2 + \|(C^{\dagger^T}QC_A^{\ddagger})^T \otimes (PQP)^{\dagger}\|_2]\frac{\|[A,\ C]\|_F}{\|C_A^{\ddagger}\|_F} \\
&= [\|C_A^{\ddagger}\|_2\|(PQP)^{\dagger}A^TJ\|_2 + \|JC_A^{\ddagger}\|_2\|(PQP)^{\dagger}\|_2 + \|C_A^{\ddagger}\|_2\|C_A^{\ddagger}\|_2 + \|(I-CC^{\dagger})\|_2\|C_A^{\ddagger}C^{\dagger^T}\|_2 \\
&\quad + \|C^{\dagger^T}QC_A^{\ddagger}\|_2\|(PQP)^{\dagger}\|_2]\frac{\|[A,\ C]\|_F}{\|C_A^{\ddagger}\|_F}.
\end{aligned}
$$

Secondly, by using Lemma 3 and Theorem 1, we obtain

$$
\begin{aligned}
m^{\ddagger}(A,C) &= \||-(C_A^{\ddagger^T} \otimes (PQP)^{\dagger}A^TJ) - ((JC_A^{\ddagger})^T \otimes (PQP)^{\dagger})\Pi_{mn}|\mathrm{vec}(|A|) + |-(C_A^{\ddagger^T} \otimes C_A^{\ddagger}) \\
&\quad + ((I-CC^{\dagger})^T \otimes C_A^{\ddagger}C^{\dagger^T})\Pi_{sn} + (C^{\dagger^T}QC_A^{\ddagger})^T \otimes (PQP)^{\dagger})\Pi_{sn}|\mathrm{vec}(|C|)\|_\infty/\|\mathrm{vec}(C_A^{\ddagger})\|_\infty \\
&\le \|[|(PQP)^{\dagger}A^TJ||A||C_A^{\ddagger^T}| + |(PQP)^{\dagger}||A^T||JC_A^{\ddagger}| + |C_A^{\ddagger}||C||C_A^{\ddagger^T}| + |C_A^{\ddagger}C^{\dagger^T}||C^T||(I-CC^{\dagger})| \\
&\quad + |(PQP)^{\dagger}||C^T||C^{\dagger^T}QC_A^{\ddagger}|]\|_{\max}/\|C_A^{\ddagger}\|_{\max},
\end{aligned}
$$

and finally, we have

$$
\begin{aligned}
c^{\ddagger}(A,C) &= \||-(C_A^{\ddagger^T} \otimes (PQP)^{\dagger}A^TJ) - ((JC_A^{\ddagger})^T \otimes (PQP)^{\dagger})\Pi_{mn}|\mathrm{vec}(|A|) + |-(C_A^{\ddagger^T} \otimes C_A^{\ddagger}) \\
&\quad + ((I-CC^{\dagger})^T \otimes C_A^{\ddagger}C^{\dagger^T})\Pi_{sn} + (C^{\dagger^T}QC_A^{\ddagger})^T \otimes (PQP)^{\dagger})\Pi_{sn}|\mathrm{vec}(|C|)/|\mathrm{vec}(C_A^{\ddagger})|\|_\infty \\
&\le \|[|(PQP)^{\dagger}A^TJ||A||C_A^{\ddagger^T}| + |(PQP)^{\dagger}||A^T||JC_A^{\ddagger}| + |C_A^{\ddagger}||C||C_A^{\ddagger^T}| + |C_A^{\ddagger}C^{\dagger^T}||C^T||(I-CC^{\dagger})| \\
&\quad + |(PQP)^{\dagger}||C^T||C^{\dagger^T}QC_A^{\ddagger}|]/C_A^{\ddagger}\|_{\max}.
\end{aligned}
$$

□

Remark 2. *Using the GHQR factorization [3] on A and C in (2) and (5):*

$$H^T AQ = \begin{pmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{pmatrix}, \quad U^T CQ = \begin{pmatrix} K_{11} & 0 \\ 0 & 0 \end{pmatrix}, \tag{29}$$

where $U \in \mathbb{R}^{s\times s}$ and $Q \in \mathbb{R}^{n\times n}$ and a $J-$orthogonal matrix, $H \in \mathbb{R}^{(p+q)\times(p+q)}$ (i.e., $HJH^T = J$), L_{22} and K_{11} are lower triangular and non-singular. We have

$$C_A^{\ddagger} = Q\begin{pmatrix} I \\ -L_{22}^{-1}L_{21} \end{pmatrix} K_{11}^{-1}U_1^T, \quad (PQP)^{\dagger}A^TJ = Q\begin{pmatrix} 0 \\ L_{22}^{-1} \end{pmatrix} H_2^T, \quad (PQP)^{\dagger} = Q\begin{pmatrix} 0 \\ -(L_{22}^TL_{22})^{-1} \end{pmatrix} Q^T,$$

$$C_A^{\ddagger}C^{\dagger^T} = \begin{pmatrix} 0 \\ -L_{22}^{-1}L_{22} \end{pmatrix}(K_{11}^{-1})^TQ^T, \quad C^{\dagger^T}QC_A^{\ddagger} = U_1^{-T}K_{11}^{-T}L_{11}^{-1}JL_{11}K_{11}^{-1}U_1^T,$$

$$JAC_A^{\ddagger} = JL_{11}K_{11}^{-T}U_1^T, \quad C_A^{\ddagger}C_A^{\ddagger^T} = Q\begin{pmatrix} I \\ -L_{22}^{-1}L_{21} \end{pmatrix}K_{11}^{-1}K_{11}^{-T}\begin{pmatrix} I & -L_{22}^{-1}L_{21} \end{pmatrix}Q^T, \quad CC^{\dagger} = U_1\begin{pmatrix} K_{11}K_{11}^{-1} & 0 \end{pmatrix}U_1^T,$$

where $U = (U_1, U_2)$, $H = [H_1, H_2]$; U_1 and H_1 are, respectively, the submatrices of U and H obtained by taking the first r columns. Putting all the above terms into (18) leads to

$$\begin{aligned} W_1(A) = -\Bigg[&\left(U_1K_{11}^{-T}\begin{pmatrix} I & -L_{22}^{-1}L_{21} \end{pmatrix}Q^T \otimes Q\begin{pmatrix} 0 \\ L_{22}^{-1} \end{pmatrix}H_2^T\right) \\ &+\left(K_{11}^{-T}U_1K_{11}L_{11}^TJ \otimes Q\begin{pmatrix} 0 \\ -(L_{22}^TL_{22})^{-1} \end{pmatrix}Q^T\right)\Pi_{mn}\Bigg], \\ W_1(C) = -\Bigg[&\left(U_1K_{11}^{-T}\begin{pmatrix} I & -L_{22}^{-1}L_{21} \end{pmatrix}Q^T \otimes Q\begin{pmatrix} I \\ -L_{22}^{-1}L_{21} \end{pmatrix}K_{11}^{-1}U_1^T\right) \\ &-\left(\left(I - U_1\begin{pmatrix} K_{11}K_{11}^{-1} & 0 \end{pmatrix}U_1^T\right)^T \otimes \begin{pmatrix} 0 \\ -L_{22}^{-1}L_{22} \end{pmatrix}(K_{11}^{-1})^TQ^T\right)\Pi_{sn} \\ &-(U_1K_{11}^{-T}L_{11}^TJL_{11}^{-T}K_{11}^{-1}U_1^{-1}) \otimes Q\begin{pmatrix} 0 \\ -(L_{22}^TL_{22})^{-1} \end{pmatrix}Q^T)\Pi_{sn}\Bigg]. \end{aligned}$$

Remark 3. *We can obtain* dx *using the* d$(C_A^{\ddagger})$ *expression, where (4) is the solution of EILS problem (3). By differentiating (4), we obtain*

$$\mathrm{d}x = \mathrm{d}(C_A^{\ddagger}h + (PQP)^{\dagger}A^TJg).$$

Thus, using (20), we obtain

$$\begin{aligned} \mathrm{d}x \quad = \quad & \mathrm{d}(C_A^{\ddagger}h + P(PQP)^{\dagger}A^TJg) = \mathrm{d}(C_A^{\ddagger})h + C_A^{\ddagger}\mathrm{d}h + \mathrm{d}P(PQP)^{\dagger}A^TJg + P\mathrm{d}(PQP)^{\dagger}A^TJg \\ & +P(PQP)^{\dagger}\mathrm{d}A^TJg + P(PQP)^{\dagger}A^TJ\mathrm{d}g. \end{aligned}$$

Substituting (25) into above equation and using (9), we have

$$\begin{aligned} \mathrm{d}x \quad = \quad & [-C_A^{\ddagger}\mathrm{d}CC_A^{\ddagger} + C_A^{\ddagger}C^{\dagger^T}\mathrm{d}C^T(I - CC^{\dagger}) + (PQP)^{\dagger}\mathrm{d}C^TC^{\dagger^T}QC_A^{\ddagger} - (PQP)^{\dagger}A^TJ\mathrm{d}AC_A^{\ddagger} \\ & -(PQP)^{\dagger}\mathrm{d}A^TJAC_A^{\ddagger}]h + \mathrm{d}(I - C^{\dagger}C)(PQP)^{\dagger}A^TJg + P[-(PQP)^{\dagger}\mathrm{d}(PQP)(PQP)^{\dagger} \\ & +(PQP)^{\dagger}(PQP)^{\dagger^T}\mathrm{d}(PQP)^T(I - (PQP)(PQP)^{\dagger}) \\ & +(I - (PQP)^{\dagger}(PQP))\mathrm{d}(PQP)^T(PQP)^{\dagger^T}(PQP)^{\dagger}]A^TJg \\ & +P(PQP)^{\dagger}\mathrm{d}A^TJg + P(PQP)^{\dagger}A^TJ\mathrm{d}g + C_A^{\ddagger}\mathrm{d}h, \end{aligned}$$

which together with (20)–(22) give

$$\begin{aligned} dx = & -C_A^{\ddagger} dC C_A^{\ddagger} h + C_A^{\ddagger} C^{\dagger^T} dC^T (I - CC^{\dagger}) h + (PQP)^{\dagger} dC^T C^{\dagger^T} Q C_A^{\ddagger} h - (PQP)^{\dagger} A^T J dA C_A^{\ddagger} h \\ & -(PQP)^{\dagger} dA^T J A C_A^{\ddagger} h - C^{\dagger} dC (PQP)^{\dagger} A^T J g - (PQP)^{\dagger} dQ P (PQP)^{\dagger} A^T J g \\ & -(PQP)^{\dagger} Q dP (PQP)^{\dagger} A^T J g + (PQP)^{\dagger} PQP (PQP)^{\dagger} dP^T (I - (QP)(PQP)^{\dagger}) A^T J g \\ & +(PQP)^{\dagger} (PQP)^{\dagger^T} dQ^T P (I - (QP)(PQP)^{\dagger}) A^T J g + P (PQP)^{\dagger} dA^T J g + P (PQP)^{\dagger} A^T J dg + C_A^{\ddagger} dh. \end{aligned}$$

Noting (24), the above equation can be rewritten as

$$\begin{aligned} dx = & -C_A^{\ddagger} dC C_A^{\ddagger} h + C_A^{\ddagger} C^{\dagger^T} dC^T (I - CC^{\dagger}) h + (PQP)^{\dagger} dC^T C^{\dagger^T} A^T J A C_A^{\ddagger} h - (PQP)^{\dagger} A^T J dA C_A^{\ddagger} h \\ & -(PQP)^{\dagger} dA^T J A C_A^{\ddagger} h - C^{\dagger} dC (PQP)^{\dagger} A^T J g + (PQP)^{\dagger} dA^T J (g - A (PQP)^{\dagger} A^T J g) \\ & -(PQP)^{\dagger} A^T J dA (PQP)^{\dagger} A^T J g + (PQP)^{\dagger} Q C^{\dagger} dC (PQP)^{\dagger} A^T J g \\ & -(PQP)^{\dagger} dC^T C^{\dagger^T} A^T J (g - A (PQP)^{\dagger} A^T J g) + (PQP)^{\dagger} A^T J dg + C_A^{\ddagger} dh. \end{aligned}$$

Further, by (20) and (4), we have

$$\begin{aligned} dx = & -C_A^{\ddagger} dC (C_A^{\ddagger} h + (PQP)^{\dagger} A^T J g) + C_A^{\ddagger} C^{\dagger^T} dC^T (I - CC^{\dagger}) h - (PQP)^{\dagger} A^T J dA (C_A^{\ddagger} h + (PQP)^{\dagger} A^T J g) \\ & -(PQP)^{\dagger} dC^T C^{\dagger^T} A^T J (g - A (C_A^{\ddagger} h + (PQP)^{\dagger} A^T J g)) \\ & +(PQP)^{\dagger} dA^T J (g - A (C_A^{\ddagger} h + (PQP)^{\dagger} A^T J g)) + (PQP)^{\dagger} A^T J dg + C_A^{\ddagger} dh \quad \text{by (20)} \\ = & -C_A^{\ddagger} dC x + C_A^{\ddagger} C^{\dagger^T} dC^T \rho - (PQP)^{\dagger} A^T J dA x - (PQP)^{\dagger} dC^T C^{\dagger^T} A^T J r \\ & +(PQP)^{\dagger} dA^T J r + (PQP)^{\dagger} A^T J dg + C_A^{\ddagger} dh, \quad \text{by (4)} \end{aligned} \tag{30}$$

where $s = Jr = J(g - Ax)$, $\beta = (I - CC^{\dagger})h$. By utilizing operator "vec" on (30), and using (6) and (7), we obtain

$$\begin{aligned} dx = & -(x^T \otimes (PQP)^{\dagger} A^T J) \mathrm{vec}(dA) + (s^T \otimes (PQP)^{\dagger}) \mathrm{vec}(dA^T) - (x^T \otimes C_A^{\ddagger}) \mathrm{vec}(dC) \\ & +(\beta^T \otimes C_A^{\ddagger} C^{\dagger^T}) \mathrm{vec}(dC^T) + (C^{\dagger^T} A^T s)^T \otimes (PQP)^{\dagger}) \mathrm{vec}(dC^T) + (PQP)^{\dagger} dg + C_A^{\ddagger} dh \quad \text{by (6)} \\ = & [-(x^T \otimes (PQP)^{\dagger} A^T J) + (s^T \otimes (PQP)^{\dagger}) \Pi_{mn}] \mathrm{vec}(dA) - [(x^T \otimes C_A^{\ddagger}) - (\beta^T \otimes C_A^{\ddagger} C^{\dagger^T}) \Pi_{sn} \\ & -((C^{\dagger^T} A^T s)^T \otimes (PQP)^{\dagger}) \Pi_{sn}] \mathrm{vec}(dC) + (PQP)^{\dagger} dg + C_A^{\ddagger} dh \quad \text{by (7)} \\ = & \big[-(x^T \otimes (PQP)^{\dagger} A^T J) + (s^T \otimes (PQP)^{\dagger}) \Pi_{mn}, \ -(x^T \otimes C_A^{\ddagger}) + (\beta^T \otimes C_A^{\ddagger} C^{\dagger^T}) \Pi_{sn} \\ & +((C^{\dagger^T} A^T s)^T \otimes (PQP)^{\dagger}) \Pi_{sn}, \ (PQP)^{\dagger}, \ C_A^{\ddagger} \big] \begin{bmatrix} \mathrm{vec}(dA) \\ \mathrm{vec}(dC) \\ dg \\ dh \end{bmatrix}. \end{aligned}$$

From the above result, we can recover the condition numbers of the EILS problem provided in [3,13,14]. Further, we observe that $r = (g - A(C_A^{\ddagger} h + (PQP)^{\dagger} A^T J g))$. Applying the same procedure, we can determine dr *and condition numbers for residuals of EILS.*

4. Componentwise Perturbation Analysis

In the following section, we derive a componentwise perturbation analysis of the augmented system for the EILS problem.

Let the perturbations $\mathrm{d}A \in \mathbb{R}^{(p+q)\times n}$, $\mathrm{d}C \in \mathbb{R}^{s\times n}$, $\mathrm{d}g \in \mathbb{R}^{m}$ and $\mathrm{d}h \in \mathbb{R}^{s}$ satisfy $|\mathrm{d}A| \le \epsilon|A|$, $|\mathrm{d}C| \le \epsilon|C|$ $|\mathrm{d}g| \le \epsilon|g|$ and $|\mathrm{d}h| \le \epsilon|h|$ for a small ϵ and $s = Jr$. Suppose that the perturbed augmented system is

$$\begin{bmatrix} 0 & 0 & C+\mathrm{d}C \\ 0 & J+\mathrm{d}J & A+\mathrm{d}A \\ (C+\mathrm{d}C)^{\mathrm{T}} & (A+\mathrm{d}A)^{\mathrm{T}} & 0 \end{bmatrix} \begin{bmatrix} \lambda+\mathrm{d}\lambda \\ s+\mathrm{d}s \\ x+\mathrm{d}x \end{bmatrix} = \begin{bmatrix} h+\mathrm{d}h \\ g+\mathrm{d}g \\ 0 \end{bmatrix}$$

Denoting

$$S = \begin{bmatrix} 0 & 0 & C \\ 0 & J & A \\ C^{\mathrm{T}} & A^{\mathrm{T}} & 0 \end{bmatrix}, \quad u = \begin{bmatrix} g \\ h \\ 0 \end{bmatrix}, \quad v = \begin{bmatrix} \lambda \\ s \\ x \end{bmatrix}$$

and the perturbations

$$\mathrm{d}S = \begin{bmatrix} 0 & 0 & \mathrm{d}C \\ 0 & \mathrm{d}J & \mathrm{d}A \\ (\mathrm{d}C)^{\mathrm{T}} & (\mathrm{d}A)^{\mathrm{T}} & 0 \end{bmatrix}, \quad \mathrm{df} = \begin{bmatrix} \mathrm{d}g \\ \mathrm{d}h \\ 0 \end{bmatrix}, \quad \mathrm{dz} = \begin{bmatrix} \mathrm{d}\lambda \\ \mathrm{d}s \\ \mathrm{d}x \end{bmatrix}.$$

When A is of full column rank and C has full row rank, S is invertible. It can be verified that

$$S^{-1} = \begin{bmatrix} {C_A^{\ddagger}}^{T} Q C_A^{\ddagger} & -\left(JAC_A^{\ddagger}\right)^{T} & {C_A^{\ddagger}}^{T} \\ -JAC_A^{\ddagger} & J - JA(PQP)^{\dagger}A^{T}J & JA(PQP)^{\dagger} \\ C_A^{\ddagger} & (PQP)^{\dagger}A^{T}J & -(PQP)^{\dagger} \end{bmatrix}.$$

If the spectral radius

$$\rho\left(\left|S^{-1}\right||\mathrm{d}S|\right) < 1 \tag{31}$$

then $I_{m+n} + S^{-1}\mathrm{d}S$ is invertible. Clearly, the condition

$$\epsilon < \rho^{-1}\left(\begin{bmatrix} |{C_A^{\ddagger}}^{T}||C|^{\mathrm{T}} & |{C_A^{\ddagger}}^{T}||A|^{\mathrm{T}} & |{C_A^{\ddagger}}^{T} Q C_A^{\ddagger}||C| + |(JAC_A^{\ddagger})^{T}||A| \\ |JA(PQP)^{\dagger}||C|^{\mathrm{T}} & |JA(PQP)^{\dagger}||A|^{\mathrm{T}} & \left|JAC_A^{\ddagger}\right||A| + |J - JA(PQP)^{\dagger}A^{T}J||C| \\ |(PQP)^{\dagger}||C|^{\mathrm{T}} & |(PQP)^{\dagger}||A|^{\mathrm{T}} & |C_A^{\ddagger}||C| + |(PQP)^{\dagger}A^{T}J||A| \end{bmatrix}\right), \tag{32}$$

implies (31). The following results [33] are important for Theorem 2.

Lemma 5. *The perturbed system of a linear system $Sv = u$ is defined as follows:*

$$(S + \mathrm{d}S)(v + \mathrm{d}v) = u + \mathrm{d}u,$$

where $v + \mathrm{d}v$ is the solution to the perturbed system, when the perturbations $\mathrm{d}S$ and $\mathrm{d}u$ are sufficiently small such that $S + \mathrm{d}S$ is invertible, the perturbation $\mathrm{d}v$ in the solution v satisfies

$$\mathrm{d}v = \left(I + S^{-1}\mathrm{d}S\right)^{-1} S^{-1}(\mathrm{d}u - \mathrm{d}Sv),$$

which implies

$$|\mathrm{d}v| \le \left|\left(I + S^{-1}\mathrm{d}S\right)^{-1}\right|\left|S^{-1}\right|(|\mathrm{d}u| + |\mathrm{d}S||v|).$$

Furthermore, when the spectral radius $\rho\left(|S^{-1}||\mathrm{d}S|\right) < 1$*, we have*

$$\begin{aligned} |\mathrm{d}v| &\leq \left(I - \left|S^{-1}\right||\mathrm{d}S|\right)^{-1}\left|S^{-1}\right|(|\mathrm{d}u| + |\mathrm{d}S||v|) \\ &= \left(I + O\left(\left|S^{-1}\right||\mathrm{d}S|\right)\right)\left|S^{-1}\right|(|\mathrm{d}u| + |\mathrm{d}S||v|). \end{aligned} \tag{33}$$

Now, we have the following bounds for the perturbations in the equality constrained indefinite least squares solution and residual.

Theorem 2. *Under the above assumption, for any* $\epsilon > 0$ *satisfying the condition (32), when the componentwise perturbations* $|\mathrm{d}A| \leq \epsilon|A|$, $|\mathrm{d}C| \leq \epsilon|C|$ $|\mathrm{d}g| \leq \epsilon|g|$ *and* $|\mathrm{d}h| \leq \epsilon|h|$*, the error in the solution is bounded by*

$$\begin{aligned} \|\mathrm{d}x\|_\infty \leq \epsilon\Big(&\|C_A^{\ddagger}(|h| + |C||x|)\|_\infty + \left\|(PQP)^{\dagger}A^T J(|g| + |A||x|)\right\|_\infty + \left\|(PQP)^{\dagger}(|C|^T|\lambda| + |A|^T|r|)\right\|_\infty\Big) \\ &+ O\left(\epsilon^2\right) \end{aligned} \tag{34}$$

and error in the residual is bounded by

$$\begin{aligned} \|\mathrm{d}r\|_\infty \leq \epsilon\Big(&\|JAC_A^{\ddagger}(|h| + |C||x|)\|_\infty + \|J - JA(PQP)^{\dagger}A^T J(|g| + |A||x|)\|_\infty \\ &+ \|JA(PQP)^{\dagger}(|C|^T|\lambda| + |A|^T|r|)\|_\infty\Big) + O\left(\epsilon^2\right). \end{aligned} \tag{35}$$

Proof. Since the condition (32) implies (31), applying (33) in Lemma 5, we obtain

$$\begin{bmatrix} \mathrm{d}\lambda \\ \mathrm{d}s \\ \mathrm{d}x \end{bmatrix} \leq \left(I + O\left(\left|S^{-1}\right||\mathrm{d}S|\right)\right)\left|S^{-1}\right|\begin{bmatrix} |\mathrm{d}h| + |\mathrm{d}C||x| \\ |\mathrm{d}g| + |\mathrm{d}A||x| \\ |\mathrm{d}C|^{\mathrm{T}}|\lambda| + |\mathrm{d}A|^{\mathrm{T}}|r| \end{bmatrix}.$$

Finally, using the conditions $|\mathrm{d}A| \leq \epsilon|A|$, $|\mathrm{d}C| \leq \epsilon|C|$ $|\mathrm{d}g| \leq \epsilon|g|$ and $|\mathrm{d}h| \leq \epsilon|h|$, and the explicit form of S^{-1}, the upper bounds (34) and (35) can be obtained. □

Furthermore, we can obtain the componentwise perturbation bounds of the indefinite least squares solution and its residual.

Remark 4. *Assume that* C *is a zero matrix,* $\lambda = 0$*, and* $h = 0$*. Using the above notations, for any* $\epsilon > 0$*, if the componentwise perturbations satisfy* $|\mathrm{d}A| \leq \epsilon|A|$ *and* $|\mathrm{d}g| \leq \epsilon|g|$*, then the error in the solution is bounded by*

$$\|\mathbf{dx}\|_\infty \leq \epsilon\left(\||\left(\left(A^{\mathrm{T}}JA\right)^{-1}\right)A^{\mathrm{T}}J|(|g| + |A||\mathbf{x}|)\|_\infty + \|\left(A^{\mathrm{T}}JA\right)^{-1}||A|^{\mathrm{T}}|r|\|_\infty\right) + O\left(\epsilon^2\right)$$

and the error in the residual is bounded by

$$\|\mathrm{d}r\|_\infty \leq \epsilon\left(\||J - JA\left(A^{\mathrm{T}}JA\right)^{-1}A^{\mathrm{T}}J|(|g| + |A||\mathbf{x}|)\|_\infty + \||JA\left(A^{\mathrm{T}}JA\right)^{-1}||A|^{\mathrm{T}}|r|\|_\infty\right) + O\left(\epsilon^2\right).$$

5. Statistical Condition Estimates

This section proposes three algorithms for estimating the normwise, mixed and componentwise condition numbers for the generalized inverse $C_A^{\ddagger}$. Algorithm 1 is based on a probabilistic condition estimator method [27] and utilized to examine the normwise condition number for K-weighted pseudoinverse $L_K^{\dagger}$ [23], ILS problem [34], constrained and weighted least squares problem [35] and Tikhonov regularization of total least squares problem [36]. Based on the SSCE method [28], we develop Algorithm 2 to estimate the normwise condition number; for details, see [23,34,37–39].

Algorithm 1 (Probabilistic condition estimator for the normwise condition number)

1. Compute the derivative $\mathrm{d}\phi(u) = [W(A), W(C)]$, and choose a starting vector u_0 uniformly and randomly from the unit t-sphere S_{t-1} with $t = n^2$.
2. Using the probabilistic spectral norm estimator [27], compute the certain lower bound α_1 and the probabilistic upper bound α_2 of $\mathrm{d}\phi(u)$.
3. Compute the normwise condition number by using (26)

$$n_p^{\ddagger}(A,C) = \frac{n_p(A,C)\|[A,\ C]\|_F}{\|C_A^{\ddagger}\|_F} \text{ with } n_p(A,C) \quad = \quad \sqrt{\frac{\alpha_1+\alpha_2}{2}}.$$

Algorithm 2 (Small-sample statistical condition estimation method for the normwise condition number)

1. Generate matrices $[\mathrm{d}A_1, \mathrm{d}C_1], [\mathrm{d}A_2, \mathrm{d}C_2], \ldots, [\mathrm{d}A_q, \mathrm{d}C_q]$ with each entry in $\mathcal{N}(0,1)$ and Orthonormalize the following matrix

$$\begin{bmatrix} \mathrm{vec}(\mathrm{d}A_1) & \mathrm{vec}(\mathrm{d}A_2) & \cdots & \mathrm{vec}(\mathrm{d}A_q) \\ \mathrm{vec}(\mathrm{d}C_1) & \mathrm{vec}(\mathrm{d}C_2) & \cdots & \mathrm{vec}(\mathrm{d}C_q) \end{bmatrix}$$

   to obtain $[\tau_1, \tau_2, \ldots, \tau_q]$ by modified Gram-Schmidt orthogonalization process. Each τ_i can be converted into the corresponding matrices $[\mathrm{d}A_i, \mathrm{d}C_i]$ by applying the unvec operation.
2. Let $p = m + mn$. Approximate ω_p and ω_q by

$$\omega_k \approx \sqrt{\frac{2}{\pi(k-\frac{1}{2})}} \tag{36}$$

3. For $i = 1, 2, \ldots, q$, compute

$$\theta_i = -C_A^{\ddagger}\mathrm{d}C_iC_A^{\ddagger} + C_A^{\ddagger}C^{\dagger^T}\mathrm{d}C_i^T(I - CC^{\dagger}) + (PQP)^{\dagger}\mathrm{d}C_i^TC^{\dagger^T}QC_A^{\ddagger} - (PQP)^{\dagger}A^TJ\mathrm{d}A_iC_A^{\ddagger} - (PQP)^{\dagger}\mathrm{d}A_i^TJAC_A^{\ddagger}.$$

4. Compute the absolute condition vector by

$$\kappa_{\text{abs}}^{\ddagger} := \frac{\omega_q}{\omega_p}\sqrt{|\theta_1|^2 + |\theta_2|^2 + \cdots + |\theta_q|^2}, \tag{37}$$

   where the square operation is applied to each entry of $\theta_i, i = 1, 2, \ldots, q$ and the square root is also applied componentwise.
5. Estimate the normwise condition number (26) by

$$n^{\ddagger}(A,C) = \frac{N_{\text{SCE}}^{\ddagger}\|[A,C]\|_F}{\left\|C_A^{\ddagger}\right\|_F}, \tag{38}$$

   where $N_{\text{SCE}}^{\ddagger} := \frac{\omega_q}{\omega_p}\sqrt{\|\sigma_1\|_2^2 + \|\sigma_2\|_2^2 + \cdots + \|\sigma_q\|_2^2} = \left\|\kappa_{\text{abs}}^{\ddagger}\right\|_F$.

To estimate the mixed and componentwise condition numbers, we need the following SSCE method, which is from [28] and has been applied to many problems (see, e.g., [23,32,34–36]).

6. Numerical Experiments

In the following section, we illustrate two specific examples. The first compares the normwise, mixed and componentwise condition numbers and their upper bounds. The second is used to present the efficiency of statistical condition estimators.

Example 1. *In this example, we first compute the condition numbers and their upper bounds by using the below matrix pair, then we demonstrate the reliability of Algorithms 1–3. Matlab2018a has been used to perform all the numerical experiments. We examine 200 matrices that are created by repeatedly applying the matrices* $A \in \mathbb{R}^{m\times n}$ *from [34] and* $C \in \mathbb{R}^{s\times n}$ *below.*

$$A = \begin{bmatrix} U_p & 0 \\ 0 & U_q \end{bmatrix} \begin{bmatrix} D \\ 0 \end{bmatrix} V, U_p = I_p - 2u_p u_p^T, U_q = I_q - 2u_q u_q^T, \text{ and } V = I_n - 2vv^T,$$

where $u_p \in \mathbb{R}^p, u_q \in \mathbb{R}^q$ *and* $v \in \mathbb{R}^n$ *are unit random vectors obtained from Matlab function* randn$(\cdot, 1)$ *and* $D = n^{-l}\,\mathrm{diag}\left(n^l, (n-1)^l, \cdots, 1^l\right)$. *It is simple to determine that the condition number of A, i.e.,* $\kappa(A) = \|A\|_2 \|A^\dagger\|_2$, *is* n^l. $C = [C_1, 0]$, *where* C_1 *is a nonsymmetric Gaussian random Toeplitz matrix generated by the Matlabs function* toeplitz(c, r) *with* $c = \mathrm{randn}(s, 1)$, $r = \mathrm{randn}(s, 1)$. *From Table 1, we can see the numerical outcomes of the ratios given by*

$$\omega_1 = n^{upper}(A, C)/n^{\ddagger}(A, C), \quad \omega_2 = m^{upper}(A, C)/m^{\ddagger}(A, C) \quad \text{and} \quad \omega_3 = c^{upper}(A, C)/c^{\ddagger}(A, C).$$

Table 1. Comparison of condition numbers and their upper bounds by choosing different values of p, q, s and n.

		Mean	Max	Mean	Max	Mean	Max
n^1	p,q,n,s	ω_1		ω_2		ω_3	
	25, 15, 20, 10	1.0763×10^0	4.8422×10^0	1.0647×10^0	2.8373×10^0	1.1538×10^0	3.9657×10^0
	50, 30, 40, 20	1.3146×10^0	6.7089×10^0	1.0861×10^0	4.4630×10^0	1.2845×10^0	5.9123×10^0
	75, 45, 60, 30	1.7422×10^0	1.6402×10^1	1.1965×10^0	1.2847×10^1	1.0784×10^0	1.5766×10^1
	100, 60, 80, 40	2.6043×10^0	1.9461×10^1	1.4574×10^0	1.6783×10^1	1.7540×10^0	1.8452×10^1
n^2	p,q,n,s	ω_1		ω_2		ω_3	
	25, 15, 20, 10	1.4032×10^0	5.7654×10^0	1.2433×10^0	4.6501×10^0	1.3601×10^0	5.3752×10^0
	50, 30, 40, 20	1.7341×10^0	8.2074×10^0	1.5623×10^0	6.4738×10^0	1.7320×10^0	7.2004×10^0
	75, 45, 60, 30	2.5254×10^0	2.8732×10^1	1.8510×10^0	1.6062×10^1	2.0653×10^0	2.2903×10^1
	100, 60, 80, 40	2.7034×10^0	3.9543×10^1	2.0312×10^0	2.0106×10^1	2.3871×10^0	2.4803×10^1
n^3	p,q,n,s	ω_1		ω_2		ω_3	
	25, 15, 20, 10	1.7301×10^0	7.9662×10^0	1.4607×10^0	6.8606×10^0	1.5296×10^0	8.0651×10^0
	50, 30, 40, 20	1.9674×10^0	3.7649×10^1	1.7065×10^0	8.5963×10^0	1.8472×10^0	9.7063×10^0
	75, 45, 60, 30	2.7055×10^0	5.6570×10^1	2.0276×10^0	3.2613×10^1	2.3601×10^0	4.6904×10^1
	100, 60, 80, 40	2.9867×10^0	7.1601×10^1	2.2760×10^0	4.9013×10^1	2.5935×10^0	5.9721×10^1
n^4	p,q,n,s	ω_1		ω_2		ω_3	
	25, 15, 20, 10	1.8271×10^0	2.3021×10^1	1.6354×10^0	1.4032×10^1	1.7925×10^0	1.5102×10^1
	50, 30, 40, 20	2.3064×10^0	3.7632×10^1	1.9642×10^0	1.5210×10^1	1.9862×10^0	1.6082×10^1
	75, 45, 60, 30	2.8063×10^0	7.4310×10^1	2.0513×10^0	5.0471×10^1	2.6743×10^0	6.0437×10^1
	100, 60, 80, 40	2.9887×10^0	8.6501×10^1	2.3810×10^0	7.1089×10^1	2.7011×10^0	7.4810×10^1

To show the efficiency of the three algorithms discussed above, we run some numerical tests and choose parameters $\delta = 0.01$ *and* $\epsilon = 0.001$ *for Algorithm 1 and* $k = 2$ *for Algorithms 2 and 3. The ratios between the exact condition numbers and their estimated values are determined as follows:*

$$\begin{aligned} r_p &= n_p^{\ddagger}(A, C)/n^{\ddagger}(A, C), \ r_s = n_s^{\ddagger}(A, C)/n^{\ddagger}(A, C), \\ r_m &= m_s^{\ddagger}(A, C)/m^{\ddagger}(A, C), \ r_c = c_s^{\ddagger}(A, C)/c^{\ddagger}(A, C), \end{aligned}$$

where r_p *is the ratio between the exact normwise condition number and the estimated value of Algorithm 1,* r_s *is the ratio between the exact normwise condition number and the estimated value of Algorithm 2, and* r_m *and* r_c *are the ratios between the exact mixed and componentwise condition numbers and estimated values of Algorithm 3.*

The results in Table 2 demonstrate that Algorithms 1–3 can reliably estimate the condition numbers in most situations, supporting the statement in ([40], Chapter 15) that an estimate of the condition number that is correct to within a factor 10 is generally appropriate because it is the magnitude of an error bound that is of interest, not its precise value. For the normwise condition number, Algorithm 1 works more effectively and stably.

Table 2. Results by choosing different values of p, q, s and n for Algorithms 1–3.

		Mean	Variance	Mean	Variance	Mean	Variance	Mean	Variance
n^1	p,q,n,s	r_p		r_s		r_m		r_c	
	25, 15, 20, 10	1.0000×10^0	5.3577×10^{-11}	1.0322×10^0	1.2063×10^{-1}	1.0067×10^0	1.3505×10^{-2}	1.2785×10^0	1.0431×10^{-2}
	50, 30, 40, 20	1.0000×10^0	7.0635×10^{-9}	1.1439×10^0	3.5027×10^{-1}	1.0134×10^0	3.9054×10^{-2}	1.3744×10^0	3.6397×10^{-2}
	75, 45, 60, 30	1.0001×10^0	1.5165×10^{-11}	1.2906×10^0	4.6021×10^{-1}	1.1075×10^0	4.1653×10^{-2}	1.5043×10^0	3.9428×10^{-2}
	100, 60, 80, 40	1.0001×10^0	1.7940×10^{-12}	1.3482×10^0	5.7803×10^{-1}	1.2306×10^0	4.9563×10^{-2}	1.8732×10^0	4.6543×10^{-2}
n^2	p,q,n,s	r_p		r_s		r_m		r_c	
	25, 15, 20, 10	1.0000×10^0	6.5102×10^{-9}	1.2654×10^0	2.7360×10^{-1}	1.3405×10^0	3.4605×10^{-2}	1.2765×10^0	2.6123×10^{-2}
	50, 30, 40, 20	1.0000×10^0	7.4738×10^{-11}	1.4783×10^0	4.4925×10^{-1}	1.7169×10^0	4.8543×10^{-2}	1.5063×10^0	4.3326×10^{-2}
	75, 45, 60, 30	1.0001×10^0	1.6062×10^{-9}	1.6295×10^0	6.8732×10^{-1}	1.8206×10^0	6.4890×10^{-2}	1.7422×10^0	5.0542×10^{-2}
	100, 60, 80, 40	1.0001×10^0	2.5106×10^{-13}	1.8693×10^0	7.9543×10^{-1}	2.1456×10^0	7.4293×10^{-2}	2.0361×10^0	6.3702×10^{-2}
n^3	p,q,n,s	r_p		r_s		r_m		r_c	
	25, 15, 20, 10	1.0000×10^0	1.7029×10^{-8}	1.2063×10^0	4.2083×10^{-1}	1.6710×10^0	5.7862×10^{-2}	1.3722×10^0	4.7031×10^{-2}
	50, 30, 40, 20	1.0000×10^0	2.4771×10^{-11}	1.7033×10^0	7.2035×10^{-1}	1.8041×10^0	6.0165×10^{-2}	1.5760×10^0	5.7402×10^{-2}
	75, 45, 60, 30	1.0002×10^0	6.1041×10^{-12}	2.0654×10^0	7.5293×10^{-1}	2.2054×10^0	8.3014×10^{-2}	2.0113×10^0	7.2461×10^{-2}
	100, 60, 80, 40	1.0003×10^0	5.6854×10^{-13}	2.1976×10^0	8.2063×10^{-1}	2.2593×10^0	8.6458×10^{-2}	2.1263×10^0	7.9432×10^{-2}
n^4	p,q,n,s	r_p		r_s		r_m		r_c	
	25, 15, 20, 10	1.0000×10^0	5.6321×10^{-7}	1.6305×10^0	6.2092×10^{-1}	1.9455×10^0	6.7402×10^{-2}	1.8240×10^0	6.0461×10^{-2}
	50, 30, 40, 20	1.0000×10^0	6.0573×10^{-9}	1.7002×10^0	8.0210×10^{-1}	1.9822×10^0	8.0549×10^{-2}	1.9701×10^0	7.4322×10^{-2}
	75, 45, 60, 30	1.0003×10^0	8.6021×10^{-11}	2.1533×10^0	9.0425×10^{-1}	2.4003×10^0	9.3614×10^{-2}	2.2764×10^0	8.4681×10^{-2}
	100, 60, 80, 40	1.0004×10^0	2.8543×10^{-12}	2.4187×10^0	9.2054×10^{-1}	2.6005×10^0	9.5370×10^{-2}	2.5711×10^0	9.4502×10^{-2}

Algorithm 3 (Small-sample statistical condition estimation method for the mixed and componentwise condition numbers)

1. Generate matrices $[\mathrm{d}A_1, \mathrm{d}C_1], [\mathrm{d}A_2, \mathrm{d}C_2], \ldots, [\mathrm{d}A_q, \mathrm{d}C_q]$ with each entry in $\mathcal{N}(0,1)$ and Orthonormalize the following matrix:

$$\begin{bmatrix} \mathrm{vec}(\mathrm{d}A_1) & \mathrm{vec}(\mathrm{d}A_2) & \cdots & \mathrm{vec}(\mathrm{d}A_q) \\ \mathrm{vec}(\mathrm{d}C_1) & \mathrm{vec}(\mathrm{d}C_2) & \cdots & \mathrm{vec}(\mathrm{d}C_q) \end{bmatrix}$$

   to obtain $[\tau_1, \tau_2, \ldots, \tau_q]$ by modified Gram-Schmidt orthogonalization process. Each τ_i can be converted into the corresponding matrices $[\mathrm{d}A_i, \mathrm{d}C_i]$ by applying the unvec operation. Let $[\mathrm{d}A_i, \mathrm{d}C_i]$ be the matrix $\left[\widetilde{\mathrm{d}A_i}, \widetilde{\mathrm{d}C_i}\right]$ multiplied by $[A, C]$ componentwise.
2. Let $p = mn + sn$. Approximate ω_p and ω_q by (36).
3. For $i = 1, 2, \ldots, q$, compute

$$\theta_i = -C_A^{\ddagger}\mathrm{d}C_i C_A^{\ddagger} + C_A^{\ddagger}C^{\dagger^T}\mathrm{d}C_i^T(I - CC^{\dagger}) + (PQP)^{\dagger}\mathrm{d}C_i^T C^{\dagger^T} Q C_A^{\ddagger} - (PQP)^{\dagger}A^T J \mathrm{d}A_i C_A^{\ddagger} - (PQP)^{\dagger}\mathrm{d}A_i^T J A C_A^{\ddagger}.$$

   Using the approximations for ω_p and ω_q, compute the absolute condition vector

$$\kappa_{sce}^{\dagger} = \frac{\omega_q}{\omega_p}\sqrt{|\theta_1|^2 + |\theta_2|^2 + \cdots + |\theta_q|^2}$$

4. Estimate the mixed and componentwise condition estimations $m^{\ddagger}_{sce}(A,C)$ and $c^{\ddagger}_{sce}(A,C)$ as follows:
$$m^{\ddagger}_{s}(A,C) = \frac{\|\kappa^{\dagger}_{sce}\|_\infty}{\|\text{vec}(C^{\ddagger}_{A})\|_\infty}, \quad c^{\ddagger}_{s}(A,C) = \left\| \frac{\kappa^{\dagger}_{sce}}{\text{vec}(C^{\ddagger}_{A})} \right\|_\infty.$$

Example 2. *On similar patrons given in [2,3,5], we generate A and C matrices using the GHQR factorization.*
$$H^T AQ = \begin{bmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{bmatrix}, \; U^T CQ = \begin{bmatrix} K_{11} & 0 \\ 0 & 0 \end{bmatrix},$$
where $H \in \mathbb{R}^{(p+q)\times(p+q)}$ is J-orthogonal, i.e., $HJH^T = J$, Q is orthogonal, and $L_{22} \in \mathbb{R}^{(n-s)\times(n-s)}$ and $K_{11} \in \mathbb{R}^{(s\times s)}$ are lower triangular and non-singular, respectively. In our experiment, we let L_{11} L_{21} be random matrices. H is a random J-orthogonal matrix with a specific condition number generated using the method described in [41]. $Q \in \mathbb{R}^{(n\times n)}$ and $U \in \mathbb{R}^{(s\times s)}$ generated randomly (by Matlabs gallery (‘qmult’,....)*), L_{22}, and K_{11} are generated by QR factorization of random matrices with specified condition numbers and pre-assigned singular value distributions (generated via Matlabs* gallery (‘randsvd’,...)*). To examine the above algorithms’ performance, we use* 500 *matrix pairs, variate the condition numbers of A and C, and set $p = 50$, $q = 30$, $n = 40$, and $s = 20$. The ratios between the exact condition numbers and their estimated values are below.*
$$\begin{aligned} r_p &= n^{\ddagger}_p(A,C)/n^{\ddagger}(A,C), \; r_s = n^{\ddagger}_s(A,C)/n^{\ddagger}(A,C), \\ r_m &= m^{\ddagger}_s(A,C)/m^{\ddagger}(A,C), \; r_c = c^{\ddagger}_s(A,C)/c^{\ddagger}(A,C), \end{aligned}$$
where the parameters δ, ϵ, k, and ratios r_p, r_s, r_m and r_c are the same as given in Example 1. We present these numerical results and CPU time in Figures 1 and 2. The time ratios are defined by
$$t_p := \frac{t_1}{t}, \quad t_s := \frac{t_2}{t}, \quad t_m := \frac{t_3}{t}, \quad t_c := \frac{t_4}{t},$$
where t is the CPU time of computing the generalized inverse $C^{\ddagger}_{A}$ by GHQR decomposition [20]. t_1 is the CPU time of Algorithm 1, t_2 is the CPU time of Algorithm 2, and t_3 and t_4 are the CPU times of Algorithm 3. From Figures 1 and 2, we can see that these three algorithms are highly efficient in estimating condition numbers. However, Table 3 shows that the CPU times of Algorithms 1 and 2 are smaller than Algorithm 3.

Table 3. CPU times for Algorithms A, B, and C by choosing different values of p, q, s and n.

p, q, n, s	t_p	t_s	t_m	t_c
25, 15, 20, 10	0.1065	0.2742	0.7601	0.4643
75, 45, 60, 30	0.3784	0.5204	1.3644	1.1677
100, 60, 80, 40	0.4842	0.6032	1.4569	1.2658
120, 80, 100, 50	0.5643	0.7411	1.6345	1.5403

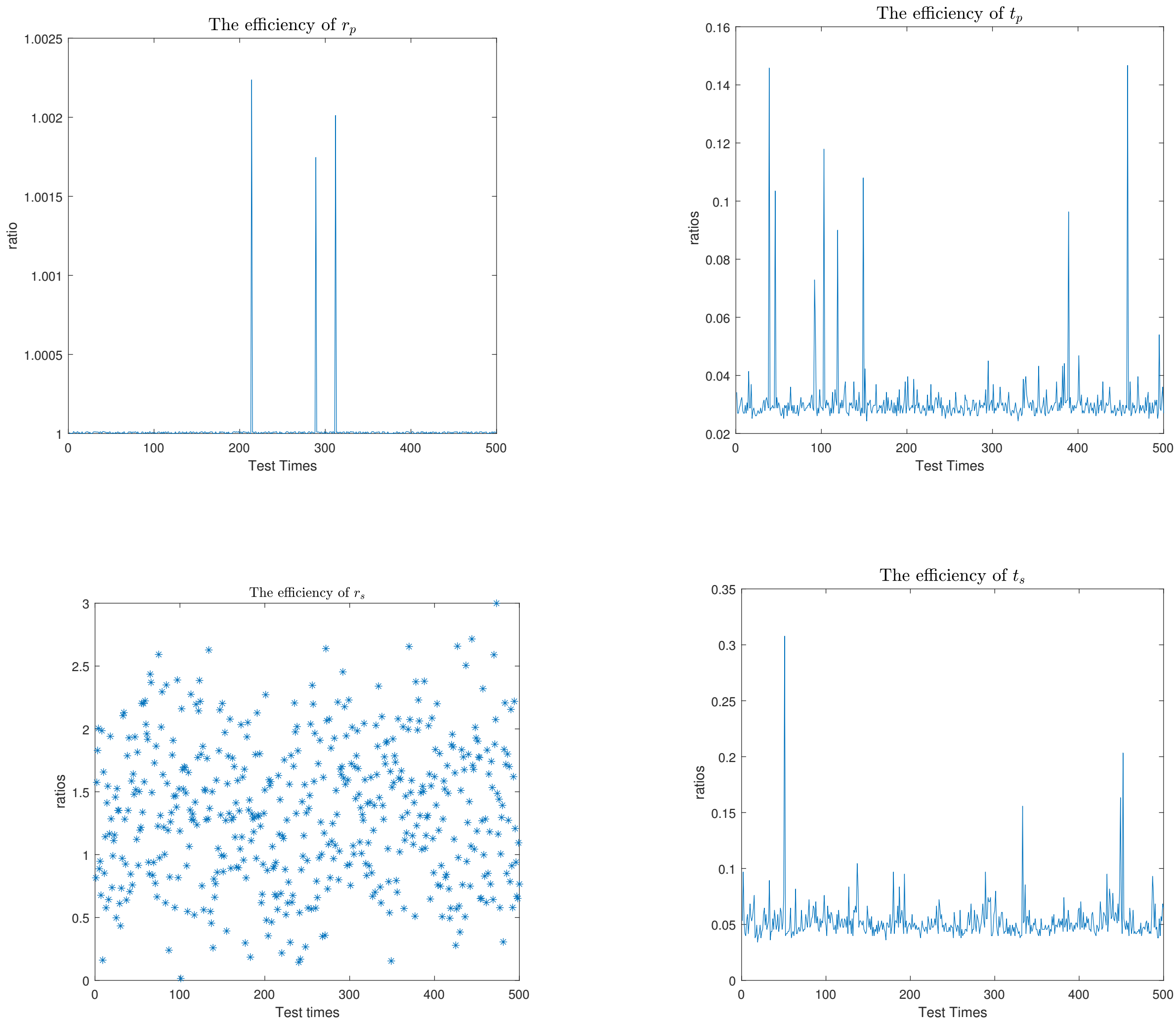

Figure 1. Efficiency of normwise condition estimators and CPU times of Algorithms 1 and 2.

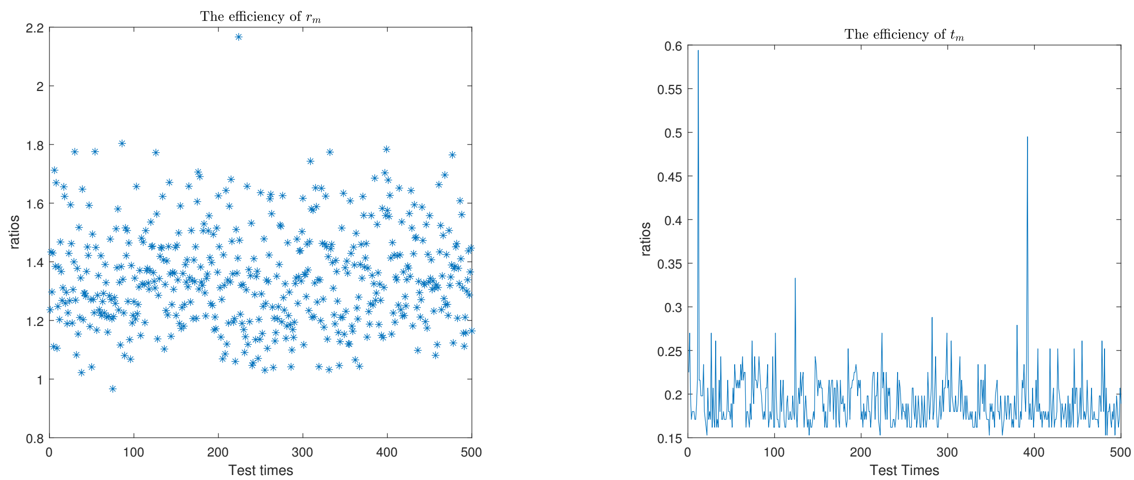

Figure 2. *Cont*.

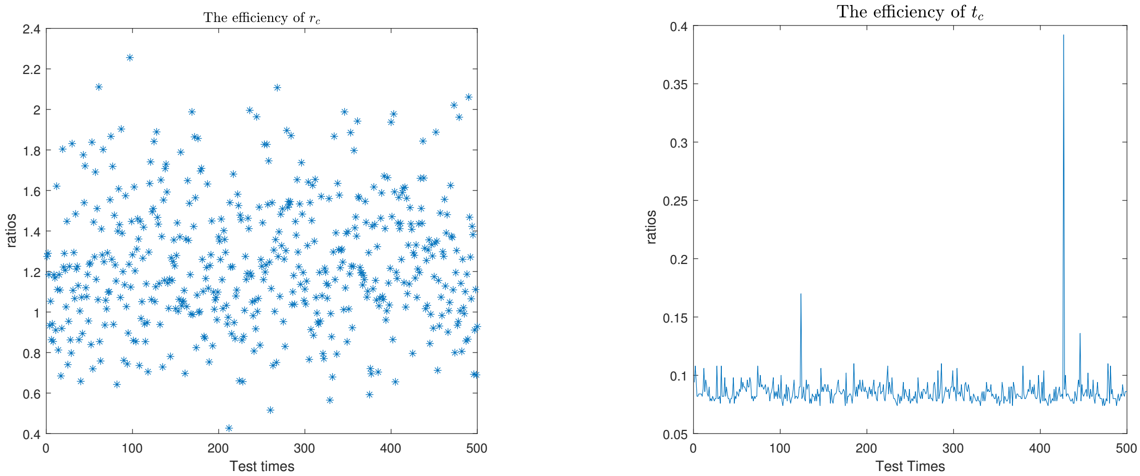

Figure 2. Efficiency of mixed and componentwise condition estimators and CPU time of Algorithm 3.

7. Conclusions

In this paper, we provided the explicit expressions and upper bounds for the normwise, mixed, and componentwise condition numbers for the generalized inverse $C_A^{\ddagger}$. Additionally, the corresponding results for the K-weighted pseudoinverse $L_K^{\dagger}$ can be obtained as a special case. We also show how to recover the previous condition numbers of the EILS solution from the generalized inverse $C_A^{\ddagger}$ condition numbers. We also developed the componentwise perturbation analysis of the EILS problem. Moreover, we designed three algorithms that efficiently estimate the normwise, mixed, and componentwise conditions for the generalized inverse $C_A^{\ddagger}$ using the probabilistic condition estimation method and the small-sample statistical condition estimation method. Finally, numerical results demonstrated the performance of these algorithms. In the future, we will continue our research on the MK-weighted generalized inverse.

Author Contributions: Methodology, Mahvish Samar; Investigation, Mahvish Samar and Xinzhong Zhu; writing—original draft, Mahvish Samar; review and editing, Abdul Shakoor.

Funding: This work was supported by the Natural Science Foundation of China (project no. 61976196), the Zhejiang Provincial Natural Science Foundation of China under Grant No.LZ22F030003.

Data Availability Statement:

Conflicts of Interest:

References

1. Cucker, F.; Diao, H.; Wei, Y. On mixed and componentwise condition numbers for Moore–penrose inverse and linear least squares problems. *Math. Comp.* **2007**, *76*, 947–963.
2. Liu, Q.; Pan, B.; Wang, Q. The hyperbolic elimination method for solving the equality constrained indefinite least squares problem. *Int. J. Comput. Math.* **2010**, *87*, 2953–2966.
3. Liu, Q.; Wang, M. Algebraic properties and perturbation results for the indefinite least squares problem with equality constraints. *Int. J. Comput. Math.* **2010**, *87*, 425–434.
4. Bjöorck, Å.; Higham, N.J.; Harikrishna, P. Solving the indefinite least squares problem by hyperbolic QR factorization. *SIAM J. Matrix Anal. Appl.* **2003**, *24*, 914–931.
5. Bjöorck, Å.; Higham, N.J.; Harikrishna, P. The equality constrained indefinite least squares problem: Theory and algorithms. *BIT Numer. Math.* **2003**, *43*, 505–517.
6. Wei, M.; Zhang, B. Structures and uniqueness conditions of *MK*-weighted pseudoinverses. *BIT* **1994**, *34*, 437–450.
7. Bjöorck, Å. Algorithms for indefinite linear least squares problems. *Linear Algebra Appl.* **2021**, *623*, 104–127.
8. Shi, C.; Liu, Q. A hyperbolic MGS elimination method for solving the equality constrained indefinite least squares problem. *Commun. Appl. Math. Comput.* **2011**, *25*, 65–73.

9. Mastronardi, N.; Dooren, P.V. An algorithm for solving the indefinite least squares problem with equality constraints. *BIT* **2014**, *54*, 201–218.
10. Mastronardi, N.; Dooren, P.V. A structurally backward stable algorithm for solving the indefinite least squares problem with equality constraints. *IMA J. Numer. Anal.* **2015**, *35*, 107–132.
11. Wang, Q. Perturbation analysis for generalized indefinite least squares problems. *J. East China Norm. Univ. Nat. Sci. Ed.* **2009**, *4*, 47–53.
12. Diao, H.; Zhou, T. Linearised estimate of the backward error for equality constrained indefinite least squares problems. *East Asian J. Appl. Math.* **2019**, *9*, 270–279.
13. Li, H.; Wang, S.; Yang, H. On mixed and componentwise condition numbers for indefinite least squares problem. *Linear Algebra Appl.* **2014**, *448*, 104–129.
14. Wang, S.; Meng, L. A contribution to the conditioning theory of the indefinite least squares problems. *Appl. Numer. Math.* **2022**, *17*, 137–159.
15. Skeel, R.D. Scaling for numerical stability in Gaussian elimination. *J. ACM* **1979**, *26*, 494–526.
16. Bjöorck, Å. Component-wise perturbation analysis and error bounds for linear least squares solutions. *BIT Numer. Math.* **1991**, *31*, 238–244.
17. Diao, H.; Liang, L.; Qiao, S. A condition analysis of the weighted linear least squares problem using dual norms. *Linear Algebra Appl.* **2018**, *66*, 1085–1103.
18. Diao, H.; Zhou, T. Backward error and condition number analysis for the indefinite linear least squares problem. *Int. J. Comput. Math.* **2019**, *96*, 1603–1622.
19. Diao, H. Condition numbers for a linear function of the solution of the linear least squares problem with equality constraints. *J. Comput. Appl. Math.* **2018**, *344*, 640–656.
20. Eldén, L. A weighted pseudoinverse, generalized singular values, and constrained least squares problems. *BIT* **1982**, *22*, 487–502.
21. Wei, M. Algebraic properties of the rank-deficient equality-constrained and weighted least squares problem. *Linear Algebra Appl.* **1992**, *161*, 27–43.
22. Gulliksson, M.E.; Wedin, P.A.; Wei, Y. Perturbation identities for regularized Tikhonov inverses and weighted pseudoinverses. *BIT* **2000**, *40*, 513–523.
23. Samar, M.; Li, H.; Wei, Y. Condition numbers for the K-weighted pseudoinverse $L_K^\dagger$ and their statistical estimation. *Linear Multilinear Algebra* **2021**, *69*, 752–770.
24. Burgisser, P.; Cucker, F. Condition: The geometry of numerical algorithms. In *Grundlehren der mathematischen Wissenschaften*; Springer: Heidelberg, Germany, 2013; Volume 349.
25. Rice, J. A theory of condition. *SIAM J. Numer. Anal.* **1966**, *3*, 287–310.
26. Gohberg, I.; Koltracht, I. Mixed, componentwise, and structured condition numbers. *SIAM J. Matrix Anal. Appl.* **1993**, *14*, 688–704.
27. Hochstenbach, M.E. Probabilistic upper bounds for the matrix two-norm. *J. Sci. Comput.* **2013**, *57*, 464–476.
28. Kenney, C.S.; Laub, A.J. Small-sample statistical condition estimates for general matrix functions. *SIAM J. Sci. Comput.* **1994**, *15*, 36–61.
29. Xie, Z.; Li, W.; Jin, X. On condition numbers for the canonical generalized polar decomposition of real matrices. *Electron. J. Linear Algebra* **2013**, *26*, 842–857.
30. Horn, R.A.; Johnson, C.R. *Topics in Matrix Analysis*; Cambridge University Press: New York, NY, USA, 1991.
31. Magnus, J.R.; Neudecker, H. *Matrix Differential Calculus with Applications in Statistics and Econometrics*, 3rd ed.; John Wiley and Sons: Chichester, UK, 2007.
32. Diao, H.; Xiang, H.; Wei, Y. Mixed, componentwise condition numbers and small sample statistical condition estimation of Sylvester equations. *Numer. Linear Algebra Appl.* **2012**, *19*, 639–654.
33. Bürgisser, P.; Cucker, R. *Condition: The Geometry of Numerical Algorithms*; Springer: Heidelberg, Germany, 2013.
34. Li, H.; Wang, S. On the partial condition numbers for the indefnite least squares problem. *Appl. Numer. Math.* **2018**, *123*, 200–220.
35. Samar, M. Condition numbers for a linear function of the solution to the constrained and weighted least squares problem and their statistical estimation. *Taiwan. J. Math.* **2021**, *25*, 717–741.
36. Samar, M.; Lin, F. Perturbation and condition numbers for the Tikhonov regularization of total least squares problem and their statistical estimation. *J. Comput. Appl. Math.* **2022**, *411*, 114230.
37. Diao, H.; Wei, Y.; Xie, P. Small sample statistical condition estimation for the total least squares problem. *Numer. Algor.* **2017**, *75*, 435–455.
38. Samar, M.; Zhu, X. Structured conditioning theory for the total least squares problem with linear equality constraint and their estimation. *AIMS Math.* **2023**, *8*, 11350–11372.
39. Baboulin, M.; Gratton, S.; Lacroix, R.; Laub, A.J. Statistical estimates for the conditioning of linear least squares problems. *Lect Notes Comput. Sci.* **2014**, *8384*, 124–133.
40. Higham, N.J. *Accuracy and Stability of Numerical Algorithms*, 2nd ed.; SIAM: Philadelphia, PA, USA, 2002.
41. Higham, N.J. J-Orthogonal matrices: Properties and generation. *SIAM Rev.* **2003**, *45*, 504–519.