Journal of Scientific Computing (2025) 104:68
https://doi.org/10.1007/s10915-025-02978-y

An Improved Shifted CholeskyQR Based on Columns

Yuwei Fan[1] · **Haoran Guan**[2] · **Zhonghua Qiao**[2]

Received: 12 August 2024 / Revised: 6 February 2025 / Accepted: 22 June 2025 /
Published online: 4 July 2025

Abstract
Among all the deterministic CholeskyQR-type algorithms, Shifted CholeskyQR3 is specifically designed to address the QR factorization of ill-conditioned matrices. This algorithm introduces a shift parameter s to prevent failure during the initial Cholesky factorization step, making the choice of this parameter critical for the algorithm's effectiveness. Our goal is to identify a smaller s compared to the traditional selection based on $\|X\|_2$. In this research, we propose a new matrix norm called the g-norm, which is based on the column properties of X. This norm allows us to obtain a reduced shift parameter s for the Shifted CholeskyQR3 algorithm, thereby improving the sufficient condition of $\kappa_2(X)$ for this method. We provide rigorous proofs of orthogonality and residuals for the improved algorithm using our proposed s. Numerical experiments confirm the enhanced numerical stability of orthogonality and residuals with the reduced s. We find that Shifted CholeskyQR3 can effectively handle ill-conditioned X with a larger $\kappa_2(X)$ when using our reduced s compared to the original s. Furthermore, we compare CPU times with other algorithms to assess performance improvements.

Keywords QR factorization · Rounding error analysis · Improved Shifted CholeskyQR3

Mathematics Subject Classification 65F30 · 15A23 · 65F25 · 65G50

1 Introduction

As a fundamental component of matrix decomposition, QR factorization plays a crucial role in various real-world applications across academia and industry. These applications include randomized singular value decomposition [12, 19], Krylov subspace methods [15], the local optimal block preconditioned conjugate gradient method (LOBPCG) [7], and block HouseholderQR algorithms [24], among others.

✉ Zhonghua Qiao
zhonghua.qiao@polyu.edu.hk

Yuwei Fan
fanyuwei2@huawei.com

Haoran Guan
21037226R@connect.polyu.hk

[1] Theory Lab, Huawei Hong Kong Research Center, Sha Tin, Hong Kong

[2] Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Hong Kong

1.1 CholeskyQR2 and its Properties

Typical algorithms for QR factorization include CGS, MGS, HouseholderQR and TSQR. For details, see [1, 5, 10, 13, 17, 20]. In recent years, a new algorithm called CholeskyQR has been developed. When $X \in \mathbb{R}^{m\times n}$, $m \geq n$, CholeskyQR begins by computing a Gram matrix $B \in \mathbb{R}^{n\times n}$, followed by performing a Cholesky factorization to obtain an upper-triangular matrix $R \in \mathbb{R}^{n\times n}$. The orthogonal factor $Q \in \mathbb{R}^{m\times n}$ can then be computed. Due to the structure of Cholesky factorization, X needs to be full rank, that is rank(X) $= n$. Algorithm 1 illustrates the fundamental version of CholeskyQR that computes the QR factorization as follows.

Algorithm 1 $[Q, R] = \text{CholeskyQR}(X)$

1: $B = X^\top X$,
2: $R = \text{Cholesky}(B)$,
3: $Q = XR^{-1}$.

Compared to TSQR, HouseholderQR, MGS, and CGS, the CholeskyQR algorithm has several advantages. It has only half the computational cost of TSQR and HouseholderQR. Additionally, it requires significantly fewer reductions in a parallel environment than the other algorithms. Moreover, CholeskyQR utilizes BLAS3 operations, which are more difficult to implement for other algorithms.

Although with many advantages, Algorithm 1 exhibits certain limitations and is rarely used directly. When considering the error of orthogonality, it is shown in [29] that

$$\|Q^\top Q - I\|_F \leq \frac{5}{64}\delta^2 \tag{1}$$

where

$$\delta = 8\kappa_2(X)\sqrt{mn\mathbf{u} + n(n+1)\mathbf{u}}. \tag{2}$$

Here, $\kappa_2(X) = \frac{\sigma_1(X)}{\sigma_n(X)}$ is the condition number of X. $\sigma_i(X)$ is the i-th largest singular value of X for $i = 1, 2, 3, ..., n$. Specifically, $\sigma_1(X) = \|X\|_2$. $\mathbf{u}$ is the unit roundoff and $\mathbf{u} = 2^{-53}$. In CholeskyQR-type algorithms, regarding the sizes of the matrices, we always define

$$mn\mathbf{u} \leq \frac{1}{64}, \tag{3}$$

$$n(n+1)\mathbf{u} \leq \frac{1}{64}. \tag{4}$$

According to (1) and (2), the orthogonality error of Algorithm 1 is proportional to $(\kappa_2(X))^2$. Numerous numerical experiments indicate that Algorithm 1 is numerically stable only when the input X is very well-conditioned. Consequently, a new algorithm, named CholeskyQR2, has been developed by performing two iterations of the CholeskyQR algorithm [9]. It is presented in Algorithm 2.

In [29], it has been shown that compared to Algorithm 1, Algorithm 2 is numerically stable in both orthogonality and residual. The following lemma holds.

Lemma 1 *(Rounding error analysis of CholeskyQR2) For* $X \in \mathbb{R}^{m\times n}$ *and* $[Q_1, R_2] = \text{CholeskyQR2}(X)$, *when* $8\kappa_2(X)\sqrt{mn\mathbf{u} + n(n+1)\mathbf{u}} \leq 1$, *we have*

$$\|Q_1^\top Q_1 - I\|_F \leq 6(mn\mathbf{u} + n(n+1)\mathbf{u}), \tag{5}$$

Algorithm 2 $[Q_1, R_2] = \text{CholeskyQR2}(X)$

1: $[Q, R] = \text{CholeskyQR}(X)$,
2: $[Q_1, R_1] = \text{CholeskyQR}(Q)$,
3: $R_2 = R_1 R$.

$$\|Q_1 R_2 - X\|_F \leq 5n^2 \mathbf{u} \|X\|_2. \tag{6}$$

Here, the smallness of (2) is a sufficient condition for Algorithm 2, indicating that Algorithm 2 is reliable when the input X is not ill-conditioned.

1.2 Shifted CholeskyQR3 and its Problems

When X is ill-conditioned, Algorithm 2 may encounter numerical breakdown due to rounding errors. To address this challenge, researchers have introduced an improved algorithm known as Shifted CholeskyQR (SCholeskyQR), which is detailed in Algorithm 3 [8].

Algorithm 3 $[Q, R] = \text{SCholeskyQR}(X)$

1: $B = X^\top X$,
2: choose $s > 0$,
3: $R = \text{Cholesky}(B + sI)$,
4: $Q = XR^{-1}$.

Algorithm 3 is a superior algorithm in terms of applicability compared to Algorithm 1. The concept behind the algorithm is straightforward. For an ill-conditioned matrix $B \in \mathbb{R}^{n\times n}$, the addition of a scaled identity matrix reduces $\kappa_2(B + sI)$ and prevents numerical breakdown. To further improve the numerical stability, CholeskyQR2 is performed subsequently, and a new algorithm called Shifted CholeskyQR3 (SCholeskyQR3) has been developed, which is given in Algorithm 4.

Algorithm 4 $[Q_2, R_4] = \text{SCholeskyQR3}(X)$

1: $[Q, R] = \text{SCholeskyQR}(X)$,
2: $[Q_1, R_1] = \text{CholeskyQR}(Q)$,
3: $R_2 = R_1 R$,
4: $[Q_2, R_3] = \text{CholeskyQR}(Q_1)$,
5: $R_4 = R_3 R_2$.

Regarding Algorithms 3 and 4, we have the following theoretical results from [8].

Lemma 2 *(Rounding error analysis of Shifted CholeskyQR) For $X \in \mathbb{R}^{m\times n}$ and $[Q, R] = \text{SCholeskyQR}(X)$, with $11(mn\mathbf{u} + n(n+1)\mathbf{u})\|X\|_2^2 \leq s \leq \frac{1}{100}\|X\|_2^2$ and $\kappa_2(X) \leq \frac{1}{6n^2\mathbf{u}}$, we have*

$$\|Q^\top Q - I\|_2 \leq 2, \tag{7}$$
$$\|QR - X\|_F \leq 2n^2 \mathbf{u} \|X\|_2. \tag{8}$$

Lemma 3 *(The relationship between $\kappa_2(X)$ and $\kappa_2(Q)$ for Shifted CholeskyQR) For $X \in \mathbb{R}^{m\times n}$ and $[Q, R] = SCholeskyQR(X)$, with $11(mn\mathbf{u} + n(n+1)\mathbf{u})\|X\|_2^2 \le s \le \frac{1}{100}\|X\|_2^2$ and $\kappa_2(X) \le \frac{1}{6n^2\mathbf{u}}$, we have*

$$\kappa_2(Q) \le 2\sqrt{3}\cdot\sqrt{1+\alpha(\kappa_2(X))^2}. \tag{9}$$

Here, $\alpha = \frac{s}{\|X\|_2^2}$. When $[Q_2, R_4] = SCholeskyQR3(X)$, if we take $s = 11(mn\mathbf{u} + n(n+1)\mathbf{u})\|X\|_2^2$ and $\kappa_2(X)$ is large enough, a sufficient condition for $\kappa_2(X)$ is

$$\kappa_2(X) \le \frac{1}{96(mn\mathbf{u} + n(n+1)\mathbf{u})}. \tag{10}$$

Lemma 4 *(Rounding error analysis of Shifted CholeskyQR3) For $X \in \mathbb{R}^{m\times n}$ and $[Q_2, R_4] = SCholeskyQR3(X)$, with $s = 11(mn\mathbf{u} + n(n+1)\mathbf{u})\|X\|_2^2$ and (10), we have*

$$\|Q_2^\top Q_2 - I\|_F \le 6(mn\mathbf{u} + n(n+1)\mathbf{u}), \tag{11}$$
$$\|Q_2 R_4 - X\|_F \le 15n^2\mathbf{u}\|X\|_2. \tag{12}$$

In particular, Lemma 3 is one of the most important properties of Shifted CholeskyQR3. It shows that when s is located in a certain interval, the larger s we take, the larger $\kappa_2(Q)$ will become. Since CholeskyQR2 following Shifted CholeskyQR will break down if $\kappa_2(Q)$ is large, the selection of the shifted parameter s is a crucial aspect of Shifted CholeskyQR3. It can neither be too large, considering the applicability of Shifted CholeskyQR3, nor too small, since Shifted CholeskyQR may break down.

In fact, when we express the first two steps of Algorithm 3 with error analysis as follows:

$$B = X^\top X + E_A, \tag{13}$$
$$R^\top R = B + E_B + sI, \tag{14}$$

we find that the error bounds for $\|E_A\|_2$ and $\|E_B\|_2$ in (13) and (14) significantly influence the choice of the parameter s. In [8], the original s is set to be 10 times the sum of $\|E_A\|_2$ and $\|E_B\|_2$. Previous researchers have used $\|X\|_2$ to bound the 2-norm of each column of X when estimating $\|E_A\|_2$ and $\|E_B\|_2$. However, in practice, both $\|E_A\|_2$ and $\|E_B\|_2$ tend to be overestimated. This overestimation results in a relatively large value of s, which gives a more stringent sufficient condition for $\kappa_2(X)$ in the context of Shifted CholeskyQR3, based on (9), (10), and the corresponding analytical steps outlined in [8]. This condition will limit the applicability of Shifted CholeskyQR3 to ill-conditioned matrices, as demonstrated by numerous numerical experiments. In most of the cases, the 2-norm of each column of X can be significantly smaller than $\|X\|_2$. Therefore, the primary objective of this work is to select a smaller shifted parameter s for Shifted CholeskyQR3 and to demonstrate that this improved s can ensure the numerical stability of the algorithm. We aim to provide a more accurate error estimation for the residuals of Shifted CholeskyQR3 theoretically. The revised choice of s improves the applicability of Shifted CholeskyQR3, which is reflected in a better sufficient condition for $\kappa_2(X)$ to some extent.

1.3 Our Contributions in this Work

In this work, we calculate the largest 2-norm among all the columns of X, which is defined as $\|X\|_g$ in Definition 1.

Definition 1 (The definition of the g-norm) For $X = [X_1, X_2, \cdots X_{n-1}, X_n] \in R^{m\times n}$,

$$\|X\|_g := \max_{1\le j\le n} \|X_j\|_2. \tag{15}$$

where

$$\|X_j\|_2 = \sqrt{x_{1,j}^2 + x_{2,j}^2 + ... + x_{m-1,j}^2 + x_{m,j}^2}.$$

We introduce several properties of the g-norm of the matrix in Section 3, which offer a new perspective on rounding error analysis. Using the g-norm, we can estimate $\|E_A\|_2$ and $\|E_B\|_2$ with tighter upper bounds based on $\|X\|_g$. Consequently, a smaller s with $\|X\|_g$ can be chosen as $s = 11(mn\mathbf{u} + n(n+1)\mathbf{u})\|X\|_g^2$ for Shifted CholeskyQR3. Regarding the g-norm, we define a constant p as

$$p = \frac{\|X\|_g}{\|X\|_2}. \tag{16}$$

Here, $\frac{1}{\sqrt{n}} \le p \le 1$. We present the following theorems related to the improved Shifted CholeskyQR (ISCholeskyQR) and improved Shifted CholeskyQR3 (ISCholeskyQR3).

Theorem 1 *(Rounding error analysis of the improved Shifted CholeskyQR) For $X \in \mathbb{R}^{m\times n}$ and $[Q, R] = ISCholeskyQR(X)$, with $11(mn\boldsymbol{u} + n(n+1)\boldsymbol{u})\|X\|_g^2 \le s \le \frac{1}{100}\|X\|_g^2$ and $\kappa_2(X) \le \frac{1}{4.89pn^2\mathbf{u}}$, we have*

$$\|Q^\top Q - I\|_2 \le 1.6, \tag{17}$$

$$\|QR - X\|_F \le 1.67pn^2\mathbf{u}\|X\|_2. \tag{18}$$

Theorem 2 *(The relationship between $\kappa_2(X)$ and $\kappa_2(Q)$ for the improved Shifted CholeskyQR) For $X \in \mathbb{R}^{m\times n}$ and $[Q, R] = ISCholeskyQR(X)$, with $11(mn\boldsymbol{u} + n(n+1)\boldsymbol{u})\|X\|_g^2 \le s \le \frac{1}{100}\|X\|_g^2$ and $\kappa_2(X) \le \frac{1}{4.89pn^2\mathbf{u}}$, we have*

$$\kappa_2(Q) \le 3.24\sqrt{1 + t(\kappa_2(X))^2}. \tag{19}$$

When $[Q_2, R_4] = ISCholeskyQR3(X)$, if we take $s = 11(mn\boldsymbol{u} + n(n+1)\boldsymbol{u})\|X\|_g^2$ and $\kappa_2(X)$ is large enough, a sufficient condition for $\kappa_2(X)$ is

$$\kappa_2(X) \le \frac{1}{86p(mn\mathbf{u} + (n+1)n\mathbf{u})} \le \frac{1}{4.89pn^2\mathbf{u}}. \tag{20}$$

Here, we define

$$t = \frac{s}{\|X\|_2^2} \le \frac{1}{100}. \tag{21}$$

Theorem 3 *(Rounding error analysis of the improved Shifted CholeskyQR3) For $X \in \mathbb{R}^{m\times n}$ and $[Q_2, R_4] = ISCholeskyQR3(X)$, with $s = 11(mn\boldsymbol{u} + n(n+1)\boldsymbol{u})\|X\|_g^2$ and (20), we have*

$$\|Q_2^\top Q_2 - I\|_F \le 6(mn\mathbf{u} + n(n+1)\mathbf{u}), \tag{22}$$

$$\|Q_2R_4 - X\|_F \le (6.57p + 4.87)n^2\mathbf{u}\|X\|_2. \tag{23}$$

Table 1 Comparison of $\kappa_2(X)$ between the improved and the original s

s	Sufficient condition of $\kappa_2(X)$	Upper bound of $\kappa_2(X)$
$11(mn\mathbf{u}+n(n+1)\mathbf{u})\|X\|_2^2$	$\frac{1}{96(mn\mathbf{u}+n(n+1)\mathbf{u})}$	$\frac{1}{6n^2\mathbf{u}}$
$11(mn\mathbf{u}+n(n+1)\mathbf{u})\|X\|_g^2$	$\frac{1}{86p(mn\mathbf{u}+n(n+1)\mathbf{u})}$	$\frac{1}{4.89pn^2\mathbf{u}}$

Table 2 Comparison of the upper bounds between the improved and the original s

s	SCholeskyQR	SCholeskyQR3
$11(mn\mathbf{u}+n(n+1)\mathbf{u})\|X\|_2^2$	$2n^2\mathbf{u}\|X\|_2$	$15n^2\mathbf{u}\|X\|_2$
$11(mn\mathbf{u}+n(n+1)\mathbf{u})\|X\|_g^2$	$1.6n^2\mathbf{u}\|X\|_g$	$(6.57p+4.87)n^2\mathbf{u}\|X\|_2$

Theorems 1-3 correspond to Lemmas 2-4, respectively, which are proved in Sections 4.4-4.6. These theorems demonstrate that the improved Shifted CholeskyQR3 has a better sufficient condition of $\kappa_2(X)$ compared to the original one. Consequently, the improved Shifted CholeskyQR3 can effectively handle X with larger $\kappa_2(X)$, as shown in Table 1 and the numerical experiments in Section 5. The property of the R-factor can also be described by the g-norm, which will loosen the upper bound for $\kappa_2(X)$ in the existing results in [8]. From a theoretical perspective, we prove the numerical stability of the algorithm when using an improved s in Section 4 and provide tighter theoretical upper bounds for the residual $\|Q_2R_4 - X\|_F$ using the properties of the g-norm under such cases compared to the original one in [8], seeing Table 2 for detailed comparisons. This provides new insights into the problem of rounding error analysis.

Defining $\|X\|_g$ offers several advantages. In many cases, when the size of X is large, *e.g.*, $m > 10^5$ or $n > 10^4$, X tends to be sparse for storage efficiency. In such scenarios, calculating the norms of the matrix can be computationally expensive. The properties of the g-norm allow us to select an s based on key elements of X without the need to compute the norms of the entire large matrix. Furthermore, the g-norm enables better utilization of the matrix structure and the inherent properties of its elements, while the 2-norm primarily highlights the general characteristics of the matrix. We plan to leverage these properties for further exploration of CholeskyQR-type algorithms in our future works. In other words, the definition of the g-norm offers a novel approach to rounding error analysis for matrices, based on their structures and elements. Although this perspective is not directly evident from numerical experiments, it represents an innovative advancement compared to existing results.

1.4 Outline

The rest of the paper is organized as follows. Section 2 reviews existing results on rounding error analysis. In Section 3, we examine some properties of the g-norm. The proof of the theoretical results for the improved Shifted CholeskyQR3 is presented in Section 4, which serves as the key contribution of this work. Next, Section 5 provides numerical results and compares them with several existing algorithms. Finally, concluding remarks are offered in Section 6.

2 Preliminary Lemmas of Rounding Error Analysis

Before presenting our main results, we introduce the following preliminary lemmas for rounding error analysis.

Lemma 5 *(Weyl's Theorem [10]) For matrices $A, B, C \in \mathbb{R}^{m\times n}$, if we have $A + B = C$, then*

$$|\sigma_i(A) - \sigma_i(B)| \le \sigma_1(C) = \|C\|_2.$$

where $\sigma_i(X)$ is the i-th greatest singular value of X, with $i = 1, 2, \cdots, \min(m, n)$.

Lemma 6 *(Rounding error in matrix multiplications [13]) For $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{n\times p}$, the error in computing the matrix product $C = AB$ in floating-point arithmetic is bounded by*

$$|AB - fl(AB)| \le \gamma_n |A||B|.$$

Here, $|A|$ is the matrix whose (i, j) element is $|a_{ij}|$ and

$$\gamma_n := \frac{n\mathbf{u}}{1 - n\mathbf{u}} \le 1.02n\mathbf{u}.$$

Lemma 7 *(Rounding error in Cholesky factorization [13]) When $A \in \mathbb{R}^{n\times n}$ is symmetric positive definite, its output R after Cholesky factorization in floating-point arithmetic satisfies*

$$R^\top R = A + \Delta A, \quad |\Delta A| \le \gamma_{n+1} |R^\top||R|.$$

Lemma 8 *(Rounding error in solving triangular systems [13]) If $R \in \mathbb{R}^{n\times n}$ is a nonsingular upper-triangular matrix, the computed solution x obtained by solving an upper-triangular linear system $Rx = b$ by back substitution in floating-point arithmetic satisfies*

$$(R + \Delta R)x = b, \quad |\Delta R| \le \gamma_n |R|.$$

To learn more about matrix perturbations, readers can refer to references [6, 11, 25].

3 Some Properties of the *g*-norm

In this section, we introduce and prove several properties of the g-norm following Definition 1. These properties will be utilized in the theoretical analysis of the improved Shifted CholeskyQR3.

Lemma 9 *(Connections between the g-norm and other norms) If $A \in \mathbb{R}^{m\times p}$, $B \in \mathbb{R}^{p\times n}$, then we have*

$$\|AB\|_g \le \|A\|_2 \|B\|_g, \|AB\|_g \le \|A\|_F \|B\|_g. \tag{24}$$

Proof Regarding $\|AB\|_g$, with Definition 1, we have

$$\begin{aligned}
\|AB\|_g &\le \max(\|AB_1\|_2, \|AB_2\|_2, \cdots, \|AB_n\|_2) \\
&\le \max(\|A\|_2\|B_1\|_2, \|A\|_2\|B_2\|_2, \cdots, \|A\|_2\|B_n\|_2) \\
&\le \|A\|_2 \cdot \max(\|B_1\|_2, \|B_2\|_2, \cdots, \|B_n\|_2) \\
&\le \|A\|_2 \|B\|_g.
\end{aligned}$$

Here, the first inequality of (24) is received. Since $\|A\|_2 \le \|A\|_F$, it is easy to get the second inequality of (24). □

Lemma 10 *(The triangular inequality of the g-norm) For $A, B, C \in \mathbb{R}^{m\times n}$, when $A = B+C$, we have*

$$\|A\|_g \leq \|B\|_g + \|C\|_g. \tag{25}$$

Proof Based on Definition 1 and the triangular inequality of the norms of vectors, we can easily get (25). □

Lemma 11 *(The relationship between different norms) For $A \in \mathbb{R}^{m\times n}$, we have*

$$\|A\|_g \leq \|A\|_2 \leq \|A\|_F. \tag{26}$$

Proof The left inequality is based on the property of the singular values of the matrix. The right inequality is obvious. □

4 Theoretical Analysis of the Improved Shifted CholeskyQR3

In this section, we provide the theoretical analysis of the improved Shifted CholeskyQR3 with a smaller s. We present the relevant settings and lemmas for our algorithm, and we also prove Theorems 1, 2 and 3.

4.1 General Settings and Assumptions

Given the presence of rounding errors at each step of the algorithm, we express Algorithm 3 with error matrices as follows:

$$B = X^\top X + E_A, \tag{27}$$
$$R^\top R = B + sI + E_B, \tag{28}$$
$$q_i^\top = x_i^\top (R + E_{Ri})^{-1}, \tag{29}$$
$$X = QR + E_X. \tag{30}$$

We let $q_i^\top$ and $x_i^\top$ represent the i-th rows of X and Q respectively. The error matrix E_A in (27) denotes the discrepancy generated when calculating the Gram matrix $X^\top X$. Similarly, E_B in (28) represents the error matrix after performing Cholesky factorization on B with a shifted item. As noted in [8], since R may be non-invertible, we describe the last step of Algorithm 3 in terms of each row of the matrices in (29), where E_{Ri} denotes the rounding error for the R factor. If we write the last step of Algorithm 3 without R^{-1}, the general error matrix of QR factorization is given by E_X in (30). A crucial aspect of the subsequent analysis is establishing connections between E_X and E_{Ri}.

Under (15), we provide a new interval of the shifted item s based on $\|X\|_g$. If $X \in \mathbb{R}^{m\times n}$, we have the following settings:

$$mn\mathbf{u} \leq \frac{1}{64}, \tag{31}$$
$$n(n+1)\mathbf{u} \leq \frac{1}{64}, \tag{32}$$
$$4.89n^2\mathbf{u} \cdot p\kappa_2(X) \leq 1, \tag{33}$$
$$11(mn\mathbf{u} + n(n+1)\mathbf{u})\|X\|_g^2 \leq s \leq \frac{1}{100}\|X\|_g^2. \tag{34}$$

Here, p is defined in (16). We observe that, compared to the original Shifted CholeskyQR based on $\|X\|_2$, the range of $\kappa_2(X)$ expands with a constant p related to n as indicated in

(33). Furthermore, (34) demonstrates that the new s is still constrained by a relative large upper bound. The applicability of this new s can be established using a method similar to those in [16, 26, 32].

4.2 Algorithms

In this section, we present the improved Shifted CholeskyQR (ISCholeskyQR) and the improved Shifted CholeskyQR3 (ISCholeskyQR3). They are detailed in Algorithm 5 and Algorithm 6, respectively.

Algorithm 5 $[Q, R] = \text{ISCholeskyQR}(X)$

1: calculate the norm of each column for X, $\|X_j\|$, $j = 1, 2, 3, \cdots n$,
2: pick $\|X\|_g = \max_{1\le j\le n} \|X_j\|$,
3: choose $s = 11(mn\mathbf{u} + (n+1)n\mathbf{u})\|X\|_g^2$,
4: $[Q, R] = \text{SCholeskyQR}(X)$.

Algorithm 6 $[Q_2, R_4] = \text{ISCholeskyQR3}(X)$

1: calculate the norm of each column for X, $\|X_j\|$, $j = 1, 2, 3, \cdots n$,
2: pick $\|X\|_g = \max_{1\le j\le n} \|X_j\|$,
3: choose $s = 11(mn\mathbf{u} + (n+1)n\mathbf{u})\|X\|_g^2$,
4: $[Q, R] = \text{SCholeskyQR}(X)$,
5: $[Q_1, R_1] = \text{CholeskyQR}(Q)$,
6: $R_2 = R_1 R$,
7: $[Q_2, R_3] = \text{CholeskyQR}(Q_1)$,
8: $R_4 = R_3 R_2$.

4.3 Some Lemmas for Proving Theorems

To prove Theorems 1- 3, we require the following lemmas. These theoretical results resemble those in [8] and their proofs closely follow those of [8]. However, by utilizing the definition of the g-norm and its properties, we can enhance many of the upper bounds for these results. We will discuss these improvements in detail below.

Lemma 12 *For E_A and E_B in (27) and (28), if (34) is satisfied, we have*

$$\|E_A\|_2 \le 1.1mn\mathbf{u}\|X\|_g^2, \tag{35}$$

$$\|E_B\|_2 \le 1.1n(n+1)\boldsymbol{u}\|X\|_g^2. \tag{36}$$

Proof In this section, we aim to estimate $\|E_A\|_2$ and $\|E_B\|_2$ using $\|X\|_g$ instead of $\|X\|_2$. While our analysis follows a similar approach to that in [8, 29], our new definition of the g-norm allows us to provide improved estimations for the 2-norms of the error matrices.

We can estimate $\|E_A\|_F$ first since $\|E_A\|_2$ is bounded by $\|E_A\|_F$. With Lemma 6 and (27), we have $B = fl(X^\top X)$. Therefore, we have

$$|E_A| = |B - X^\top X|$$

$$\leq \gamma_m |X^\top||X|. \tag{37}$$

With (37), for E_{Aij} which denotes the element of E_A in the i-th row and the j-th column, we have

$$|E_{Aij}| \leq \gamma_m |X_i||X_j|. \tag{38}$$

Here, X_i denotes the i-th column of X. We combine (38) with (15) and have

$$|E_{Aij}| \leq \gamma_m \|X\|_g^2. \tag{39}$$

Since $\|E_A\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^n (|E_{Aij}|)^2}$, with (39), we can bound $\|E_A\|_2$ as

$$\begin{aligned}\|E_A\|_2 \leq \||E_A|\|_F &\leq \gamma_m \sqrt{\sum_{i=1}^n \sum_{j=1}^n (|E_{Aij}|)^2} \\ &\leq \gamma_m n \|X\|_g^2 \\ &\leq 1.1mn\mathbf{u}\|X\|_g^2.\end{aligned}$$

Then, (35) is proved. (35) is a more accurate estimation of $\|E_A\|_2$ compared to that in [8, 29] since $\|X\|_g \leq \|X\|_2$.

When estimating $\|E_B\|_F$, we focus on (28). We use the same idea as that in [8, 29] for this estimation. With (15), we have

$$\|R\|_F^2 = \||R|\|_F^2 \leq n\|R\|_g^2. \tag{40}$$

Using Lemma 7, Lemma 10, (27), (28) and (40), we can get

$$\begin{aligned}\|E_B\|_2 \leq \||E_B|\|_F &\leq \gamma_{n+1}\|R\|_F^2 \\ &\leq \gamma_{n+1} \cdot n\|R\|_g^2 \\ &\leq \gamma_{n+1} \cdot n(\|X\|_g^2 + s + \|E_A\|_2 + \|E_B\|_2).\end{aligned} \tag{41}$$

With (31), (32), (34), (35) and (41), we can bound $\|E_B\|_2$ as

$$\begin{aligned}\|E_B\|_2 &\leq \frac{\gamma_{n+1}n(1+\gamma_m n + t_1)}{1-\gamma_{n+1}n}\|X\|_g^2 \\ &\leq \frac{1.02(n+1)\mathbf{u} \cdot n(1+1.02m\mathbf{u} \cdot n + 0.01)}{1-1.02(n+1)\mathbf{u} \cdot n}\|X\|_g^2 \\ &\leq \frac{1.02 \cdot n(n+1)\mathbf{u} \cdot (1+1.02 \cdot \frac{1}{64} + 0.01)}{1-\frac{1.02}{64}}\|X\|_g^2 \\ &\leq 1.1n(n+1)\mathbf{u}\|X\|_g^2.\end{aligned}$$

(36) is proved. Here, we define $t_1 = \frac{s}{\|X\|_2^2} \leq 0.01$ based on (34). In all, Lemma 12 is proved. □

Remark 1 The last step of (41) relies on Lemma 10 and Lemma 11. While the approach for estimating $\|E_B\|_2$ parallels that in [8, 29], we utilize the relationships between the 2-norm and the g-norm established in Lemma 11, which derive from a distinctly different perspective on the norms of matrices compared to the original work on CholeskyQR-type algorithms.

Lemma 13 *For R^{-1} and XR^{-1} from (29), when (34) is satisfied, we have*

$$\|R^{-1}\|_2 \leq \frac{1}{\sqrt{(\sigma_n(X))^2 + 0.9s}}, \tag{42}$$

$$\|XR^{-1}\|_2 \leq 1.5. \tag{43}$$

Proof The steps of analysis to get (42) and (43) are similar to those in [8]. Lemma 13 is proved. □

Lemma 14 *For E_{Ri} from (29), when (34) is satisfied, we have*

$$\|E_{Ri}\|_2 \leq 1.03n\sqrt{n}\boldsymbol{u}\|X\|_g. \tag{44}$$

Proof The steps to get (44) are similar to those in [8]. However, the property of the g-norm can provide a tighter bound for $\|E_{Ri}\|_2$. For $1 \leq i \leq m$, based on Lemma 8, we have

$$\|E_{Ri}\|_2 \leq \|E_{Ri}\|_F \leq \gamma_n\sqrt{n}\|R\|_g. \tag{45}$$

Based on the properties of Cholesky factorization and the structure of the algorithm, we find that the square of the g-norm of the matrix corresponds to the largest entry on the diagonal of the Gram matrix. With Lemma 11, (27), (28) and (34), we obtain

$$\|R\|_g^2 \leq \|X\|_g^2 + s + (\|E_A\|_2 + \|E_B\|_2) \leq 1.01\|X\|_g^2. \tag{46}$$

With (46), it is easy to see that

$$\|R\|_g \leq 1.006\|X\|_g. \tag{47}$$

Therefore, we put (47) into (45) and we can get(44). Lemma 14 is proved. □

Lemma 15 *For E_X from (30), when (34) is satisfied, we have*

$$\|E_X\|_2 \leq \frac{1.15n^2\boldsymbol{u}\|X\|_g^2}{\sqrt{(\sigma_n(X))^2 + 0.9s}}. \tag{48}$$

Proof For Shifted CholeskyQR, R will not always be invertible due to errors in numerical computations. Therefore, we estimate this by examining each row. Similar to the approach in [8], we can express (29) as

$$q_i^\top = x_i^\top(R + E_{Ri})^{-1} = x_i^\top(I + R^{-1}E_{Ri})^{-1}R^{-1}. \tag{49}$$

When we define

$$(I + R^{-1}E_i)^{-1} = I + \theta_i \tag{50}$$

where

$$\theta_i := \sum_{j=1}^{\infty}(-R^{-1}E_{Ri})^j, \tag{51}$$

based on (29) and (30), we can have

$$E_{Xi}^\top = x_i^\top\theta_i \tag{52}$$

which is the i-th row of E_X. Based on (32), (34), (42) and (44), we can bound $\|R^{-1}E_{Ri}\|_2$ as

$$\begin{aligned}
\|R^{-1}E_{Ri}\|_2 &\le \|R^{-1}\|_2\|E_{Ri}\|_2 \\
&\le \frac{1.03n\sqrt{n}\mathbf{u}\|X\|_g}{\sqrt{(\sigma_n(X))^2+0.9s}} \\
&\le \frac{1.03n\sqrt{n}\mathbf{u}\|X\|_g}{\sqrt{0.9s}} \\
&\le \frac{1.03n\sqrt{n}\mathbf{u}\|X\|_g}{\sqrt{9.9(mn\mathbf{u}+n(n+1)\mathbf{u})\|X\|_g^2}} \\
&\le \frac{1.03n\sqrt{n}\mathbf{u}\|X\|_g}{\sqrt{9.9n(n+1)\mathbf{u}\|X\|_g^2}} \\
&\le 0.35\cdot\sqrt{n\mathbf{u}} \\
&\le 0.1. \qquad (53)
\end{aligned}$$

Putting (42), (44), (53) into (51) and we have

$$\begin{aligned}
\|\theta_i\|_2 &\le \sum_{j=1}^{\infty}(\|R^{-1}\|_2\|E_{Ri}\|_2)^j \\
&= \frac{\|R^{-1}\|_2\|E_{Ri}\|_2}{1-\|R^{-1}\|_2\|E_{Ri}\|_2} \\
&\le \frac{1}{0.9}\cdot\frac{1.03n\sqrt{n}\mathbf{u}\|X\|_g}{\sqrt{(\sigma_n(X))^2+0.9s}} \\
&\le \frac{1.15n\sqrt{n}\mathbf{u}\|X\|_g}{\sqrt{(\sigma_n(X))^2+0.9s}}. \qquad (54)
\end{aligned}$$

Summing all the items of (52) together and with (54), we have

$$\|E_X\|_2 \le \|E_X\|_F \le \|X\|_F\|\theta_i\|_2 \le \frac{1.15n^2\mathbf{u}\|X\|_g^2}{\sqrt{(\sigma_n(X))^2+0.9s}}.$$

with $\|X\|_F \le \sqrt{n}\|X\|_g$. Therefore, Lemma 15 is proved. □

Remark 2 The derivations of Lemmas 12-15 utilize the properties of the g-norm and we can get sharper upper bounds compared to those in [8]. This shows that Shifted CholeskyQR can be analyzed from the column of the input matrix X. The calculation of the Gram matrix and the existence of Cholesky factorization make it possible for us to improve the algorithm from this perspective.

4.4 Proof of Theorem 1

Proof Using the previous lemmas in Section 4.3, we begin to estimate the orthogonality and residual of our improved Shifted CholeskyQR. The proof of Theorem 1 is similar to that in [8]. We aim to demonstrate that comparable results hold, even with our enhanced bounds in the previous lemmas, based on the properties of the g-norm discussed in Section 4.3.

First, we consider the orthogonality. Based on (30), we can get

$$Q^\top Q = R^{-\top}(X+E_X)^\top(X+E_X)R^{-1}$$

$$
\begin{aligned}
&= R^{-\top}X^{\top}XR^{-1} + R^{-\top}X^{\top}E_XR^{-1} \\
&\quad + R^{-\top}E_X^{\top}XR^{-1} + R^{-\top}E_X^{\top}E_XR^{-1} \\
&= I - R^{-\top}(sI + E_A + E_B)R^{-1} + (XR^{-1})^{\top}E_XR^{-1} \\
&\quad + R^{-\top}E_X^{\top}(XR^{-1}) + R^{-\top}E_X^{\top}E_XR^{-1}. \qquad (55)
\end{aligned}
$$

With (55), we have

$$
\begin{aligned}
\|Q^{\top}Q - I\|_2 \le\ & \|R^{-1}\|_2^2(\|E_A\|_2 + \|E_B\|_2 + s) + 2\|R^{-1}\|_2\|XR^{-1}\|_2\|E_X\|_2 \\
& + \|R^{-1}\|_2^2\|E_X\|_2^2. \qquad (56)
\end{aligned}
$$

According to (34)-(36), we can get $\|E_A\|_2 + \|E_B\|_2 \le 1.1(mn\mathbf{u} + n(n+1)\mathbf{u})\|X\|_g^2 \le 0.1s$. With (42), we can get

$$
\begin{aligned}
\|R^{-1}\|_2^2(\|E_A\|_2 + \|E_B\|_2 + s) &\le \frac{1.1s}{(\sigma_n(X))^2 + 0.9s} \\
&\le \frac{11}{9} \\
&\le 1.23. \qquad (57)
\end{aligned}
$$

Based on (34), (42), (43) and (48), we can obtain

$$
\begin{aligned}
2\|R^{-1}\|_2\|XR^{-1}\|_2\|E_X\|_2 &\le 2 \cdot \frac{1}{\sqrt{(\sigma_n(X))^2 + 0.9s}} \cdot 1.5 \cdot \frac{1.15n^2\mathbf{u}\|X\|_g^2}{\sqrt{(\sigma_n(X))^2 + 0.9s}} \\
&\le \frac{3.45n^2\mathbf{u}\|X\|_g^2}{(\sigma_n(X))^2 + 0.9s} \\
&\le \frac{\frac{3.45}{11} \cdot s}{0.9s} \\
&le\ 0.35. \qquad (58)
\end{aligned}
$$

With (42) and (48), we have

$$
\begin{aligned}
\|R^{-1}\|_2^2\|E_X\|_2^2 &\le \frac{1}{(\sigma_n(X))^2 + 0.9s} \cdot \frac{(1.15n^2\mathbf{u}\|X\|_g^2)^2}{(\sigma_n(X))^2 + 0.9s} \\
&\le \frac{(\frac{3.45}{11} \cdot s)^2}{(0.9s)^2} \\
&\le 0.02. \qquad (59)
\end{aligned}
$$

We put (57)-(59) into (56) and we can get

$$
\begin{aligned}
\|Q^{\top}Q - I\|_2 &\le 1.23 + 0.35 + 0.02 \\
&\le 1.6.
\end{aligned}
$$

Therefore, (17) is proved.

From (17), it is easy to see that

$$\|Q\|_2 \le 1.62. \qquad (60)$$

For the residual, from (60), we can easily get

$$\|Q\|_F \le 1.62\sqrt{n}. \qquad (61)$$

For $\|QR - X\|_F$, based on (44) and (61), similar to the corresponding steps in [8], we will have (18). In all, Theorem 1 is proved □

Remark 3 In the proof of Theorem 1, we demonstrate that our improved s is sufficient to ensure numerical stability for Shifted CholeskyQR, with enhanced bounds established in the previous lemmas. This represents significant progress compared to that in [8]. The residual in (18) shows a tighter upper bound compared to that in [8]. More importantly, (18) can improve the condition for $\kappa_2(X)$ in the estimation of the singular values of Q in the next section.

4.5 Proof of Theorem 2

In this section, we give the proof for Theorem 2.

Proof We have already estimated $\|Q\|_2$. To estimate $\kappa_2(X)$, we need to estimate $\sigma_n(Q)$. The primary steps of analysis are similar to that in [8]. When (30) holds, according to Lemma 5, we can get

$$\sigma_n(Q) \geq \sigma_n(XR^{-1}) - \|E_X R^{-1}\|_2. \tag{62}$$

With (42) and (48), we can obtain

$$\|E_X R^{-1}\|_2 \leq \|E_X\|_2 \|R^{-1}\|_2 \leq \frac{1.67n^2\mathbf{u}\|X\|_g}{(\sigma_n(X))^2 + 0.9s}. \tag{63}$$

Using the similar method in [8], we have

$$\sigma_n(XR^{-1}) \geq \frac{\sigma_n(X)}{\sqrt{(\sigma_n(X))^2 + s}} \cdot 0.9. \tag{64}$$

When (33) holds, we put (63) and (64) into (62) and with $t = \frac{s}{\|X\|_2^2}$, we can get

$$\begin{aligned}
\sigma_n(Q) &\geq \frac{0.9\sigma_n(X)}{\sqrt{(\sigma_n(X))^2 + s}} - \frac{1.67n^2\mathbf{u}\|X\|_g}{\sqrt{(\sigma_n(X))^2 + 0.9s}} \\
&\geq \frac{0.9}{\sqrt{(\sigma_n(X))^2 + s}} \cdot (\sigma_n(X) - \frac{1.67}{0.9 \cdot \sqrt{0.9}} \cdot n^2\mathbf{u}\|X\|_g) \\
&\geq \frac{\sigma_n(X)}{2\sqrt{(\sigma_n(X))^2 + s}} \\
&= \frac{1}{2\sqrt{1 + t(\kappa_2(X))^2}}.
\end{aligned} \tag{65}$$

Based on (60) and (65), we have

$$\kappa_2(Q) \leq 3.24 \cdot \sqrt{1 + t(\kappa_2(X))^2}.$$

Therefore, we can get (19).

To improve the stability of orthogonality and residual, we add a CholeskyQR2 following the Shifted CholeskyQR, resulting in the Shifted CholeskyQR3. The numerical stability of this approach will be demonstrated in the next section similar to that in [8]. To obtain the sufficient condition of $\kappa_2(X)$ without encountering the numerical breakdown, based on (2) in [29], we let

$$\kappa_2(Q) \leq 3.24\sqrt{1 + t(\kappa_2(X))^2} \leq \frac{1}{8\sqrt{mn\mathbf{u} + n(n+1)\mathbf{u}}}. \tag{66}$$

When (34) is satisfied, along with (16) and $t = \frac{s}{\|X\|_2^2}$, we can have $11p^2(mn\mathbf{u}+n(n+1)\mathbf{u}) \le t \le \frac{1}{100}p^2$. When $s = 11(mn\mathbf{u}+n(n+1)\mathbf{u})\|X\|_g^2$, $t = 11p^2(mn\mathbf{u}+n(n+1)\mathbf{u})$. If $\kappa_2(X)$ is large enough, *e.g.*, $\kappa_2(X) \ge \mathbf{u}^{-\frac{1}{2}}$, we have $1 + t(\kappa_2(X))^2 \approx t(\kappa_2(X))^2$. Therefore, using (66), we can conclude that

$$\kappa_2(X) \le \frac{1}{25.92\sqrt{t}\cdot\sqrt{mn\mathbf{u}+n(n+1)\mathbf{u}}}. \tag{67}$$

We put $t = 11p^2(mn\mathbf{u}+n(n+1)\mathbf{u})$ into (67) and we can obtain (20). Therefore, Theorem 2 is proved. □

Remark 4 We have shown that our improved Shifted CholeskyQR, with a smaller s, has advantages in terms of the requirement for $\kappa_2(X)$ and its sufficient condition compared to the original method. A comprehensive comparison of the theoretical results is provided in Section 1, highlighting these advantages, which are further illustrated in Section 5.

4.6 Proof of Theorem 3

In this section, we prove Theorem 3 with some results in Theorem 1.

Proof We write CholeskyQR2 in Shifted CholeskyQR3 with error matrices below:

$$\begin{aligned}
&C - Q^\top Q = E_1,\\
&R_1^\top R_1 - C = E_2,\\
&Q_1R_1 - Q = E_3, &(68)\\
&R_1R - R_2 = E_4, &(69)\\
&C_1 - Q_1^\top Q_1 = E_5,\\
&R_3^\top R_3 - C_1 = E_6,\\
&Q_2R_3 - Q_1 = E_7, &(70)\\
&R_3R_2 - R4 = E_8. &(71)
\end{aligned}$$

Similar to the proof of Theorem 1, we consider the orthogonality first. For our improved Shifted CholeskyQR3, similar to that in [29], when Shifted CholeskyQR3 is applicable, we can get

$$\kappa_2(Q) \le \frac{1}{8\sqrt{mn\mathbf{u}+n(n+1)\mathbf{u}}}, \tag{72}$$

$$\kappa_2(Q_1) \le 1.1. \tag{73}$$

Therefore, we can obtain (22).

When considering the residual, based on (68)-(71), we have

$$\begin{aligned}
Q_2R_4 &= (Q_1+E_7)R_3^{-1}(R_3R_2 - E_8)\\
&= (Q_1+E_7)R_2 - (Q_1+E_7)R_3^{-1}E_8\\
&= Q_1R_2 + E_7R_2 - Q_2E_8\\
&= (Q+E_3)R_1^{-1}(R_1R - E_4) + E_7R_2 - Q_2E_8\\
&= (Q+E_3)R - (Q+E_3)R_1^{-1}E_4 + E_7R_2 - Q_2E_8\\
&= QR + E_3R - Q_1E_4 + E_7R_2 - Q_2E_8 \qquad (74)
\end{aligned}$$

Therefore, with (74), it is obvious that

$$\begin{aligned}\|Q_2R_4 - X\|_F \leq \|QR - X\|_F + \|E_3\|_F\|R\|_2 + \|Q_1\|_2\|E_4\|_F \\ + \|E_7\|_F\|R_2\|_2 + \|Q_2\|_2\|E_8\|_F. \end{aligned} \tag{75}$$

Similar to (29), we express (68) in each row as $q_{1i}^\top = q_i^\top(R_1 + E_{R1i})^{-1}$, where $q_{1i}^\top$ and $q_i^\top$ denote the i-th rows of Q_1 and Q. Following the methodologies outlined in [8, 29] and the concepts presented in our work, we have

$$\|R\|_2 \leq 1.006\|X\|_2, \tag{76}$$

$$\begin{aligned}\|E_{R1i}\|_2 &\leq 1.2n\sqrt{n}\mathbf{u} \cdot \|Q\|_2 \\ &\leq 2.079n\sqrt{n}\mathbf{u}, \end{aligned} \tag{77}$$

$$\|Q_1\|_2 \leq 1.039, \tag{78}$$

$$\begin{aligned}\|R_1\|_2 &\leq 1.1\|Q\|_2 \\ &\leq 1.906. \end{aligned} \tag{79}$$

We combine (76)-(79) with Lemma 6, Lemma 9, (47) and similar steps in [8], we can bound $\|E_3\|_F$, $\|E_4\|_F$ and $\|E_4\|_g$ in (68) and (69) as

$$\begin{aligned}\|E_3\|_F &\leq \|Q_1\|_F \cdot \|E_{R1i}\|_2 \\ &\leq 1.039 \cdot \sqrt{n} \cdot 2.079n\sqrt{n}\mathbf{u} \\ &\leq 2.16n^2\mathbf{u}, \end{aligned} \tag{80}$$

$$\begin{aligned}\|E_4\|_F &\leq \gamma_n(\|R_1\|_F \cdot \|R\|_F) \\ &\leq \gamma_n(\sqrt{n} \cdot \|R_1\|_2 \cdot \sqrt{n} \cdot \|R\|_g) \\ &\leq 1.1n^2\mathbf{u} \cdot 1.906 \cdot 1.006p\|X\|_2 \\ &\leq 2.11pn^2\mathbf{u}\|X\|_2, \end{aligned} \tag{81}$$

$$\begin{aligned}\|E_4\|_g &\leq \gamma_n(\|R_1\|_F \cdot \|R\|_g) \\ &\leq \gamma_n(\sqrt{n}\|R_1\|_2 \cdot \|R\|_g) \\ &\leq 1.1n\sqrt{n}\mathbf{u} \cdot 1.906 \cdot 1.006p\|X\|_2 \\ &\leq 2.11pn\sqrt{n}\mathbf{u}\|X\|_2. \end{aligned} \tag{82}$$

Moreover, based on Lemma 9, Lemma 10, (47), (76), (81) and (82), $\|R_2\|_2$ and $\|R_2\|_g$ in (69) can be bounded as

$$\begin{aligned}\|R_2\|_2 &\leq \|R_1\|_2\|R\|_2 + \|E_4\|_2 \\ &\leq 1.906 \cdot 1.006\|X\|_2 + 2.11pn^2\mathbf{u}\|X\|_2 \\ &\leq 1.95\|X\|_2, \end{aligned} \tag{83}$$

$$\begin{aligned}\|R_2\|_g &\leq \|R_1\|_2\|R\|_g + \|E_4\|_g \\ &\leq 1.906 \cdot 1.006p\|X\|_2 + 2.11pn\sqrt{n}\mathbf{u}\|X\|_2 \\ &\leq 1.95p\|X\|_2. \end{aligned} \tag{84}$$

Similar to (29), we write (70) in each row as $q_{2i}^\top = q_{1i}^\top(R_3 + E_{R3i})^{-1}$, where $q_{2i}^\top$ and $q_{1i}^\top$ represent the i-th rows of Q_2 and Q_1. Similar to (77)-(79) and with (22), (31) and (32), we can get

$$\|Q_2\|_2 \leq 1.1, \tag{85}$$

$$\|E_{R3i}\|_2 \leq 1.2n\sqrt{n}\|Q_1\|_2$$

$$\begin{aligned} &\leq 1.2n\sqrt{n}\mathbf{u} \cdot 1.039 \\ &\leq 1.246n\sqrt{n}\mathbf{u}, \end{aligned} \tag{86}$$

$$\begin{aligned} \|R_3\|_2 &\leq 1.1\|Q_1\|_2 \\ &\leq 1.143, \end{aligned} \tag{87}$$

With Lemma 6 and (84)-(87), we can bound $\|E_7\|_F$ and $\|E_8\|_F$ in (70) and (71) as

$$\begin{aligned} \|E_7\|_F &\leq \|Q_2\|_F \cdot \|E_{R3i}\|_2, \\ &\leq 1.1\sqrt{n} \cdot 1.246n\sqrt{n}\mathbf{u} \\ &\leq 1.38n^2\mathbf{u}, \end{aligned} \tag{88}$$

$$\begin{aligned} \|E_8\|_F &\leq \gamma_n(\|R_3\|_F \cdot \|R_2\|_F) \\ &\leq \gamma_n(\sqrt{n} \cdot \|R_3\|_2 \cdot \sqrt{n} \cdot \|R_2\|_g) \\ &\leq 1.1pn^2\mathbf{u} \cdot 1.143 \cdot 1.95p\|X\|_2 \\ &\leq 2.46pn^2\mathbf{u}\|X\|_2. \end{aligned} \tag{89}$$

Therefore, we put (18), (76), (78), (80), (81), (83), (85), (88) and (89) into (75) and we can get (23). In all, Theorem 3 is proved. □

Remark 5 Based on (23), we find that we obtain a sharper upper bound for the residual of the algorithm compared to that in [8], utilizing the properties of the g-norm. This represents a theoretical advancement in rounding error analysis. The steps leading to (84) highlight the effectiveness of Lemmas 9 and 11. Although the second inequality of (24) appears weaker than the first inequality of (24), it cannot be dismissed in estimating the g-norm of the error matrix in terms of its absolute value. This lays a solid foundation for (84) and (89), marking advancements in estimation methods for problems related to matrix multiplications.

Moreover, if X is not highly ill-conditioned, meaning that $\kappa_2(X)$ is small, our estimation of $\|E_A\|_2$ and $\|E_B\|_2$ can also be directly applied to CholeskyQR2. Therefore, the sufficient condition for $\kappa_2(X)$ can be expressed as

$$\kappa_2(X) \leq \frac{1}{8p\sqrt{mn\mathbf{u} + n(n+1)\mathbf{u}}}.$$

This condition is a better sufficient condition compared to (2) in [29].

5 Experimental Results

In this study, we conduct numerical experiments using MATLAB R2022a on a laptop. We compare our improved Shifted CholeskyQR3 with the original Shifted CholeskyQR3, focusing on three key properties: numerical stability(assessed through orthogonality $\|{Q_2}^\top Q_2 - I\|_F$ and residual $\|Q_2R_4 - X\|_F$ for Shifted CholeskyQR), the condition number of Q(denoted as $\kappa_2(Q)$) and the computational time(CPU time measured in seconds). Additionally, we present the p-value, defined as $p = \frac{\|X\|_g}{\|X\|_2}$, to illustrate the extent of improvement brought by our reduced s compared to the original method in [8]. As a comparison group, we also evaluate the properties of HouseholderQR, which is considered one of the most stable numerically stable algorithms and used in many softwares, to demonstrate the effectiveness and advantages of our improved Shifted CholeskyQR3. The specifications of our computer used for these experiments are provided in Table 3. We assess the performance of our method in multi-core CPU environments.

Table 3 The specifications of our computer

Item	Specification
System	Windows 11 family(10.0, Version 22000)
BIOS	GBCN17WW
CPU	Intel(R) Core(TM) i5-10500H CPU @ 2.50GHz -2.5 GHz
Number of CPUs / node	12
Memory size / node	8 GB
Direct Version	DirectX 12

5.1 Numerical Examples

In this part, we introduce the numerical examples, specifically the test matrix X utilized in this work. The primary test matrix $X \in \mathbb{R}^{m\times n}$ is similar to that used in [8, 29] and is constructed by SVD. It is straightforward to observe the influence of $\kappa_2(X)$, m and n while controlling the other two factors. Additionally, to test the applicability and the numerical stability of our improved Shifted CholeskyQR3, we present two examples widely used in engineering and other fields.

5.1.1 The Input *X* Based on SVD

We first construct the matrix X for the numerical experiments using Singular Value Decomposition (SVD), similar to the approach described in [8, 29]. We control $\kappa_2(X)$ through $\sigma_n(X)$. Specifically, we set

$$X = U\Sigma V^T.$$

Here, $U \in \mathbb{R}^{m\times m}$, $V \in \mathbb{R}^{n\times n}$ are random orthogonal matrices and

$$\Sigma = \mathrm{diag}(1, \sigma^{\frac{1}{n-1}}, \cdots, \sigma^{\frac{n-2}{n-1}}, \sigma) \in \mathbb{R}^{m\times n}.$$

Here, $0 < \sigma < 1$ is a constant. Therefore, we have $\sigma_1(X) = \|X\|_2 = 1$ and $\kappa_2(X) = \frac{1}{\sigma}$.

5.1.2 The Hilbert Matrix

The Hilbert matrix $X \in \mathbb{R}^{n\times n}$ is a well-known ill-conditioned matrix. It is widely used in many applications, including numerical approximation theory and solving linear systems, see [2, 4, 14] and the references therein. As n increases, $\kappa_2(X)$ also increases. The Hilbert matrix X is defined as below:

$$X_{ij} = \frac{1}{i+j-1}, i, j = 1, 2, \cdots, n.$$

We can use $X = hilb(n)$ in MATLAB to receive a Hilbert matrix $X \in \mathbb{R}^{n\times n}$.

5.1.3 The Arrowhead Matrix

The arrowhead matrix $X \in \mathbb{R}^{n\times n}$ plays an important role in graph theory, control theory and some eigenvalue problems, see [3, 18, 22, 23, 27] and the references therein. Its primary

characteristic is that all the elements are zero except for those in the first column, the first row and the diagonal. In this work, we define an arrowhead matrix as follows:

$$\begin{aligned} X_{1j} &= 30,\ j = 1, 2, \cdots, n, \\ X_{ii} &= 10,\ i = 2, 3, \cdots, n-1, \\ X_{ii} &= 10^{-16},\ i = n, \\ X_{ij} &= 0, \text{others}. \end{aligned}$$

5.2 Numerical Stability of the Algorithms

In this section, we test the numerical stability of the algorithms. To assess this, we conduct experiments considering three factors: $\kappa_2(X)$, m and n to demonstrate the properties of Shifted CholeskyQR3. For clarity, we refer to our improved Shifted CholeskyQR3 as 'Improved', while the original Shifted CholeskyQR3 is referred to as 'Original'.

To assess the potential influence of $\kappa_2(X)$, we obtain X using SVD first. We fix $m = 2048$ and $n = 64$, varying σ to evaluate the effectiveness of our algorithm with different $\kappa_2(X)$. The numerical results are listed in Tables 4 and 5. Numerical experiments show that our improved Shifted CholeskyQR3 exhibit better orthogonality and residual compared to HouseholderQR, demonstrating strong numerical stability. The numerical stability of our improved algorithm is comparable to that of the original Shifted CholeskyQR3. A key advantage of our improved Shifted CholeskyQR3 over the original one is that our improved algorithm can handle more ill-conditioned X with $\kappa_2(X) \geq 10^{12}$. The conservative choice of s in the original Shifted CholeskyQR3 limits its computational range, as reflected in the comparison of $\kappa_2(X)$ between (10) and (20). In our practical example of the Hilbert matrix, we take $n = 12$ and $\kappa_2(X) = 1.62e+16$. In the example of the arrowhead matrix, we take $n = 64$ and $\kappa_2(X) = 3.40e+18$. The numerical results are shown in Tables 6 and 7. They also demonstrate that our improved Shifted CholeskyQR3 has better applicability and is able to handle more ill-conditioned matrices effectively than the original one.

To examine the influence of m and n, we construct X based on SVD while maintaining $\kappa_2(X) = 10^{12}$. When m is varying, we keep $n = 64$. When n is varying, we keep $m = 2048$. The numerical results are presented in Tables 8- 11. Our findings indicate that the increasing n leads to greater rounding errors in orthogonality and residual, while m does not impact these aspects significantly. Our improved Shifted CholeskyQR3 maintains a level of the numerical stability comparable to that of the original Shifted CholeskyQR3 and is more accurate compared to HouseholderQR across various values of m and n. This set of experiments shows that our improved Shifted CholeskyQR3 is numerical stable across different problem sizes.

Overall, our examples demonstrate that our improved Shifted CholeskyQR3 is more applicable for ill-conditioned matrices without sacrificing numerical stability, performing at a level comparable to the original Shifted CholeskyQR3. In many cases, it even exhibits better accuracy compared to the traditional HouseholderQR.

5.3 $\kappa_2(Q)$ Under Different Conditions

In this group of experiments, we evaluate the impact of $\kappa_2(X)$, m and n on $\kappa_2(Q)$ using different values of s for Shifted CholeskyQR3, which is crucial for assessing the applicability

Table 4 Orthogonality for the algorithms with $\kappa_2(X)$ varying when $m = 2048$ and $n = 64$

$\kappa_2(X)$	$1.00e+8$	$1.00e+10$	$1.00e+12$	$1.00e+14$	$1.00e+16$
Improved	$2.07e-15$	$2.04e-15$	$2.03e-15$	$2.04e-15$	-
Original	$2.14e-15$	$2.21e-15$	$1.90e-15$	-	-
HouseholderQR	$2.77e-15$	$2.46e-15$	$2.48e-15$	$2.75e-14$	$2.67e-15$

Table 5 Residual for the algorithms with $\kappa_2(X)$ varying when $m = 2048$ and $n = 64$

$\kappa_2(X)$	$1.00e+8$	$1.00e+10$	$1.00e+12$	$1.00e+14$	$1.00e+16$
Improved	$6.35e-16$	$6.01e-16$	$5.80e-16$	$5.64e-16$	-
Original	$6.67e-16$	$6.20e-16$	$6.22e-16$	-	-
HouseholderQR	$1.26e-15$	$1.38e-15$	$1.27e-15$	$1.27e-15$	$9.61e-16$

Table 6 Numerical results for the Hilbert matrix with $n = 12$

Algorithm	Improved	Original
Orthogonality	$3.59e-15$	–
Residual	$2.14e-16$	–

Table 7 Numerical results for the arrowhead matrix with $n = 64$

Algorithm	Improved	Original
Orthogonality	$1.24e-14$	–
Residual	$1.40e-14$	–

of the algorithms. We compare our improved Shifted CholeskyQR3 with the original Shifted CholeskyQR3.

In this group of experiments, we use X based on SVD. Initially, we fix $m = 2048$ and $n = 64$, varying $\kappa_2(X)$ to see the corresponding $\kappa_2(Q)$ with different values of s in Shifted CholeskyQR3. The results are listed in Table 12. From Table 12, we can see that

Table 8 Orthogonality for all the algorithms with m varying when $\kappa_2(X) = 10^{12}$ and $n = 64$

m	128	256	512	1024	2048
Improved	$3.62e-15$	$4.07e-15$	$3.11e-15$	$2.12e-15$	$2.03e-15$
Original	$3.31e-15$	$3.93e-15$	$2.89e-15$	$2.36e-15$	$1.90e-15$
HouseholderQR	$6.54e-15$	$6.35e-15$	$3.56e-15$	$2.80e-15$	$2.48e-15$

Table 9 Residual for all the algorithms with m varying when $\kappa_2(X) = 10^{12}$ and $n = 64$

m	128	256	512	1024	2048
Improved	$6.04e-16$	$5.92e-16$	$6.08e-16$	$6.06e-16$	$5.80e-16$
Original	$6.09e-16$	$5.91e-16$	$5.95e-16$	$5.86e-16$	$6.22e-16$
HouseholderQR	$7.31e-16$	$9.45e-16$	$7.55e-16$	$7.48e-16$	$1.27e-15$

Table 10 Orthogonality for all the algorithms with n varying when $\kappa_2(X) = 10^{12}$ and $m = 2048$

n	64	128	256	512	1024
Improved	$2.03e-15$	$3.25e-15$	$5.29e-15$	$9.53e-15$	$1.69e-14$
Original	$1.90e-15$	$3.33e-15$	$5.19e-15$	$1.66e-15$	$1.77e-14$
HouseholderQR	$2.48e-15$	$4.66e-15$	$9.39e-15$	$2.07e-14$	$5.02e-14$

Table 11 Residual for all the algorithms with n varying when $\kappa_2(X) = 10^{12}$ and $m = 2048$

n	64	128	256	512	1024
Improved	$5.80e-16$	$1.07e-15$	$2.01e-15$	$3.06e-15$	$4.32e-15$
Original	$6.22e-16$	$1.08e-15$	$2.04e-15$	$3.08e-15$	$4.33e-15$
HouseholderQR	$1.27e-15$	$1.76e-15$	$2.55e-15$	$3.62e-15$	$5.00e-15$

Table 12 $\kappa_2(Q)$ with $\kappa_2(X)$ varying with different s when $m = 2048$ and $n = 64$

$\kappa_2(X)$	$1.00e+8$	$1.00e+10$	$1.00e+12$	$1.00e+14$	$1.00e+16$
Improved	358.60	$3.37e+04$	$3.18e+06$	$3.01e+08$	–
Original	$1.29e+03$	$1.29e+05$	$1.29e+07$	–	–

Table 13 $\kappa_2(Q)$ with m varying using different s when $\kappa_2(X) = 10^{12}$ and $n = 64$

m	128	256	512	1024	2048
Improved	$9.62e+05$	$1.24e+06$	$1.66e+06$	$2.29e+06$	$3.18e+06$
Original	$3.88e+06$	$5.01e+06$	$6.72e+06$	$9.23e+06$	$1.29e+07$

$\kappa_2(X)$ exhibits a nearly direct proportionality to $\kappa_2(Q)$. With an improved smaller s, our improved Shifted CholeskyQR3 achieves a smaller $\kappa_2(X)$ compared to the original Shifted CholeskyQR3, which is consistent with (9) and (19).

Next, we test the influence of m and n on $\kappa_2(X)$. When varying m, we fix $\kappa_2(X) = 10^{12}$ and $n = 64$. For different n, we set $\kappa_2(X) = 10^{12}$ and $m = 2048$. The numerical results are listed in Tables 13 and 14. These results indicate that when dealing with a tall-skinny matrix $X \in \mathbb{R}^{m\times n}$ with $m > n$, increasing both m and n leads to a larger $\kappa_2(Q)$ while keeping $\kappa_2(X)$ fixed. This arises from the structures of both our improved s and the original s. Across Tables 12- 14, we consistently observe that our method achieves a smaller $\kappa_2(Q)$ compared to the original Shifted CholeskyQR3, demonstrating the effectiveness of the improved s.

In conclusion, our reduced s in this work results in a smaller $\kappa_2(Q)$, enhancing the applicability of our improved Shifted CholesyQR3 compared to the original algorithm. This represents a significant advancement in our research.

5.4 CPU Times of the Algorithms

In addition to considering numerical stability and $\kappa_2(Q)$, we also need to take into account the CPU time required by these algorithms to demonstrate the efficiency of our improved algorithm. We test the corresponding CPU time with respect to the two variables, m and n.

Table 14 $\kappa_2(Q)$ with n varying using different s when $\kappa_2(X) = 10^{12}$ and $m = 2048$

n	64	128	256	512	1024
Improved	$3.18e+06$	$4.24e+06$	$5.76e+06$	$8.11e+06$	$1.11e+07$
Original	$1.29e+07$	$1.84e+07$	$2.68e+07$	$4.00e+07$	$6.20e+07$

Table 15 CPU time with m varying (in second) when $\kappa_2(X) = 10^{12}$ and $n = 64$

m	128	256	512	1024	2048
Improved	$6.90e-04$	$8.65e-04$	$1.70e-03$	$3.80e-03$	$4.70e-03$
Original	$2.10e-03$	$9.55e-04$	$1.50e-03$	$4.40e-03$	$6.20e-03$
HouseholderQR	$1.21e-02$	$3.45e-02$	$3.38e-01$	$2.00e+00$	$1.24e+01$

Table 16 CPU time with n varying (in second) when $\kappa_2(X) = 10^{12}$ and $m = 2048$

n	64	128	256	512	1024
Improved	$4.70e-03$	$1.25e-02$	$4.66e-02$	$9.80e-02$	$3.52e-01$
Original	$6.20e-03$	$1.46e-02$	$4.59e-02$	$9.02e-02$	$4.45e-01$
HouseholderQR	$1.12e+01$	$2.59e+01$	$5.66e+01$	$1.16e+02$	$3.11e+02$

Similar to the previous section, we use X based on SVD. For varying values of m, we set $n = 64$ and $\kappa_2(X) = 10^{12}$. When n is varying, we fix $m = 2048$ and $\kappa_2(X) = 10^{12}$. We observe the variation in CPU time for our improved Shifted CholeskyQR3, the original Shifted CholeskyQR3 algorithm and HouseholderQR. The CPU times for these algorithms are listed in Tables 15 and 16. Numerical experiments show that both our improved Shifted CholeskyQR3 and the original Shifted CholeskyQR3 are significantly more efficient compared to HouseholderQR, highlighting a primary drawback of the widely-used HouseholderQR. Our improved Shifted CholeskyQR3 exhibits comparable speed to the original Shifted CholeskyQR3 with $normest$. Additionally, n has a greater influence on CPU time compared to m. However, as both m and n increase, our improved Shifted CholeskyQR3 maintains a level of efficiency similar to that of the original Shifted CholeskyQR3. Therefore, we conclude that our improved Shifted CholeskyQR3 is an efficient algorithm with good accuracy for problems with moderate sizes.

5.5 *p*-Values

Here, we aim to show the p-values in this work by using some examples. Based on Tables 1 and (16), we can find that the proportion of our improved s to the original s is p^2. Therefore, the p-value reflects how much the shifted item s is reduced according to our definition of the g-norm. In the future, we will investigate how to estimate p under different cases.

In this part, we test the p-value with varying values of m and n using X based on SVD. With m varying, we fix $n = 64$ and $\kappa_2(X) = 10^{12}$. For different values of n, we fix $m = 2048$ and $\kappa_2(X) = 10^{12}$. The numerical experiments are listed in Tables 17 and 18. The numerical results indicate that p is relatively small compared to 1. Notably, n significantly influences p more than m. With n increasing, p decreases markedly, which aligns with the theoretical

Table 17 p with m varying when $\kappa_2(X) = 10^{12}$ and $n = 64$

m	128	256	512	1024	2048
p	0.2824	0.2762	0.2386	0.2453	0.2498

Table 18 p with n varying when $\kappa_2(X) = 10^{12}$ and $m = 2048$

n	64	128	256	512	1024
p	0.2498	0.2396	0.2127	0.2024	0.1726

lower bound of the p-value. This observation suggests that our improved s is likely more effective for relatively large matrices.

6 Discussions and Conclusions

This study focuses on determining an optimal choice for the shifted item s based on the properties of the input matrix X for Shifted CholeskyQR3. We introduce a new g-norm for X based on column properties and derive a new smaller s using $\|X\|_g$. We demonstrate that this smaller s provides a better upper bound for $\kappa_2(X)$, thereby enhancing the applicability of Shifted CholeskyQR3 while maintaining its numerical stability in terms of both orthogonality and residuals. In terms of computational efficiency, our improved Shifted CholeskyQR3 outperforms the commonly used HouseholderQR method and exhibits a similar CPU time to the original Shifted CholeskyQR3 for moderately sized matrices, demonstrating that our algorithm is effective in terms of speed as a three-step deterministic method.

There are still several issues that need to be addressed in the future. Specifically, the g-norm of the matrix warrants further exploration. Developing efficient methods to quickly estimate $\|X\|_g$ for input matrices X remains an open topic for future research, particularly for large matrices. In this work, we calculate $\|X\|_g$ by comparing the 2-norms of the columns of X. However, as the size of the matrix increases, the CPU time and computational cost of this method increase significantly. Therefore, new techniques for estimating $\|X\|_g$ more cost-effectively need to be developed. Moreover, the process of calculating $\|X\|_g$ indicates that parallel computing can be employed to obtain the g-norm more efficiently. We are currently developing an estimator that leverages parallel computing to calculate the g-norm for the improved Shifted CholeskyQR. In this study, we leverage the connections between the g-norm and other norms to conduct rounding error analysis. Given that the g-norm can be applied to various problems, such as HouseholderQR and Nyström approximation, we aim to explore its relationship with the singular values of the matrix and other factors, such as the condition number. We are also focusing on more properties related to the g-norm.

Recent years have seen significant advancements in CholeskyQR methodologies, as evidenced by studies such as [28, 30, 31]. While deterministic methods like Shifted CholeskyQR3 offer good accuracy, they are often relatively slow. The choice of the parameter s continues to influence the applicability of the algorithm, even with the improvements proposed in this work. In addition, randomized methods for the CholeskyQR algorithm have been introduced [21, 33] in recent years. However, all Cholesky-type algorithms face with issues of the sufficient condition for the condition number $\kappa_2(X)$ applicable to the input matrix X, which limits their practical use in industry. To address this, we are exploring new preconditioning steps designed to balance speed, accuracy, and applicability, thereby enhancing the performance of CholeskyQR-type algorithms.

Acknowledgements This work is supported by the CAS AMSS-PolyU Joint Laboratory of Applied Mathematics. The contributions of H. Guan and Z. Qiao are funded by the Hong Kong Research Grants Council through the RFS grant RFS2021-5S03 and GRF grant 15302122, as well as by the Hong Kong Polytechnic University under grant 4-ZZLS. We would like to express our gratitude to Mr. Yuan Liang from Beijing Normal University, Zhuhai, for his valuable suggestions regarding the coding aspects of this research. Additionally, we appreciate the insightful discussions with Mr. Renfeng Peng from the Chinese Academy of Sciences, Professor Valeria Simoncini, and Dr. Davide Palitta from University of Bologna, Italy, regarding the properties of the g-norm and potential future directions in this area. Our thanks also go to Dr. Nan Zheng from the Hong Kong Polytechnic University for her assistance in revising this manuscript. Finally, we are grateful to the two anonymous referees for their constructive feedback, which has contributed to enhancing this work.

Funding Open access funding provided by The Hong Kong Polytechnic University

Data Availability The authors declare that all data supporting the findings of this study are available within this article.

Declarations

Conflicts of Interest The authors declare that they have no conflict of interest.

References

1. Ballard, G., Demmel, J., Holtz, O., Schwartz, O.: Minimizing communication in numerical linear algebra. SIAM Journal on Matrix Analysis and Applications **32**(3), 866–901 (2011)
2. Beckermann, Bernhard: The condition number of real Vandermonde, Krylov and positive definite Hankel matrices. Numerische Mathematik **85**, 553–577 (2000)
3. Borobia, Alberto: Constructing matrices with prescribed main-diagonal submatrix and characteristic polynomial. Linear Algebra and its Applications **418**, 886–890 (2006)
4. Choi, Man-Duen.: Tricks or Treats with the Hilbert Matrix. The American Mathematical Monthly **90**(5), 301–312 (1983)
5. Constantine P.G., Gleich,D.F.: Tall and skinny QR factorizations in MapReduce architectures. In Proceedings of the second international workshop on MapReduce and its applications, pages 43–50, 2011
6. Jeannerod, C.P., Rump, S.M.: Improved error bounds for inner products in floating-point arithmetic. SIAM Journal on Matrix Analysis and Applications **34**, 338–344 (2013)
7. Duersch, J.A., Shao, M., Yang, C., Gu, M.: A robust and efficient implementation of LOBPCG. SIAM Journal on Scientific Computing **40**(5), C655–C676 (2018)
8. Fukaya, Takeshi, Kannan, Ramaseshan, Nakatsukasa, Yuji, Yamamoto, Yusaku, Yanagisawa, Yuka: Shifted Cholesky QR for computing the QR factorization of ill-conditioned matrices. SIAM Journal on Scientific Computing **42**(1), A477–A503 (2020)
9. Fukaya,T., Nakatsukasa,Y., Yanagisawa,Y., Yamamoto,Y.: CholeskyQR2: a simple and communication-avoiding algorithm for computing a tall-skinny QR factorization on a large-scale parallel system. In 2014 5th workshop on latest advances in scalable algorithms for large-scale systems, pages 31–38. IEEE, 2014
10. Gene, H.: Golub and Charles F, Van Loan. Matrix Computations. 4th edn. The Johns Hopkins University Press, Baltimore (2013)
11. Stewart,G.W., Sun,J.: Matrix perturbation theory. Academic Press, San Diego, CA, USA, sixth ed. edition, 1990
12. Halko, N., Martinsson, P.-G., Tropp, J.A.: Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. SIAM review **53**(2), 217–288 (2011)

13. Higham,N. J.: Accuracy and Stability of Numerical Algorithms. SIAM, Philadelphia, PA, USA, second ed. edition, 2002
14. Hilbert,D.: Ein Beitrag zur Theorie des Legendre'schen Polynoms. Acta Mathematica, 18(none):155 – 159, 1900
15. Hoemmen, Mark: Communication-avoiding Krylov subspace methods. University of California, Berkeley (2010)
16. Demmel,J.: On floating point errors in Cholesky. Tech.Report 14, LAPACK working Note, 1989
17. Demmel,J., Grigori,L., Hoemmen,M., Langou,J.: Communication-optimal parallel and sequential QR and LU factorizations. in Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis, pages 36:1–36:12, 2009
18. Li,Z., Wang,Y., Li,S.: The inverse eigenvalue problem for generalized Jacobi matrices with functional relationship. 2015 12th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), pages 473–475, 2015
19. Martinsson, Per-Gunnar., Tropp, Joel A.: Randomized numerical linear algebra: Foundations and algorithms. Acta Numerica **29**, 403–572 (2020)
20. Rozloznik,M., Tuma,M., Smoktunowicz,A., Kopal,J.: Numerical stability of orthogonalization methods with a non-standard inner product. BIT, pages 1–24, 2012
21. Balabanov,O.: Randomized CholeskyQR factorizations. arxiv preprint arXiv:2210.09953, 2022
22. O'Leary, D.P., Stewart, G.W.: Computing the eigenvalues and eigenvectors of symmetric arrowhead matrices. Journal of Computational Physics **90**(2), 497–505 (1990)
23. Peng, Juan, Xiyan, Hu., Zhang, Lei: Two inverse eigenvalue problems for a special kind of matrices. Linear Algebra and its Applications **416**, 336–347 (2006)
24. Schreiber, Robert, Van Loan, Charles: A storage-efficient WY representation for products of Householder transformations. SIAM Journal on Scientific and Statistical Computing **10**(1), 53–57 (1989)
25. Rump, S.M., Jeannerod, C.P.: Improved backward error bounds for LU and Cholesky factorization. SIAM Journal on Matrix Analysis and Applications **35**, 684–698 (2014)
26. Rump, S.M., Ogita, T.: Super-fast vallidated solution of linear systems. J. Comput. Appl. Math. **199**, 199–206 (2007)
27. Stor,N.J., Slapnicar,I., Barlow ,J.L.: Accurate eigenvalue decomposition of real symmetric arrowhead matrices and applications. Linear Algebra and its Applications, 464:62–89, 2015. Special issue on eigenvalue problems
28. Terao, Takeshi, Ozaki, Katsuhisa, Ogita, Takeshi: LU-Cholesky QR algorithms for thin QR decomposition. Parallel Computing **92**, 102571 (2020)
29. Yamamoto, Y., Nakatsukasa, Y., Yanagisawa,Y., Fukaya, T.: Roundoff error analysis of the CholeskyQR2 algorithm. Electron. Trans. Numer. Anal, 44(01), 2015
30. Yamamoto, Yusaku, Nakatsukasa, Yuji, Yanagisawa, Yuka, Fukaya, Takeshi: Roundoff error analysis of the CholeskyQR2 algorithm in an oblique inner product. JSIAM Letters **8**, 5–8 (2016)
31. Yamasaki, Ichitaro, Tomov, Stanimire, Dongarra, Jack: Mixed-precision Cholesky QR factorization and its case studies on Multicore CPU with Multiple GPUs. SIAM Journal on Scientific Computing **37**, C307–C330 (2015)
32. Yanagisawa, Yuka, Ogita, Takeshi, Oishi, Shin'ichi: A modified algorithm for accurate inverse Cholesky factorization. Nonlinear Theory and Its Applications, IEICE **5**, 35–46 (2014)
33. Fan,Y., Guo,Y., Lin,T.: A Novel Randomized XR-Based Preconditioned CholeskyQR Algorithm. arxiv preprint arXiv:2111.11148, 2021