CHOLESKYQR-TYPE ALGORITHMS:DEVELOPMENT AND ANALYSIS

HAORAN GUAN

PhD

The Hong Kong Polytechnic University

2025

The Hong Kong Polytechnic University

Department of Applied Mathematics

CholeskyQR-type Algorithms:Development and Analysis

Haoran Guan

A thesis submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy

June 2025

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____ (Signed)

Haoran Guan
_____ (Name of student)

Abstract

This thesis focuses on the development of CholeskyQR-type algorithms, which are very popular in recent years due to their efficiency and accuracy. Compared to the traditional algorithms for QR factorization, such as HouseholderQR and MGS, CholeskyQR-type algorithms have special advantages and have raised much attention from both academia and industry. In this thesis, We present some progress we have made in CholeskyQR-type algorithms in the past several years.

Though with good efficiency and accuracy, CholeskyQR is seldom used alone due to its lack of orthogonality. In order to receive numerical stability in orthogonality, CholeskyQR2 has been developed by repeating CholeskyQR twice. In recent years, researchers has proposed Shifted CholeskyQR3 to deal with QR factorization of ill-conditioned matrices, with a shifted item s in the step of Cholesky factorization to avoid numerical breakdown in ill-conditioned cases. Moreover, some other CholeskyQR-type algorithms have occurred, such as LU-CholeskyQR2 and some randomized algorithms. The development of CholeskyQR-type algorithms aims for improving the applicability of the algorithms. In this thesis, we show our improvements on the applicability of CholeskyQR-type algorithms, especially for Shifted CholeskyQR3. Some cases based on real-world problems are also considered.

Shifted CholeskyQR3 avoids the problem of encountering numerical breakdown in ill-conditioned cases which belongs to CholeskyQR2. With the structure of CholeskyQR2 after Shifted CholeskyQR, Shifted CholeskyQR3 can keep numerical stability and replace CholeskyQR2. However, the original shifted item $s = 11(mn\mathbf{u} + (n+1)n\mathbf{u})\|X\|_2^2$ for the input matrix $X \in \mathbb{R}^{m\times n}$ is relatively conservative due to overestimation in rounding error analysis. We introduce a new matrix norm $\|X\|_c$ and propose an improved shifted item $s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$ for Shifted CholeskyQR3. Our theoretical analysis and numerical experiments demonstrate that our new s can enhance the applicability of Shifted CholeskyQR3, while maintaining numerical stability and efficiency.

In fact, in many real-world applications, the input matrix $X \in \mathbb{R}^{m\times n}$ is often sparse, especially when m and n are large. Due to the structure of the algorithm, the sparsity of the input matrix will influence rounding error analysis of CholeskyQR and exhibit different properties compared to those of dense matrices. For sparse matrices, we build a new model and divide them into two types, T_1 matrices with the dense columns and T_2 matrices whose columns are all sparse. Therefore, an alternative choice of the shifted item s is proposed for Shifted CholeskyQR3 based on the structure and the key element of the input X. We prove that such an alternative s are optimal compared to the original s we propose in the previous part with certain element-norm conditions(ENCs). It can improve the applicability of Shifted CholeskyQR3 for T_1 matrices and maintain numerical stability of the algorithm in this way. Numerical experiments demonstrate our findings and show that shifted CholeskyQR3 with the

alternative s can also deal with more ill-conditioned cases for T_2 matrices because of the potential sparsity of the orthogonal factor after Shifted CholeskyQR. The algorithm with such an s is also as efficient as the case with the original s. $\|\cdot\|_g$, a definition connected to $\|\cdot\|_c$, is utilized in the theoretical analysis.

In recent years, probabilistic rounding error analysis has become a hot topic in numerical linear algebra. We can receive tighter error bounds compared to the deterministic ones. Based on the theoretical analysis of CholeskyQR-type algorithms, probabilistic error analysis can improve the sufficient condition of $\kappa_2(X)$ for $X \in \mathbb{R}^{m\times n}$ and bring more accurate error analysis. Therefore, we do probabilistic error analysis of CholeskyQR-type algorithms. We receive tighter upper bounds of both orthogonality and residual for CholeskyQR-type algorithms, together with looser sufficient conditions of $\kappa_2(X)$ with the corresponding probabilities. Additionally, a probabilistic s with $\|X\|_c$ is proposed for Shifted CholeskyQR3. Numerical experiments show that such a probabilistic s can improve the applicability of the algorithm further. Shifted CholeskyQR3 with such a probabilistic s is also numerical stable and robust enough after numerous experiments.

Generally speaking, we propose and utilize new tools for more accurate rounding error analysis of CholeskyQR-type algorithms theoretically, which also helps to improve the properties of the algorithm. Our improvements on the applicability of the algorithm are effective according to numerical experiments, which correspond to the new theoretical results in this work.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my supervisor, Dr. Zhonghua Qiao from the Hong Kong Polytechnic University, and Dr. Yuwei Fan from Huawei for their invaluable guidance, unwavering support, and academic insights throughout my four-year PhD journey. Their expertise in computational mathematics has significantly enhanced my research skills and academic development. I feel fortunate to have met Dr. Qiao and Dr. Fan during my studies at the Hong Kong Polytechnic University. They not only taught me how to conduct scientific research but also guided me on how to become a good scientist in my future career. I am deeply thankful for their assistance and advice during the challenges I faced in my research over the past several years. Their support and encouragement were crucial in helping me overcome obstacles and successfully complete my thesis.

Moreover, I am very grateful for many colleagues and professors for their academic advice and help in my research. Dr. Ting-kei Pong, Dr. Zhian Wang, Dr. Xingqiu Zhao and Dr. Buyang Li taught me several courses in mathematics. Dr. Xiaojun Chen told us how to present our work appropriately in a brief talk. Dr. Yanping Lin served as the BoE Chair of mine. Dr. Tiexiang Li from Southeast University, China gave me the idea of CholeskyQR-type algorithms in the case of sparse matrices. Dr. Qinmeng Zou from Beijing University of Posts and Telecommunications, China gave me many suggestions regarding probabilistic error analysis in numerical linear algebra. Dr. Valeria Simoncini and Dr. Davide Palitta from University of Bologna, Italy had some discussions with me regarding my topic and worked with me in solving Lyapunov equations with RPCholesky. I am also very thankful for their supervision and support during my RSAP study in Italy in 2024. Dr. Michael Kwok-Po Ng from Hong Kong Baptist University attended my oral examination and discussed with me regarding my research topic. Dr. Yuji Nakatsukasa from Oxford University, England provided many helpful suggestions regarding CholeskyQR and presentations. Mr. Renfeng Peng from Chinese Academy of Science, China discussed with me about sparse matrices. Mr. Yuan Liang from Beijing Normal University, Zhuhai, China gave me useful examples in the numerical experiments of my articles. Moreover, I thank Dr. Defeng Sun, Miss. Natalie Cheung Ting-ting, Ms. Cynthia Hau, Miss. Teresa Ko Shuk-wai, Ms. Elki Wong Ya-king and all other staffs of AMA department for their appropriate support.

I feel fortunate to have met many wonderful PhD students and research fellows in the AMA department. I would like to express my deep appreciation to my group members at PolyU: Dr. Xiao Li, Dr. Qian Zhang, Dr. Chaoyu Liu, Dr. Qian Yin, Dr. Limin Ma, Dr. Yonghui Bo, Dr. Dianming Hou, Dr. Shuyu Sun, Dr. Jianbo Cui, Dr. Yuze Zhang, Dr. Nan Zheng, Dr. Yaping Chen, Dr.

Xuguang Yang, Dr. Caixia Nan, Dr. Jingyun Lv, Dr. Yifan Wei, Dr. Yunzhuo Guo, Dr. Wangbo Luo, Mr. Qizhe Fan, Dr. Gaohang Chen, Miss Jiayi Duan, Mr. Yunzhuo Guo, Dr. Huiting Yang, Mr. Yongchen Fan, Mr. Jingwen Dai, Miss Xin Wang, Miss Yuyan Chang, Mr. Shengtong Liang, Miss Li Xia, Miss Yue Qian, and Mr. Guangshen Liu for their continuous support and camaraderie over the past four years. I would also like to thank Dr. Jun Li and Dr. Yipei Chen from Huawei for their valuable help and advice. Additionally, I am grateful to my old friends—Mr. Hongzhan Zhao, Ms. Yingchao Xu, Miss Mingrui Li, Mr. Zheyuan Zhao, Mr. Chaoyi Cai, and Miss Yaping Dong—who provided me with support and encouragement throughout the years. I truly appreciate their kindness and friendship.

Finally, I would like to express my deep gratitude to my parents and family for their unwavering love and patience. I am here to honor my grandma passing away in recent days, who has taken care of me and always been proud of me for many years.

Contents

List of Figures

List of Tables

CHAPTER 1.

INTRODUCTION

QR factorization is one of the most important components of numerical linear algebra and is widely used in both academia and industry. There are many illustrations regarding such an issue in the existing works, see [18, 30, 36, 47, 53] for more details. Among all the various algorithms for QR factorization, CholeskyQR has gained popularity in recent years due to its ability to balance efficiency and accuracy. Different from other algorithms for QR factorization, CholeskyQR exclusively utilizes BLAS3 operations and requires only simple reductions in parallel environments, which is a significant advantage compared to other algorithms such as HouseholderQR, CGS (MGS), and TSQR [5, 12, 16, 25, 32, 50, 55].

This chapter is an introduction of this thesis. Notations used in this thesis are introduced in Section 1.1. We show some existing CholeskyQR-type algorithms and their theoretical properties in Section 1.2. Then, some comparisons and considerations are presented in Section 1.3, which illustrates our purposes to improve the properties of CholeskyQR-type algorithms. We show our contributions in Section 1.4 and the primary tools for analysis are in Section 1.5. In the end of this chapter, we put an outline of this thesis in Section 1.6.

1.1 Notations of this thesis

In this thesis, all vectors and matrices are real. The notations $\|\cdot\|_2$ and $\|\cdot\|_F$ denote the 2-norm and the Frobenius norm of the matrix, respectively. The condition number $\kappa_2(\cdot)$ utilized in this thesis refers to the 2-norm condition number and is defined as

$$\kappa_2(X) = \frac{\|X\|_2}{\sigma_{min}(X)},$$

where $\|X\|_2$ equals to the largest singular value of X. $\sigma_{min}(X)$ denotes the smallest singular value of matrix X. $\mathbf{u}$ is the machine precision and $\mathbf{u} = 2^{-53}$. For the input matrix $X, |X|$ is the matrix whose elements are all the absolute values of the elements of X.

1.2 Existing CholeskyQR-type algorithms

CholeskyQR is a novel algorithm which is primarily designed for the QR factorizaton of tall-skinny matrices that are prevalent in the real problems of engineering. It is primarily designed for full rank matrices. For $X \in \mathbb{R}^{m\times n}$ with $m \geq n$ and rank(X)=n, the Gram matrix B is computed first through

$X^\top X$, and the upper-triangular R-factor is obtained through Cholesky factorization. The orthogonal factor $Q \in \mathbb{R}^{m\times n}$ can then be derived using $Q = XR^{-1}$. The basic form of CholeskyQR is outlined in Algorithm 1.

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

Input: $X \in \mathbb{R}^{m\times n}$.

Output: Orthogonal factor $Q \in \mathbb{R}^{m\times n}$, Upper triangular factor $R \in \mathbb{R}^{n\times n}$.

1: $G = X^\top X$,

2: $R = \text{Cholesky}(B)$,

3: $Q = XR^{-1}$.

In CholeskyQR-type algorithms, regarding the sizes of the matrices, we always define

$$mn\mathbf{u} \leq \frac{1}{64}, \tag{1.1}$$

$$n(n+1)\mathbf{u} \leq \frac{1}{64}. \tag{1.2}$$

Here, $\mathbf{u}$ is the machine precision and $\mathbf{u} = 2^{-53}$. It shows that CholeskyQR can deal with matrices with millions of dimensions.

1.2.1 CholeskyQR2

Although with many advantages, Algorithm 1 exhibits certain limitations and is seldom used directly. When considering the error of orthogonality, it is shown in [68] that

$$\left\|Q^\top Q - I\right\|_F \leq \frac{5}{64}\delta^2, \tag{1.3}$$

where

$$\delta = 8\kappa_2(X)\sqrt{mn\mathbf{u} + n(n+1)\mathbf{u}}. \tag{1.4}$$

Here, $\kappa_2(X) = \frac{\|X\|_2}{\sigma_{min}(X)}$ is the condition number of X. $\|X\|_2$ equals to the largest singular value of X while $\sigma_{min}(X)$ denotes the smallest one.

According to (1.3) and (1.4), the orthogonality error of CholeskyQR is proportional to $(\kappa_2(X))^2$. Numerous numerical experiments indicate that CholeskyQR 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 [22]. It is presented in Algorithm 2.

Algorithm 2: $[Q, R] = \text{CholeskyQR2}(X)$

Input: $X \in \mathbb{R}^{m\times n}$.

Output: Orthogonal factor $Q \in \mathbb{R}^{m\times n}$, Upper triangular factor $R \in \mathbb{R}^{n\times n}$.

1: $[W, Y] = \text{CholeskyQR}(X)$,

2: $[Q, Z] = \text{CholeskyQR}(W)$,

3: $R = ZY$.

In [68], it has been shown that compared to Algorithm 1, Algorithm 2 is numerically stable in both orthogonality and residual. Rounding error analysis of CholeskyQR2 is shown below in Lemma 1.1

Lemma 1.1. *For $X \in \mathbb{R}^{m\times n}$ and $[Q,R] = CholeskyQR2(X)$, with $\delta = 8\kappa_2(X)\sqrt{mn\mathbf{u} + n(n+1)\mathbf{u}} \leq 1$, (1.1) and (1.2), we have*

$$\left\|Q^\top Q - I\right\|_F \leq 6(mn\mathbf{u} + n(n+1)\mathbf{u}), \tag{1.5}$$

$$\|QR - X\|_F \leq 5n^2\sqrt{n}\mathbf{u}\|X\|_2 . \tag{1.6}$$

According to Lemma 1.1, CholeskyQR2 is numerically stable in terms of both orthogonality and residual compared to CholeskyQR in Algorithm 1.

1.2.2 Shifted CholeskyQR3

When X is ill-conditioned, CholeskyQR2 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 [21].

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

Input: $X \in \mathbb{R}^{m\times n}$.

Output: Orthogonal factor $Q \in \mathbb{R}^{m\times n}$, Upper triangular factor $R \in \mathbb{R}^{n\times n}$.

1: $G = X^\top X$,

2: take $s > 0$,

3: $R = \text{Cholesky}(B + sI)$,

4: $Q = XR^{-1}$.

Shifted CholeskyQR is a superior algorithm in terms of applicability compared to CholeskyQR. 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: $[Q,R] = \text{SCholeskyQR3}(X)$

Input: $X \in \mathbb{R}^{m\times n}$.

Output: Orthogonal factor $Q \in \mathbb{R}^{m\times n}$, Upper triangular factor $R \in \mathbb{R}^{n\times n}$.

1: $[W,Y] = \text{SCholeskyQR}(X)$,

2: $[Q,Z] = \text{CholeskyQR2}(W)$,

3: $R = ZY$.

For Shifted CholeskyQR and Shifted CholeskyQR3, some theoretical results are provided in [21]. They are shown below in Lemma 1.2-Lemma 1.4.

Lemma 1.2. *For $X \in \mathbb{R}^{m\times n}$ and $[Q,R] = SCholeskyQR(X)$, with $11(mn\mathbf{u}+n(n+1)\mathbf{u})\|X\|_2^2 \le s \le \frac{1}{100}\|X\|_2^2$, $\kappa_2(X) \le \frac{1}{6n^2\mathbf{u}}$, (1.1) and (1.2), we have*

$$\left\|Q^\top Q - I\right\|_2 \le 2, \tag{1.7}$$

$$\|QR - X\|_F \le 2n^2\mathbf{u}\|X\|_2 . \tag{1.8}$$

Lemma 1.3. *For $X \in \mathbb{R}^{m\times n}$ and $[W,Y] = SCholeskyQR(X)$, with $11(mn\mathbf{u}+n(n+1)\mathbf{u})\|X\|_2^2 \le s \le \frac{1}{100}\|X\|_2^2$, $\kappa_2(X) \le \frac{1}{6n^2\mathbf{u}}$, (1.1) and (1.2), we have*

$$\kappa_2(W) \le 2\sqrt{3}\cdot\sqrt{1+\alpha(\kappa_2(X))^2}. \tag{1.9}$$

Here, $\alpha = \frac{s}{\|X\|_2^2}$. When $[Q,R] = SCholeskyQR3(X)$, 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

$$\kappa_2(X) \le \frac{1}{96(mn\mathbf{u}+n(n+1)\mathbf{u})}. \tag{1.10}$$

Lemma 1.4. *For $X \in \mathbb{R}^{m\times n}$ and $[Q,R] = SCholeskyQR3(X)$, with $s = 11(mn\mathbf{u}+n(n+1)\mathbf{u})\|X\|_2^2$, (1.1), (1.2) and (1.10), we have*

$$\left\|Q^\top Q - I\right\|_F \le 6(mn\mathbf{u}+n(n+1)\mathbf{u}), \tag{1.11}$$

$$\|QR - X\|_F \le 15n^2\mathbf{u}\|X\|_2 . \tag{1.12}$$

In particular, Lemma 1.3 highlights one of the most important properties of Shifted CholeskyQR3. It demonstrates that when s is within a certain interval, increasing s results in a larger value of $\kappa_2(Q)$. Since CholeskyQR2, which follows Shifted CholeskyQR, may break down if $\kappa_2(Q)$ is large, the selection of the shifted item s is crucial for Shifted CholeskyQR3. It cannot be too large, as this would affect the applicability of Shifted CholeskyQR3, nor too small, as this could lead to the breakdown of Shifted CholeskyQR. Therefore, the most important point of Shifted CholeskyQR3 is to pick a proper s, which will greatly influence the properties of the algorithm.

1.2.3 LU-CholeskyQR2

Among all the deterministic CholeskyQR-type algorithms, LU-CholeskyQR [62] is particularly noteworthy. It has the unique advantage of not imposing a restriction on $\kappa_2(X)$, which is crucial for real-world applications in industry. LU-CholeskyQR combines LU factorization with CholeskyQR, as shown in Algorithm 5. Here, $L \in \mathbb{R}^{m\times n}$ is a unit tall-skinny lower triangular matrix, and $U \in \mathbb{R}^{n\times n}$ is an upper triangular matrix when $X \in \mathbb{R}^{m\times n}$ is tall-skinny. A CholeskyQR step is performed afterwards to form LU-CholeskyQR2, ensuring orthogonality, as detailed in Algorithm 6. There are also some other works discussing CholeskyQR-type algorithms, see [69, 70].

Algorithm 5: $[Q, R] = \text{LU-CholeskyQR}(X)$

Input: $X \in \mathbb{R}^{m\times n}$.

Output: Orthogonal factor $Q \in \mathbb{R}^{m\times n}$, Upper triangular factor $R \in \mathbb{R}^{n\times n}$.

1: $PX = LU$,

2: $G = L^\top L$,

3: $S = \text{Cholesky}(G)$,

4: $R = SU$,

5: $Q = XR^{-1}$.

Algorithm 6: $[Q, R] = \text{LU-CholeskyQR2}(X)$

Input: $X \in \mathbb{R}^{m\times n}$.

Output: Orthogonal factor $Q \in \mathbb{R}^{m\times n}$, Upper triangular factor $R \in \mathbb{R}^{n\times n}$.

1: $[W, Y] = \text{LU-CholeskyQR}(X)$,

2: $[Q, Z] = \text{CholeskyQR}(W)$,

3: $R = ZY$.

Regarding LU-CholeskyQR2 in Algorithm 6, both (1.1) and (1.2) are required. Additionally, the following assumptions concerning $\kappa_2(L)$ and $\kappa_2(U)$ are shown below.

$$8\kappa_2(L)\sqrt{mn\mathbf{u} + n(n+1)\mathbf{u}} \le 1, \tag{1.13}$$

$$64n^2\mathbf{u} \cdot \kappa_2(L)\kappa_2(U) \le 1. \tag{1.14}$$

Under the assumptions stated above, we present rounding error analysis of LU-CholeskyQR2 [62] in Lemma 1.5.

Lemma 1.5. *For $X \in \mathbb{R}^{m\times n}$ and $[Q, R] = $ LU-CholeskyQR2(X), when* (1.1)*,* (1.2)*,* (1.13) *and* (1.14) *are satisfied, we have*

$$\left\| Q^\top Q - I \right\|_2 \le 6(mn\mathbf{u} + n(n+1)\mathbf{u}), \tag{1.15}$$

$$\|QR - X\|_2 \le 4.09n^2\mathbf{u}\|X\|_2 . \tag{1.16}$$

According to Lemma 1.5, we find that LU-CholeskyQR2 is as stable as CholeskyQR2.

1.2.4 Randomized algorithms

In recent years, randomized numerical linear algebra [30, 47] has become a hot topic worldwide. A new randomized technique called sketching [1] has been widely applied to various problems. The sketching technique primarily aims to reduce computational costs by replacing the original large matrices with alternative matrices of smaller sizes after some preconditioning steps. The sketching matrix is typically

chosen to be an ϵ-subspace embedding or a linear map to a lower-dimensional space, preserving the inner products and norms of all vectors within the subspace up to a factor of $\sqrt{1+\epsilon}$, where $0 \leq \epsilon < 1$. Several existing methods for sketching matrices include Gaussian Sketch, SRHT, and Count Sketch [4, 42, 56, 57, 73]. Research on randomized CholeskyQR has also emerged. Y. Fan and his collaborators [20] proposed the initial version of randomized CholeskyQR, while O. Balabanov [3] provided a detailed analysis of CholeskyQR-type algorithms with various randomized strategies, including sketching. A recent work by A. J. Higgins and his collaborators [31] introduced a novel method called multi-sketching, which employs two different sketching steps consecutively. This work developed a new algorithm called Randomized Householder QR(RHQR), utilizing HouseholderQR to replace the original generation step of the R-factor in CholeskyQR to avoid numerical breakdown. The multi-sketching technique is designed to accelerate the entire algorithm. The corresponding algorithm is outlined in Algorithm 7. Here, $\Omega_1 \in \mathbb{R}^{s_1 \times m}$ is the matrix for CountSketch while $\Omega_2 \in \mathbb{R}^{s_2 \times s_1}$ is the matrix for Gaussian Sketch, with $n \leq s_2 \leq s_1 \leq 1$. To ensure good orthogonality, a CholeskyQR step is added afterwards, resulting in the Rand.Householder-Cholesky algorithm(RHC), as shown in Algorithm 8. They provide a clear rounding error analysis of the algorithms with two ϵ-subspace embeddings, and numerical experiments demonstrate that their Randomized Householder-CholeskyQR is more efficient and numerically stable than the algorithms in [3].

Algorithm 7: $[Q, R] = \mathrm{RHQR}(X)$

Input: $X \in \mathbb{R}^{m \times n}$.

Output: Orthogonal factor $Q \in \mathbb{R}^{m \times n}$, Upper triangular factor $R \in \mathbb{R}^{n \times n}$.

1: $K = \Omega_2 \Omega_1 X$,

2: $[W, R] = \mathrm{HouseholderQR}(K)$,

3: $Q = XR^{-1}$.

Algorithm 8: $[Q, R] = \mathrm{RHC}(X)$

Input: $X \in \mathbb{R}^{m \times n}$.

Output: Orthogonal factor $Q \in \mathbb{R}^{m \times n}$, Upper triangular factor $R \in \mathbb{R}^{n \times n}$.

1: $[W, Y] =$ Randomized Householder $\mathrm{QR}(X)$

2: $[Q, Z] = \mathrm{CholeskyQR}(W)$,

3: $R = ZY$.

1.3 Theoretical results of the existing algorithms and some considerations

According to the previous algorithms, we find that all the CholeskyQR-type algorithms are all in the form of 'Preconditioning step+CholeskyQR/CholeskyQR2'. The primary purpose of this type of structure is due to (1.3) and (1.4). If $\kappa_2(X)$ of the input matrix X for the last step of CholeskyQR is

very small, the whole algorithm can have good orthogonality. Otherwise, we need to put CholeskyQR2 after the preconditioning step to guarantee numerical stability. It is also easy to prove the numerical stability of residual for CholeskyQR and the corresponding preconditioning steps in the level of '$n^2\mathbf{u}\|X\|_2$' if $X \in \mathbb{R}^{m\times n}$.

1.3.1 Theoretical results of the existing algorithms

In this section, we show the theoretical properties of some CholeskyQR-type algorithms. We measure the properties of these numerical algorithms from three perspectives, numerical stability, applicability and efficiency, which corresponds to the theoretical upper bounds of orthogonality and residual($\left\|Q^\top Q - I\right\|_F$ and $\|QR - X\|_F$), upper bounds of $\kappa_2(X)$ for the input matrix $X \in \mathbb{R}^{m\times n}$. Although some randomized algorithms are introduced in Section 1.2.4, we focus on deterministic algorithms in this thesis. The comparisons of the theoretical results are listed in Table 1.1.

Table 1.1: Upper bounds of $\kappa_2(X)$, orthogonality and residual for $X \in \mathbb{R}^{m\times n}$

Algorithms	CholeskyQR2	SCholeskyQR3	LU-CholeskyQR2
$\kappa_2(X)$	$\frac{1}{8\sqrt{mn\mathbf{u}+n(n+1)\mathbf{u}}}$	$\frac{1}{96(mn\mathbf{u}+n(n+1)\mathbf{u})}$	No requirements
Orthogonality	$6(mn\mathbf{u}+n(n+1)\mathbf{u})$	$6(mn\mathbf{u}+n(n+1)\mathbf{u})$	$6.5(mn\mathbf{u}+n(n+1)\mathbf{u})$
Residual	$5n^2\sqrt{n}\|X\|_2$	$15n^2\mathbf{u}\|X\|_2$	$4.09n^2\mathbf{u}\|X\|_2$

1.3.2 Considerations of the existing algorithms

Although CholeskyQR2 is numerical stable and the computational cost of CholeskyQR2 is about $\frac{2}{3}$ of that of Shifted CholeskyQR3 according to Lemma 1.1, it is a very vulnerable algorithm for the existence of Cholesky factorization in calculating the first R-factor Y in Algorithm 2. A sufficient condition for Cholesky factorization here to work on is that W must be positive definite. However, with the existence of the rounding errors in calculating the gram matrix W and Cholesky factorization, W may not be positive definite if $\kappa_2(X)$ is large enough for the input matrix X, which will lead to numerical breakdown in Cholesky factorization. Therefore, there is a sufficient condition of $\kappa_2(X)$ for CholeskyQR2 according to Table 1.1. The primary target for the research regarding CholeskyQR-type algorithms is improving its applicability. From this perspective, Shifted CholeskyQR3 can almost 'cover' all the properties of CholeskyQR2 and deal with ill-conditioned cases with numerical stability, see the comparison in Table 1.1 and Lemma 1.4. Therefore, the properties of Shifted CholeskyQR3 are primarily discussed in this thesis.

Regarding Shifted CholeskyQR3, we write the first two steps of Shifted CholeskyQR with error

matrices below.

$$G = X^\top X + E_A, \tag{1.17}$$

$$R^\top R = G + E_B + sI. \tag{1.18}$$

In fact, the error bounds for $\|E_A\|_2$ and $\|E_B\|_2$ in (1.17) and (1.18) significantly influence the choice of the shifted item s. In [21], 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. In most cases, the 2-norm of each column of X can be significantly smaller than $\|X\|_2$. This overestimation leads to a conservative choice of s, limiting the applicability of Shifted CholeskyQR3 for matrices X with a large $\kappa_2(X)$. Therefore, one of our primary objectives is to select a smaller shifted item 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. Such an alternative s improves the applicability of Shifted CholeskyQR3, reflected in a better sufficient condition for $\kappa_2(X)$ to some extent.

Furthermore, the rounding error analysis of [21, 68] on CholeskyQR-type algorithms is primarily based on deterministic models by Higham [32]. However, using these deterministic models may lead to overestimating the norms of error matrices, especially $\|E_A\|_2$ and $\|E_B\|_2$ in (1.17) and (1.18). This may result in a conservative s and poorer sufficient conditions for $\kappa_2(X)$ for Shifted CholeskyQR3. In floating-point arithmetic, the norms of the error matrices rarely reach the upper bounds predicted by deterministic models. Recently, randomized linear algebra has gained popularity, with several works addressing probabilistic error analysis using the randomized models, see [10, 37, 74] and related references. Many tools and conclusions regarding the randomized models for probabilistic error analysis [11, 33, 35, 66, 67] have been developed, which can significantly improve the error analysis. In the context of rounding error analysis for matrix multiplications, the randomized models have been provided in [11] to provide smaller upper bounds for error estimations, see Section 1.5.2. Since the theoretical results of CholeskyQR-type algorithms are primarily based on the estimations of some error matrices, we can apply the randomized models to CholeskyQR-type algorithms to conduct new error analysis. Such advancements can also improve various properties of these algorithms, including the shifted item s in Shifted CholeskyQR3.

Generally speaking, all the existing algorithms and our previous thoughts are designed for QR factorization of dense X. However, in many real-world problems such as numerical PDEs and their applications in physics, chemistry, and astronomy, X is often large and sparse. Sparse matrices often exhibit different properties compared to those of dense matrices. In recent years, there are many works for the design of algorithms and the analysis of properties for sparse matrices, see [2, 13, 14, 23, 72, 73]

and their references. In CholeskyQR, it is meaningful for us to do analysis and explore the properties of CholeskyQR-type algorithms for sparse matrices. Regarding sparsity, we aim to receive a new perspective on CholeskyQR-type algorithms based on the structure of X. When the input matrix X is sparse, it is possible for us to achieve a different and more accurate rounding error analysis based on its structure, thus leading to a better shifted item s for Shifted CholeskyQR3.

1.4 Our contributions

Based on all our considerations before, we do some innovative works with some new techniques regarding CholeskyQR. The primary target of us is to improve the preconditioning steps of CholeskyQR-type algorithms, together with the theoretical analysis of the algorithms.

In Chapter 2, we define a new $\|\cdot\|_c$ in Definition 2.1 as $\|X\|_c = \sqrt{n}\|X\|_g$ for $X \in \mathbb{R}^{m\times n}$, where $\|\cdot\|_g$ denotes the largest 2-norm of the columns of X. Some properties of $\|\cdot\|_c$ are shown in Section 2.1. With $\|\cdot\|_c$, we can take a smaller s for Shifted CholeskyQR3 as $s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$ and construct the improved Shifted CholeskyQR (ISCholeskyQR) and improved Shifted CholeskyQR3 (ISCholeskyQR3) in Section 2.3.2 for the input matrix $X \in \mathbb{R}^{m\times n}$. Our rounding error analysis demonstrates that this improved s based on $\|X\|_c$ can keep numerical stability of Shifted CholeskyQR3 as reflected in Theorem 2.3. Numerical experiments show that our improved Shifted CholeskyQR3 has better applicability compared to Shifted CholeskyQR3 with the original s [21] while achieving numerical stability and efficiency comparable to those of the original algorithm. The comparisons between the original Shifted CholeskyQR3 and our improved one are listed in Table 1.2 and Table 1.3. Here, j is defined in (2.7). We discover the relationship between the column of X and CholeskyQR in this chapter. The definition of $\|\cdot\|_c$ provides us with a new perspective on CholeskyQR-type algorithms and will be utilized in our subsequent research.

Table 1.2: Comparison of $\kappa_2(X)$ between the improved and the original s

s	Sufficient condition of $\kappa_2(X)$	Upper bound of $\kappa_2(X)$
$11(mn\mathbf{u} + n(n+1)\mathbf{u})\|X\|_2^2$	$\frac{1}{96(mn\mathbf{u}+n(n+1)\mathbf{u})}$	$\frac{1}{6n^2\mathbf{u}}$
$11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$	$\frac{1}{86j(m\sqrt{n}\mathbf{u}+(n+1)\sqrt{n}\mathbf{u})}$	$\frac{1}{4.89jn\sqrt{n}\mathbf{u}}$

Table 1.3: Comparison of the upper bounds of residual between the improved and the original s

s	SCholeskyQR	SCholeskyQR3
$11(mn\mathbf{u} + n(n+1)\mathbf{u})\|X\|_2^2$	$2n^2\mathbf{u}\|X\|_2$	$15n^2\mathbf{u}\|X\|_2$
$11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$	$1.6n\sqrt{n}\mathbf{u}\|X\|_c$	$(6.57\cdot\frac{j}{\sqrt{n}} + 4.87)n^2\mathbf{u}\|X\|_2$

In Chapter 3, we combine the properties of sparse matrices with theoretical analysis, which is the first to build connections between sparsity and rounding error analysis to the best our knowledge. We introduce a new classification for the sparse $X \in \mathbb{R}^{m\times n}$ based on the presence of dense columns, dividing sparse matrices into T_1 and T_2 matrices. For Shifted CholeskyQR3, when the input matrix X is sparse, we propose an alternative choice of s in (3.1) based on the structure and the element with the largest absolute value of X according to Definition 3.1. This approach differs significantly from those in [21]. We prove that this alternative s can prevent numerical breakdown and ensure numerical stability of Shifted CholeskyQR3 with proper element-norm conditions(ENCs) in Theorem 3.1 and Theorem 3.2. For T_1 matrices satisfying these ENCs, such an s and the corresponding sufficient condition of $\kappa_2(X)$ are significantly better than those of our improved Shifted CholeskyQR3. The theoretical analysis in this part is deeply tied to the properties of $\|\cdot\|_g$ as shown in Chapter 2. Numerical experiments illustrate the properties of Shifted CholeskyQR3 for sparse matrices and confirm the effectiveness of the improved s for T_1 matrices satisfying proper ENCs. Additionally, Shifted CholeskyQR3 can handle more ill-conditioned cases for T_2 matrices compared to dense cases. Moreover, the efficiency of Shifted CholeskyQR3 with our alternative s is comparable to that of the original s in Chapter 2 for sparse matrices. The comparisons between the improved Shifted CholeskyQR3 and our alternative for T_1 matrices are listed in Table 1.4 and Table 1.5. Here, l, h and k are mentioned in Section 3.1.2 and Theorem 3.1. As far as we know, this work is the first to explore the connection between QR factorization and sparse matrices and provide detailed theoretical analysis, which is very meaningful in the research of sparse matrices and many real applications.

Table 1.4: Comparison of $\kappa_2(X)$ between the improved and the alternative s for T_1 matrices

s	Sufficient condition of $\kappa_2(X)$	Upper bound of $\kappa_2(X)$
$11(m\mathbf{u}+(n+1)\mathbf{u})\|X\|_c^2$	$\frac{1}{86j(m\sqrt{n}\mathbf{u}+(n+1)\sqrt{n}\mathbf{u})}$	$\frac{1}{4.89jn\sqrt{n}\mathbf{u}}$
$11(m\mathbf{u}+(n+1)\mathbf{u})\cdot(vt_1+nt_2)c^2$	$\frac{1}{16\sqrt{11nk}\cdot(m\mathbf{u}+(n+1)\mathbf{u})h}$	$\frac{1}{4n^2\mathbf{u}hl}$

Table 1.5: Comparison of the upper bounds of $\|QR-X\|_F$ between the improved and the alternative s for T_1 matrices

s	SCholeskyQR	SCholeskyQR3
$11(m\mathbf{u}+(n+1)\mathbf{u})\|X\|_c^2$	$1.6n\sqrt{n}\mathbf{u}\|X\|_c$	$(6.57\cdot\frac{j}{\sqrt{n}}+4.87)n^2\mathbf{u}\|X\|_2$
$11(m\mathbf{u}+(n+1)\mathbf{u})\cdot(vt_1+nt_2)c^2$	$1.03hln^2\mathbf{u}\|X\|_2$	$(2.19+3.4l)hn^2\mathbf{u}\|X\|_2$

We have already discussed the sparse cases which can lead to better error analysis and theoretical results compared to our improved Shifted CholeskyQR3 in Chapter 3. In Chapter 4, we focus on

common cases again based on the randomized models in recent years, doing probabilistic error analysis of CholeskyQR-type algorithms with such tools. We apply the randomized models in [33] to provide probabilistic error analysis of CholeskyQR2 and Shifted CholeskyQR3 for the input matrix $X \in \mathbb{R}^{m\times n}$, which also utilizes $\|X\|_c$ defined in Chapter 2. Specially, a new probabilistic s is also taken in Theorem 4.3 for Shifted CholeskyQR3. We can get tighter upper bounds for both orthogonality and residual in Theorem 4.1 and Theorem 4.3. Numerical experiments demonstrate that such a probabilistic s can improve the applicability of Shifted CholeskyQR3 again compared to our work in Chapter 2. We also show the robustness of Shifted CholeskyQR3 with such an s through extensive and numerous experiments. The comparisons between the theoretical results of the deterministic and probabilistic error analysis are listed in Table 1.6-Table 1.9. Here, j_1, j_2, j_3, L and $\phi_1(j_1, j_2, j_3, n)$ are defined in Section 4.1.1, Section 4.2.1 and Theorem 4.3. η is a positive constant in the randomized models, which occurred in Section 1.5.2. Our work is the first to conduct probabilistic error analysis for CholeskyQR-type algorithms. The combination of $\|\cdot\|_c$ and the randomized models in analysis is distinct from other works regarding probabilistic error analysis. The utilization of $\|\cdot\|_c$ can also minimize the influence of the constant η in the randomized models which are shown in Section 1.5.2.

Table 1.6: Comparison of $\kappa_2(X)$ of CholeskyQR2 between the deterministic and the probabilistic analysis

Type of analysis	Sufficient condition of $\kappa_2(X)$
Deterministic	$\frac{1}{8\sqrt{mn\mathbf{u}+n(n+1)\mathbf{u}}}$
Probabilistic	$\frac{1}{8j_1\sqrt{\eta(\sqrt{m}\mathbf{u}+\sqrt{n+1}\mathbf{u})}}$

Table 1.7: Comparison of the upper bounds of CholeskyQR2 between the deterministic and the probabilistic analysis

Type of analysis	$\left\|Q^\top Q - I\right\|_F$	$\|QR - X\|_F$
Deterministic	$6(mn\mathbf{u} + n(n+1)\mathbf{u})$	$5n^2\sqrt{n}\mathbf{u}\|X\|_2$
Probabilistic	$6\eta \cdot j_2^2(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u})$	$(1.2j_1 + 1.32j_2 + 1.32 \cdot \frac{j_1 j_2}{\sqrt{n}})\eta \cdot n\mathbf{u}\|X\|_2$

Table 1.8: Comparison of $\kappa_2(X)$ of Shifted CholeskyQR3 between the improved and the probabilistic s

s	Sufficient condition of $\kappa_2(X)$	Upper bound of $\kappa_2(X)$
$11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$	$\frac{1}{86j(m\sqrt{n}\mathbf{u}+(n+1)\sqrt{n}\mathbf{u})}$	$\frac{1}{4.89jn\sqrt{n}\mathbf{u}}$
$11\eta(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u})\|X\|_c^2$	L	$\frac{1}{4.89j_1\cdot\eta n\mathbf{u}}$

Table 1.9: Comparison of the upper bounds of Shifted CholeskyQR3 between the improved and the probabilistic s

s	$\left\|Q^\top Q - I\right\|_F$	$\|QR - X\|_F$
$11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$	$6(mn\mathbf{u} + n(n+1)\mathbf{u})$	$(6.57p + 4.87)n^2\mathbf{u}\|X\|_2$
$11\eta(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u})\|X\|_c^2$	$6\eta \cdot j_3^2(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u})$	$\phi_1(j_1, j_2, j_3, n)\eta \cdot n\mathbf{u}\|X\|_2$

1.5 Some preliminaries for the theoretical analysis

Before presenting detailed theoretical analysis of CholeskyQR-type algorithms in this thesis, we introduce some preliminaries related to deterministic rounding error analysis and probabilistic error analysis in this section. They are widely used in Chapter 2-Chapter 4.

1.5.1 Deterministic rounding error analysis

In the beginning of this section, we show the following classical model for floating-point arithmetic from [32].

$$fl(a \quad \text{op} \quad b) = (a \quad \text{op} \quad b)(1+\delta), \quad |\delta| \le u, \quad \text{op} \in \{+, -, \times, /, \sqrt{}\}. \tag{1.19}$$

Here, $fl(\cdot)$ denotes the computed value in floating-point arithmetic. This model holds for IEEE arithmetic and the IEEE standard even requires that $fl(a \quad \text{op} \quad b)$ be the correctly rounded (to nearest) value of $a \quad \text{op} \quad b$. We will refer to δ as the rounding error in the operation, though it is perhaps more common to describe the absolute error $a \quad \text{op} \quad b - fl(a \quad \text{op} \quad b)$ in this way.

This thesis primarily focuses on rounding error analysis originating from (1.19). In the following, we show some fundamental lemmas of deterministic rounding error analysis [25, 32], which are widely used in the error estimations of numerical linear algebra.

Lemma 1.6. *If $A, B \in \mathbb{R}^{m\times n}$, then*

$$\sigma_{min}(A+B) \ge \sigma_{min}(A) - \|B\|_2 .$$

Lemma 1.7. *For $A \in \mathbb{R}^{m\times n}, B \in \mathbb{R}^{n\times p}$, the error in computing the matrix product AB in floating-point arithmetic is bounded by*

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

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

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

Lemma 1.8. *If Cholesky factorization applied to the symmetric positive definite $A \in \mathbb{R}^{n\times n}$ runs to completion, then the computed factor $R \in \mathbb{R}^{n\times n}$ satisfies*

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

Lemma 1.9. *Let the triangular system $Tx = b$, where $T \in \mathbb{R}^{n\times n}$ is non-singular, be solved by institution with any ordering. Then the computed solution x satisfies*

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

To learn more about matrix perturbations, readers can refer to [38, 51, 58] for more details.

1.5.2 Probabilistic error analysis

In this section, we introduce the probabilistic error bounds in the probabilistic techniques and present the following lemmas related to probabilistic error analysis. We show the probabilistic model of rounding errors [33] first.

Lemma 1.10. *In the computation of interest, the quantities δ in* (1.19) *associated with every pair of operands are independent random variables of mean zero.*

Before showing the lemmas of probabilistic error analysis, we define

$$P(\eta) = 1 - 2\exp(-\frac{\eta^2(1-\mathbf{u})^2}{2}), \tag{1.20}$$

$$Q(\eta, n) = 1 - n(1 - P(\eta)), \tag{1.21}$$

$$\tilde{\gamma}_n = \exp(\eta\sqrt{n}\mathbf{u} + \frac{n\mathbf{u}^2}{1-\mathbf{u}}) - 1. \tag{1.22}$$

Here, η is a positive constant. From (1.22), we can find that when $\eta\sqrt{n}\mathbf{u}$ is small and close to 0, $\tilde{\gamma}_n \approx 1.02\eta\sqrt{n}\mathbf{u}$. This setting is used in the following of this work. Below, we present several lemmas from [33] for probabilistic error analysis, which corresponds to Lemma 1.7-Lemma 1.9.

Lemma 1.11. *For $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{n\times p}$, under Lemma 1.10, the error in computing the matrix product $C = AB$ in floating-point arithmetic satisfies*

$$\left|AB - fl(AB)\right| \leq \tilde{\gamma_n}(\eta)|A||B|\,,$$

with probability at least $Q(\eta, mnp)$.

Lemma 1.12. *If Cholesky factorization applied to the symmetric positive definite matrix $A \in \mathbb{R}^{n\times n}$ runs to completion, under Lemma 1.10, the computed factor $R \in \mathbb{R}^{n\times n}$ satisfies*

$$R^\top R = A + \Delta A, \quad |\Delta R| \leq \tilde{\gamma}_{n+1}(\eta)\left|R^\top\right||R|\,,$$

with probability at least $Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$.

Lemma 1.13. *Let the triangular system $Tx = b$, where $T \in \mathbb{R}^{n\times n}$ is non-singular, be solved by institution with any ordering. Under Lemma 1.10, the computed solution x satisfies*

$$(T + \Delta T)x = b, \quad |\Delta T| \leq \tilde{\gamma}_n(\eta)|T|\,,$$

with probability at least $Q(\eta, \frac{n(n+1)}{2})$.

1.6 Outline of this thesis

The remainder of this thesis is organized as follows. In Chapter 2, we introduce and analyze the properties of the improved Shifted CholeskyQR based on the new matrix norm $\|\cdot\|_c$. Chapter 3 focuses on Shifted CholeskyQR for sparse matrices with a new division of sparse matrices based on the presence of dense columns. In Chapter 4, we show the probabilistic error analysis of CholeskyQR-type algorithms with $\|\cdot\|_c$. We show the conclusion of this thesis and list some potential directions for our future work in Chapter 5. The content of this thesis is based on several existing articles, including [19, 27, 29].

CHAPTER 2.
AN IMPROVED SHIFTED CHOLESKYQR BASED ON COLUMNS

In this chapter, we focus on taking an optimal choice of the shifted item s for Shifted CholeskyQR3. We introduce a new matrix norm $\|X\|_c$ for the input matrix $X \in \mathbb{R}^{m\times n}$, which is based on the column properties of the input matrix. Thus, we can take an improved s with $\|X\|_c$. We show that such an s can improve the applicability of Shifted CholeskyQR3 while maintaining its numerical stability and efficiency from both theoretical analysis and numerical experiments. This chapter is organized as follows. In Section 2.1, we present the definition of $\|\cdot\|_c$ and some of its properties. Section 2.2 outlines the primary theorems of the improved Shifted CholeskyQR3. The theoretical analysis of the improved Shifted CholeskyQR3 are detailed in Section 2.3, which serves as the key contribution of this chapter. Furthermore, Section 2.4 presents numerical results of the improved Shifted CHoleskyQR3 and the comparison between our new algorithm and the existing algorithms. Section 2.5 is a summary of this chapter.

2.1 $\|\cdot\|_c$ and its properties

In this section, we introduce a new matrix norm $\|\cdot\|_c$. Before introducing $\|\cdot\|_c$, we take consideration of the largest 2-norm among all the columns of X, which is defined as $\|X\|_g$ in Definition 2.1.

Definition 2.1. *For $X = [X_1, X_2, \cdots X_{n-1}, X_n] \in R^{m\times n}$,*

$$\|X\|_g := \max_{1\le j\le n} \|X_j\|_2 , \tag{2.1}$$

where

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

In the following, we present several properties of $\|X\|_g$ of the matrix, which will be used in the theoretical analysis of this thesis.

Lemma 2.1. *For $X \in \mathbb{R}^{m\times n}$, we have*

$$\|X\|_g \le \|X\|_2 \le \|X\|_F .$$

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

Lemma 2.2. *For $X, Y \in \mathbb{R}^{m\times n}$, we have*

$$\|X+Y\|_g \le \|X\|_g + \|Y\|_g . \tag{2.2}$$

Proof. Based on Definition 2.1 and the triangular inequality of the norms of vectors, we can easily get (2.2). □

Lemma 2.3. *For $X \in \mathbb{R}^{m\times p}$ and $Y \in \mathbb{R}^{p\times n}$, we have*

$$\|XY\|_g \le \|X\|_2 \|Y\|_g , \quad \|XY\|_g \le \|X\|_F \|Y\|_g . \tag{2.3}$$

Proof. Regarding $\|XY\|_g$, with Definition 2.1, we have

$$\begin{aligned}
\|XY\|_g &\le \max(\|XY_1\|_2 , \|XY_2\|_2 , \cdots , \|XY_n\|_2) \\
&\le \max(\|X\|_2 \|Y_1\|_2 , \|X\|_2 \|Y_2\|_2 , \cdots , \|X\|_2 \|Y_n\|_2) \\
&\le \|X\|_2 \cdot \max(\|Y_1\|_2 , \|Y_2\|_2 , \cdots , \|Y_n\|_2) \\
&\le \|X\|_2 \|Y\|_g .
\end{aligned}$$

Here, the first inequality of (2.3) is received. Since $\|X\|_2 \le \|X\|_F$, it is easy to get the second inequality of (2.3). □

Though $\|\cdot\|_g$ is a matrix norm, it is not sub-multiplicative. With $\|\cdot\|_g$, we introduce a new $\|\cdot\|_c$ in Definition 2.2.

Definition 2.2. *When $X \in \mathbb{R}^{m\times n}$, we define $\|X\|_c$ as*

$$\|X\|_c = \sqrt{n}\|X\|_g .$$

With Definition 2.2, we prove that $\|X\|_c$ is a matrix norm in Lemma 2.4.

Lemma 2.4. *For $X \in \mathbb{R}^{m\times n}$, $\|X\|_c$ is a matrix norm.*

Proof. The non-negativity of $\|X\|_c$ when $X \in \mathbb{R}^{m\times n}$ is clear according to Definition 2.1. For X, $Y \in \mathbb{R}^{m\times n}$, with (2.2), we can get the triangular inequality

$$\sqrt{n}\|X+Y\|_g \le \sqrt{n}(\|X\|_g + \|Y\|_g). \tag{2.4}$$

For $X \in \mathbb{R}^{m\times n}$ and $Y \in \mathbb{R}^{n\times p}$, with (2.3), we can have

$$\begin{aligned}
\sqrt{p}\|XY\|_g &\le \sqrt{p}\|X\|_F \|Y\|_g \\
&\le (\sqrt{n}\|X\|_g) \cdot (\sqrt{p}\|Y\|_g).
\end{aligned} \tag{2.5}$$

Based on (2.4) and (2.5), $\|X\|_c$ is a matrix norm. □

For $\|\cdot\|_c$, we show the relationship between $\|\cdot\|_c$ and other matrix norms.

Lemma 2.5. *For $X \in \mathbb{R}^{m\times n}$, we have*

$$\|X\|_2 \le \|X\|_F \le \|X\|_c \le \sqrt{n}\|X\|_2 . \tag{2.6}$$

Proof. (2.6) is easy to get with Definition 2.2 and Lemma 2.1. □

In this thesis, we use $\|\cdot\|_c$ to improve the properties of Shifted CholeskyQR3. $\|\cdot\|_g$ is also used in some steps of theoretical analysis.

2.2 Theorems of the improved Shifted CholeskyQR3

With $\|\cdot\|_c$, we can estimate $\|E_A\|_2$ and $\|E_B\|_2$ with tighter upper bounds based on $\|X\|_c$ for the input matrix X. A smaller s with $\|X\|_c$ can be taken for Shifted CholeskyQR3. Regarding $\|X\|_c$, we define a constant j as

$$j = \frac{\|X\|_c}{\|X\|_2}. \tag{2.7}$$

Here, $1 \le j \le \sqrt{n}$. j is taken for the comparison of residual with $\|\cdot\|_2$. We present the following theorems related to the improved Shifted CholeskyQR (ISCholeskyQR) and improved Shifted CholeskyQR3 (ISCholeskyQR3).

Theorem 2.1 (Rounding error analysis of the improved Shifted CholeskyQR)**.** *For $X \in \mathbb{R}^{m\times n}$ and $[Q, R] = ISCholeskyQR(X)$, with $11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2 \le s \le \frac{1}{100n}\|X\|_c^2$ and $\kappa_2(X) \le \frac{1}{4.89jn\sqrt{n}\mathbf{u}}$, we have*

$$\left\|Q^\top Q - I\right\|_2 \le 1.6, \tag{2.8}$$

$$\|QR - X\|_F \le 1.67jn\sqrt{n}\mathbf{u}\|X\|_2 . \tag{2.9}$$

Theorem 2.2 (The relationship between $\kappa_2(X)$ and $\kappa_2(Q)$ for the improved Shifted CholeskyQR)**.** *For $X \in \mathbb{R}^{m\times n}$ and $[W, Y] = ISCholeskyQR(X)$, with $11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2 \le s \le \frac{1}{100n}\|X\|_c^2$ and $\kappa_2(X) \le \frac{1}{4.89jn\sqrt{n}\mathbf{u}}$, we have*

$$\kappa_2(W) \le 3.24\sqrt{1 + t(\kappa_2(X))^2}. \tag{2.10}$$

Here, we have $t = \frac{s}{\|X\|_2^2} \le \frac{1}{100}$. When $[Q, R] = ISCholeskyQR3(X)$, if we take $s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$ and $\kappa_2(X)$ is large enough, a sufficient condition for $\kappa_2(X)$ is

$$\begin{aligned}\kappa_2(X) &\le \frac{1}{86j(m\sqrt{n}\mathbf{u} + (n+1)\sqrt{n}\mathbf{u})} \\ &\le \frac{1}{4.89jn\sqrt{n}\mathbf{u}}.\end{aligned} \tag{2.11}$$

Theorem 2.3 (Rounding error analysis of the improved Shifted CholeskyQR3)**.** *For $X \in \mathbb{R}^{m\times n}$ and $[Q, R] = ISCholeskyQR3(X)$, with $s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$ and (2.11), we have*

$$\left\|Q^\top Q - I\right\|_F \leq 6(mn\mathbf{u} + n(n+1)\mathbf{u}), \tag{2.12}$$

$$\|QR - X\|_F \leq (6.57 \cdot \frac{j}{\sqrt{n}} + 4.87)n^2\mathbf{u}\|X\|_2 . \tag{2.13}$$

Theorem 2.1-Theorem 2.3 correspond to Lemma 1.2-Lemma 1.4, respectively, which are proved in Section 2.3.4-Section 2.3.6. These theorems show that the improved Shifted CholeskyQR3 has a better sufficient condition of $\kappa_2(X)$ compared to the original one in [21]. Consequently, the improved Shifted CholeskyQR3 can effectively handle more ill-conditioned X, as shown in the numerical experiments in Section 3.3. The properties of the Y-factor can also be described by the $\|X\|_c$, which will loosen the upper bound of $\kappa_2(X)$. From a theoretical perspective, we prove the numerical stability of the improved Shifted CholeskyQR3 in Section 2.3 and provide tighter theoretical upper bounds of the residual $\|QR - X\|_F$ using the properties of $\|X\|_c$ compared to the original one in [21]. This provides new insights into the problem of rounding error analysis. The definition of $\|\cdot\|_g$ and $\|\cdot\|_c$ shows the connection between CholeskyQR-type algorithms and the column properties of the input matrix. Similar definitions regarding the largest or smallest norm among the columns or the rows of a matrix is widely used in many other problems, such as some methods for low-rank approximation and matrix factorization [8, 26], together with strategies for some iteration methods, *e.g.*, Randomized Kaczmarz method [61].

Defining $\|\cdot\|_c$ and $\|\cdot\|_g$ offers several advantages. $\|\cdot\|_c$ is a more accurate approach for researchers to estimate $\|\cdot\|_F$, as shown in (2.6). 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 $\|\cdot\|_g$ 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, $\|\cdot\|_g$ enables better utilization of the matrix structure and the inherent properties of its elements, while $\|\cdot\|_2$ 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 $\|\cdot\|_g$ offers a novel approach to rounding error analysis for matrices, based on their structures and elements. In Chapter 3, we utilize the properties of $\|\cdot\|_g$ to conduct an error analysis for Shifted CholeskyQR3 in sparse cases. Although this perspective is not directly evident from numerical experiments in this chapter, it represents an innovative advancement compared to existing results.

2.3 Theoretical analysis of the improved Shifted CholeskyQR3

In this section, we provide the theoretical analysis of the improved Shifted CholeskyQR3 with an s based on $\|\cdot\|_c$ of the input $X \in \mathbb{R}^{m\times n}$. In this section, we present the relevant settings and lemmas for the improved Shifted CholeskyQR3, and we prove Theorem 2.1-Theorem 2.3 theoretically.

2.3.1 General settings and assumptions

Given the presence of rounding errors at each step of the algorithm, we express the first Shifted CholeskyQR of Algorithm 4 with error matrices as follows.

$$G = X^\top X + E_A, \tag{2.14}$$

$$Y^\top Y = G + sI + E_B, \tag{2.15}$$

$$w_i^\top = x_i^\top (Y + E_{Yi})^{-1}, \tag{2.16}$$

$$WY = X + E_X. \tag{2.17}$$

We let $w_i^\top$ and $x_i^\top$ represent the i-th rows of w and Q respectively. The error matrix E_A in (2.14) denotes the discrepancy generated when calculating the Gram matrix $X^\top X$. Similarly, E_B in (2.15) represents the error matrix after performing Cholesky factorization on G with a shifted item. Since Y may be non-invertible, the $w_i^\top$ can be solved by solving the linear system $(Y^\top + (\Delta Y_i)^\top)(w_i^\top)^\top = (x_i^\top)^\top$, that is, the transpose of (2.16). We do not write this step into the form of the whole matrices because each ΔY_i depends on Y and $x_i^\top$, where E_{Yi} denotes the rounding error for the Y-factor when calculating $w_i^\top$. In spite of this, ΔY_i has an uniform upper bound according to Lemma 1.9. If we write the last step of Algorithm 3 without Y^{-1}, the general error matrix of QR factorization is given by E_X in (2.17). A crucial aspect of the subsequent analysis is establishing connections between E_X and E_{Yi}.

Under (2.1), we provide a new interval of the shifted item s based on $\|X\|_c$ and $\|X\|_g$. If $X \in \mathbb{R}^{m\times n}$, except (1.1) and (1.2), we have the following settings.

$$4.89jn\sqrt{n}\mathbf{u} \cdot \kappa_2(X) \le 1, \tag{2.18}$$

$$11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2 \le s \le \frac{1}{100n}\|X\|_c^2. \tag{2.19}$$

Here, j is defined in (2.7). We observe that, compared to the original Shifted CholeskyQR based on $\|X\|_2$, the range of $\kappa_2(X)$ expands with a constant j related to n as indicated in (2.18). Furthermore, (2.19) 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 [15, 52, 71].

2.3.2 Algorithms

In this section, we present the improved Shifted CholeskyQR (ISCholeskyQR) and the improved Shifted CholeskyQR3 (ISCholeskyQR3). They are detailed in Algorithm 9 and Algorithm 10, respectively.

Algorithm 9: $[Q, R] = \text{ISCholeskyQR}(X)$

Input: $X \in \mathbb{R}^{m\times n}$.

Output: Orthogonal factor $Q \in \mathbb{R}^{m\times n}$, Upper triangular factor $R \in \mathbb{R}^{n\times n}$.

1: calculate $\|X\|_c$,

2: take $s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$,

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

Algorithm 10: $[Q, R] = \text{ISCholeskyQR3}(X)$

Input: $X \in \mathbb{R}^{m\times n}$.

Output: Orthogonal factor $Q \in \mathbb{R}^{m\times n}$, Upper triangular factor $R \in \mathbb{R}^{n\times n}$.

1: calculate $\|X\|_c$,

2: take $s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$,

3: $[Q, R] = \text{SCholeskyQR3}(X)$.

2.3.3 Some lemmas for proving theorems

To prove Theorem 2.1-Theorem 2.3, we require the following lemmas. These theoretical results resemble those in [21] and their proofs closely follow those of [21]. However, by utilizing the definition of the $\|\cdot\|_c$ and its properties, we can improve many upper bounds of the algorithm. We will discuss these improvements in detail below.

Lemma 2.6. *For E_A and E_B in (2.14) and (2.15), if (2.19) is satisfied, we have*

$$\|E_A\|_2 \le 1.1m\mathbf{u}\|X\|_c^2, \tag{2.20}$$

$$\|E_B\|_2 \le 1.1(n+1)\mathbf{u}\|X\|_c^2. \tag{2.21}$$

Proof. In this part, we aim to estimate $\|E_A\|_2$ and $\|E_B\|_2$ using $\|X\|_c$ instead of $\|X\|_2$. Although our analysis follows a similar approach to that in [21, 68], our new definitions of $\|\cdot\|_c$ and $\|\cdot\|_g$ allow us to provide improved analysis for error matrices.

We can estimate $\|E_A\|_F$ first since $\|E_A\|_2$ is bounded by $\|E_A\|_F$. With Lemma 1.7 and (2.14), we have

$$G = fl(X^\top X).$$

Therefore, we have

$$\begin{aligned} |E_A| &= \left|G - X^\top X\right| \\ &\leq \gamma_m \left|X^\top\right| |X| \,. \end{aligned} \tag{2.22}$$

With (2.22), for E_{Aij} which denotes the element of E_A in the i-th row and the j-th column, we have

$$\left|E_{Aij}\right| \leq \gamma_m |X_i| \left|X_j\right| . \tag{2.23}$$

Here, X_i denotes the i-th column of X. We combine (2.23) with (2.1) and have

$$\left|E_{Aij}\right| \leq \gamma_m \|X\|_g^2 . \tag{2.24}$$

Since we have

$$\|E_A\|_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}(\left|E_{Aij}\right|)^2},$$

with (2.24), we can bound $\|E_A\|_2$ as

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

Then, (2.20) is proved. (2.20) is a more accurate estimation of $\|E_A\|_2$ compared to that in [21, 68] based on Lemma 2.1.

When estimating $\|E_B\|_F$, we focus on (2.15). We use the same idea as that in [21, 68] for this estimation. With (2.1), we have

$$\begin{aligned} \|Y\|_F^2 &= \left\||Y|\right\|_F^2 \\ &\leq n\|Y\|_g^2 . \end{aligned} \tag{2.25}$$

Based on the properties of Cholesky factorization and the structure of the algorithm, we find that the square of the $\|\cdot\|_g$ of the matrix corresponds to the largest entry on the diagonal of the Gram matrix. Using Lemma 1.8, (2.14), (2.15) and (2.25), we can get

$$\begin{aligned} \|E_B\|_2 &\leq \left\||E_B|\right\|_F \\ &\leq \gamma_{n+1}\left\||Y|\right\|_F^2 \\ &\leq \gamma_{n+1} \cdot n\|Y\|_g^2 \\ &\leq \gamma_{n+1} \cdot n(\|X\|_g^2 + s + \|E_A\|_2 + \|E_B\|_2). \end{aligned} \tag{2.26}$$

With (1.1), (1.2), (2.19), (2.20) and (2.26), we can bound $\|E_B\|_2$ as

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

(2.21) is proved. Here, we take $t=\frac{s}{\|X\|_2^2}\leq 0.01$ based on (2.7) and (2.19). In all, Lemma 2.6 is proved. □

Remark 2.1. *The last step of* (2.26) *relies on Lemma 2.2 and Lemma 2.1. While the approach for estimating* $\|E_B\|_2$ *parallels that in [21, 68], we utilize the relationships between* $\|\cdot\|_2$ *and* $\|\cdot\|_g$ *established in Lemma 2.1, which derive from a distinctly different perspective on the norms of matrices compared to the existing works about CholeskyQR-type algorithms.*

Lemma 2.7. *For* Y^{-1} *and* XY^{-1} *from* (2.16)*, when* (2.19) *is satisfied, we have*

$$\left\|Y^{-1}\right\|_2 \leq \frac{1}{\sqrt{(\sigma_{min}(X))^2+0.9s}}, \tag{2.27}$$

$$\left\|XY^{-1}\right\|_2 \leq 1.5. \tag{2.28}$$

Proof. With Lemma 1.6, (2.14) and (2.15), we can get

$$(\sigma_{min}(Y))^2 \geq (\sigma_{min}(X))^2+s-\|E_A\|_2-\|E_B\|_2\,. \tag{2.29}$$

According to (2.19)-(2.21), it is easy to see that

$$\begin{aligned}\|E_A\|_2+\|E_B\|_2 &\leq 1.1(mn\mathbf{u}+n(n+1)\mathbf{u})\|X\|_g^2\\ &\leq 0.1s.\end{aligned} \tag{2.30}$$

Therefore, we put (2.30) into (2.29) and we can have

$$(\sigma_{min}(Y))^2 \geq \sigma_{min}(X)^2+0.9s. \tag{2.31}$$

With (2.31), (2.27) holds. Regarding $\left\|XY^{-1}\right\|_2$, based on (2.14), (2.15), (2.30) and (2.31), we can get

$$\begin{aligned}\left\|XY^{-1}\right\|_2 &\leq \sqrt{1+\left\|Y^{-1}\right\|_2^2(s+\|E_A\|_2+\|E_B\|_2)}\\ &\leq \sqrt{1+\frac{1.1s}{(\sigma_{min}(X))^2+0.9s}}\\ &\leq \sqrt{1+\frac{1.1}{0.9}}\\ &\leq 1.5.\end{aligned}$$

(2.28) holds. □

Lemma 2.8. *For E_{Yi} from* (2.16)*, when* (2.19) *is satisfied, we have*

$$\|E_{Yi}\|_2 \leq 1.03n\mathbf{u}\|X\|_c\,. \tag{2.32}$$

Proof. The steps to get (2.32) are similar to those in [21]. However, we can get a tighter bound of $\|E_{Yi}\|_2$ with $\|X\|_c$. For $1 \leq i \leq m$, based on Lemma 1.9 and Definition 2.1, we have

$$\begin{aligned}\|E_{Yi}\|_2 \leq & \|E_{Yi}\|_F \\ \leq & \gamma_n \cdot \|Y\|_F \\ \leq & 1.02n\sqrt{n}\mathbf{u}\|Y\|_g\,.\end{aligned} \tag{2.33}$$

With (2.14), (2.15) and (2.19), we obtain

$$\begin{aligned}\|Y\|_g^2 \leq & \|X\|_g^2 + s + (\|E_A\|_2 + \|E_B\|_2) \\ \leq & 1.011\|X\|_g^2\,.\end{aligned} \tag{2.34}$$

With (2.34), it is easy to see that

$$\|Y\|_g \leq 1.006\|X\|_g\,. \tag{2.35}$$

Therefore, we put (2.35) into (2.33) and we can get(2.32). Lemma 2.8 is proved. □

Lemma 2.9. *For E_X from* (2.17)*, when* (2.19) *is satisfied, we have*

$$\|E_X\|_2 \leq \frac{1.15n\mathbf{u}\|X\|_c^2}{\sqrt{(\sigma_{min}(X))^2 + 0.9s}}. \tag{2.36}$$

Proof. For Shifted CholeskyQR, Y will not always be invertible due to errors in numerical computations. Therefore, we estimate this by examining each row. Similar to the approach in [21], we can express (2.16) as

$$\begin{aligned}w_i^\top = & x_i^\top (Y + E_{Yi})^{-1} \\ = & x_i^\top (I + Y^{-1}E_{Yi})^{-1}Y^{-1}.\end{aligned} \tag{2.37}$$

When we define

$$(I + Y^{-1}E_i)^{-1} = I + \theta_i, \tag{2.38}$$

where

$$\theta_i := \sum_{j=1}^{\infty}(-Y^{-1}E_{Yi})^j, \tag{2.39}$$

based on (2.16) and (2.17), we can have

$$E_{Xi}^\top = x_i^\top \theta_i \tag{2.40}$$

which is the i-th row of E_X. Based on (1.2), (2.19), (2.27) and (2.32), we can bound $\left\|Y^{-1}E_{Yi}\right\|_2$ as

$$\begin{aligned}\left\|Y^{-1}E_{Yi}\right\|_2 &\leq \left\|Y^{-1}\right\|_2 \|E_{Yi}\|_2 \\ &\leq \frac{1.03n\mathbf{u}\|X\|_c}{\sqrt{(\sigma_{min}(X))^2+0.9s}} \\ &\leq \frac{1.03n\mathbf{u}\|X\|_c}{\sqrt{0.9s}} \\ &\leq \frac{1.03n\mathbf{u}\|X\|_c}{\sqrt{9.9(m\mathbf{u}+(n+1)\mathbf{u}\|X\|_c^2}} \\ &\leq \frac{1.03n\mathbf{u}\|X\|_c}{\sqrt{9.9(n+1)\mathbf{u}\|X\|_c^2}} \\ &\leq 0.35\cdot\sqrt{n\mathbf{u}} \\ &\leq 0.1.\end{aligned} \tag{2.41}$$

Putting (2.27), (2.32), (2.41) into (3.35) and we have

$$\begin{aligned}\|\theta_i\|_2 &\leq \sum_{j=1}^{\infty}(\left\|Y^{-1}\right\|_2\|E_{Yi}\|_2)^j \\ &= \frac{\left\|Y^{-1}\right\|_2\|E_{Yi}\|_2}{1-\left\|Y^{-1}\right\|_2\|E_{Yi}\|_2} \\ &\leq \frac{1}{0.9}\cdot\frac{1.03n\mathbf{u}\|X\|_c}{\sqrt{(\sigma_{min}(X))^2+0.9s}} \\ &\leq \frac{1.15n\mathbf{u}\|X\|_c}{\sqrt{(\sigma_{min}(X))^2+0.9s}}.\end{aligned} \tag{2.42}$$

We sum all the items of (2.40), together with Lemma 2.5 and (2.42), and we can have

$$\begin{aligned}\|E_X\|_2 &\leq \|E_X\|_F \\ &\leq \|X\|_F\|\theta_i\|_2 \\ &\leq \frac{1.15n\mathbf{u}\|X\|_c^2}{\sqrt{(\sigma_{min}(X))^2+0.9s}}.\end{aligned}$$

Therefore, Lemma 2.9 is proved. □

Remark 2.2. *The derivations of Lemma 2.6-Lemma 2.9 utilize the properties of $\|\cdot\|_c$. We can get sharper upper bounds compared to those in [21]. 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.*

2.3.4 Proof of Theorem 2.1

Proof. Using the previous lemmas in Section 2.3.3, we begin to estimate the orthogonality and residual of our improved Shifted CholeskyQR. The proof of Theorem 2.1 is similar to that in [21]. We aim to

demonstrate that comparable results hold, even with our enhanced bounds in the previous lemmas discussed in Section 2.3.3.

First, we consider the orthogonality. Based on (2.17), we can get

$$\begin{aligned} W^\top W &= Y^{-\top}(X+E_X)^\top(X+E_X)Y^{-1} \\ &= Y^{-\top}X^\top XY^{-1} + Y^{-\top}X^\top E_X Y^{-1} \\ &+ Y^{-\top}E_X^\top XY^{-1} + Y^{-\top}E_X^\top E_X Y^{-1} \\ &= I - Y^{-\top}(sI + E_A + E_B)Y^{-1} + (XY^{-1})^\top E_X Y^{-1} \\ &+ Y^{-\top}E_X^\top (XY^{-1}) + Y^{-\top}E_X^\top E_X Y^{-1}. \end{aligned} \tag{2.43}$$

With (2.43), we have

$$\begin{aligned} \left\|W^\top W - I\right\|_2 \leq & \left\|Y^{-1}\right\|_2^2 (\|E_A\|_2 + \|E_B\|_2 + s) + 2\left\|Y^{-1}\right\|_2 \left\|XY^{-1}\right\|_2 \|E_X\|_2 \\ & + \left\|Y^{-1}\right\|_2^2 \|E_X\|_2^2 . \end{aligned} \tag{2.44}$$

With (2.27) and (2.30), we can get

$$\begin{aligned} \left\|Y^{-1}\right\|_2^2 (\|E_A\|_2 + \|E_B\|_2 + s) &\leq \frac{1.1s}{(\sigma_{min}(X))^2 + 0.9s} \\ &\leq \frac{11}{9} \\ &\leq 1.23. \end{aligned} \tag{2.45}$$

Based on (2.19), (2.27), (2.28) and (2.36), we can obtain

$$\begin{aligned} 2\left\|Y^{-1}\right\|_2 \left\|XY^{-1}\right\|_2 \|E_X\|_2 &\leq 2 \cdot \frac{1}{\sqrt{(\sigma_{min}(X))^2 + 0.9s}} \cdot 1.5 \cdot \frac{1.15n\mathbf{u}\|X\|_c^2}{\sqrt{(\sigma_{min}(X))^2 + 0.9s}} \\ &\leq \frac{3.45n\mathbf{u}\|X\|_c^2}{(\sigma_{min}(X))^2 + 0.9s} \\ &\leq \frac{\frac{3.45}{11} \cdot s}{0.9s} \\ &\leq 0.35. \end{aligned} \tag{2.46}$$

With (2.27) and (2.36), we have

$$\begin{aligned} \left\|Y^{-1}\right\|_2^2 \|E_X\|_2^2 &\leq \frac{1}{(\sigma_{min}(X))^2 + 0.9s} \cdot \frac{(1.15n\mathbf{u}\|X\|_c^2)^2}{(\sigma_{min}(X))^2 + 0.9s} \\ &\leq \frac{(\frac{3.45}{11} \cdot s)^2}{(0.9s)^2} \\ &\leq 0.02. \end{aligned} \tag{2.47}$$

We put (2.45)-(2.47) into (2.44) and we can get

$$\begin{aligned} \left\|W^\top W - I\right\|_2 &\leq 1.23 + 0.35 + 0.02 \\ &\leq 1.6. \end{aligned}$$

Therefore, (2.8) is proved.

From (2.8), it is easy to see that

$$\|W\|_2 \leq 1.62. \tag{2.48}$$

For the residual, from (2.48), we can easily get

$$\|W\|_F \leq 1.62\sqrt{n}. \tag{2.49}$$

For $\|WY - X\|_F$, based on (2.16), we can get

$$\begin{aligned} \left\|w_i^\top Y - x_i^\top\right\|_F \leq & \left\|w_i^\top Y - w_i^\top (Y + E_{Yi})\right\|_F \\ \leq & \|w_i\|_F \|E_{Yi}\|_F . \end{aligned} \tag{2.50}$$

With (2.50), we can easily get

$$\|WY - X\|_F \leq \|W\|_F \|E_{Yi}\|_2 . \tag{2.51}$$

We put (2.32) and (2.49) into (2.51) and we can have (2.9). In all, Theorem 2.1 is proved □

Remark 2.3. *In the proof of Theorem 2.1, 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 [21]. The residual in* (2.9) *shows a tighter upper bound compared to that in [21]. More importantly,* (2.9) *can improve the condition for* $\kappa_2(X)$ *in the estimation of the singular values of* W *in the next section.*

2.3.5 Proof of Theorem 2.2

In this section, we give the proof for Theorem 2.2.

Proof. We have already estimated $\|W\|_2$. To estimate $\kappa_2(X)$, we need to estimate $\sigma_{min}(W)$. The primary steps of analysis are similar to that in [21]. When (2.17) holds, according to Lemma 1.6, we can get

$$\sigma_{min}(W) \geq \sigma_{min}(XY^{-1}) - \left\|E_X Y^{-1}\right\|_2 . \tag{2.52}$$

With (2.27) and (2.36), we can obtain

$$\begin{aligned} \left\|E_X Y^{-1}\right\|_2 \leq & \|E_X\|_2 \left\|Y^{-1}\right\|_2 \\ \leq & \frac{1.67n\sqrt{n}\mathbf{u}\|X\|_c}{(\sigma_{min}(X))^2 + 0.9s}. \end{aligned} \tag{2.53}$$

Using the similar method in [21], we have

$$\sigma_{min}(XY^{-1}) \geq \frac{\sigma_{min}(X)}{\sqrt{(\sigma_{min}(X))^2 + s}} \cdot 0.9. \tag{2.54}$$

When (2.18) holds, we put (2.53) and (2.54) into (2.52), together with $t = \frac{s}{\|X\|_2^2}$ and (2.7), we can get

$$\begin{aligned}
\sigma_{min}(Q) &\geq \frac{0.9\sigma_{min}(X)}{\sqrt{(\sigma_{min}(X))^2 + s}} - \frac{1.67n\sqrt{n}\mathbf{u}\|X\|_c}{\sqrt{(\sigma_{min}(X))^2 + 0.9s}} \\
&\geq \frac{0.9}{\sqrt{(\sigma_{min}(X))^2 + s}} \cdot (\sigma_{min}(X) - \frac{1.67}{0.9 \cdot \sqrt{0.9}} \cdot jn\sqrt{n}\mathbf{u}\|X\|_2) \\
&\geq \frac{\sigma_{min}(X)}{2\sqrt{(\sigma_{min}(X))^2 + s}} \\
&= \frac{1}{2\sqrt{1 + t(\kappa_2(X))^2}}.
\end{aligned} \tag{2.55}$$

Based on (2.48) and (2.55), we have

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

Therefore, we can get (2.10).

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 [21]. To obtain the sufficient condition of $\kappa_2(X)$ without encountering the numerical breakdown, based on (1.4) in [68], we let

$$\begin{aligned}
\kappa_2(W) &\leq 3.24\sqrt{1 + t(\kappa_2(X))^2} \\
&\leq \frac{1}{8\sqrt{mn\mathbf{u} + n(n+1)\mathbf{u}}}.
\end{aligned} \tag{2.56}$$

When $s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$, we can have

$$t = \frac{s}{\|X\|_2^2} = 11j^2(m\mathbf{u} + (n+1)\mathbf{u}). \tag{2.57}$$

With (2.57), if $\kappa_2(X)$ is large enough, *e.g.*, $\kappa_2(X) \geq \mathbf{u}^{-\frac{1}{2}}$, we can get

$$t(\kappa_2(X))^2 \geq 11(m+n) >> 1.$$

So it is easy to see that

$$1 + t(\kappa_2(X))^2 \approx t(\kappa_2(X))^2.$$

Therefore, using (2.56), we can conclude that

$$\kappa_2(X) \leq \frac{1}{25.92\sqrt{t} \cdot \sqrt{mn\mathbf{u} + n(n+1)\mathbf{u}}}. \tag{2.58}$$

We put $t = 11p^2(mn\mathbf{u} + n(n+1)\mathbf{u})$ into (2.58) and we can obtain (2.11). Therefore, Theorem 2.2 is proved. □

Remark 2.4. *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 Table 1.2 and Table 1.3, highlighting these advantages, which are further illustrated in Section 2.4.*

2.3.6 Proof of Theorem 2.3

In this part, we prove Theorem 2.3 with some results in Theorem 2.1.

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

$$
\begin{aligned}
C - W^\top W &= E_1, \\
D^\top D - C &= E_2, \\
VD - W &= E_3, && (2.59)\\
DY - N &= E_4, && (2.60)\\
B - V^\top V &= E_5, \\
J^\top J - B &= E_6, \\
QJ - V &= E_7, && (2.61)\\
JN - R &= E_8. && (2.62)
\end{aligned}
$$

Here, the calculation of R in Algorithm 12 is divided into two steps, that is, (2.60) and (2.62). Similar to that of [68], Z in Algorithm 10 satisfies

$$Z = JD, \tag{2.63}$$

without error matrices. With (2.63), R should be written as

$$R = JDY, \tag{2.64}$$

if we do not consider rounding errors. In order to simplify rounding error analysis of (2.64), we write the multiplication of D and Y with error matrices as (2.60) and the multiplication of J and N can be written as (2.62).

Similar to the proof of Theorem 2.1, we consider the orthogonality first. For our improved Shifted CholeskyQR3, similar to that in [68], when Shifted CholeskyQR3 is applicable, we can get

$$\kappa_2(W) \leq \frac{1}{8\sqrt{mn\mathbf{u} + n(n+1)\mathbf{u}}}, \tag{2.65}$$

$$\kappa_2(V) \leq 1.1. \tag{2.66}$$

Therefore, we can obtain (2.12).

When considering the residual, based on (2.59)-(2.62), we have

$$\begin{aligned} QR &= (V+E_7)J^{-1}(JN-E_8) \\ &= (V+E_7)N-(V+E_7)J^{-1}E_8 \\ &= VN+E_7N-QE_8 \\ &= (W+E_3)D^{-1}(DY-E_4)+E_7N-QE_8 \\ &= (W+E_3)Y-(W+E_3)D^{-1}E_4+E_7N-QE_8 \\ &= WY+E_3Y-VE_4+E_7N-QE_8. \end{aligned} \tag{2.67}$$

Therefore, with (2.67), it is obvious that

$$\begin{aligned} \|QR-X\|_F \leq &\|WY-X\|_F+\|E_3\|_F\|Y\|_2+\|V\|_2\|E_4\|_F \\ &+\|E_7\|_F\|N\|_2+\|Q\|_2\|E_8\|_F. \end{aligned} \tag{2.68}$$

Similar to (2.16), we express (2.59) in each row as

$$v_i^\top = w_i^\top (D+E_{Di})^{-1},$$

where $v_i^\top$ and $w_i^\top$ denote the i-th rows of V and W. Following the methodologies outlined in [21, 68] and the concepts presented in this chapter, we have

$$\|Y\|_2 \leq 1.006\|X\|_2, \tag{2.69}$$

$$\begin{aligned} \|E_{Di}\|_2 &\leq 1.2n\sqrt{n}\mathbf{u}\cdot\|W\|_2 \\ &\leq 2.079n\sqrt{n}\mathbf{u}, \end{aligned} \tag{2.70}$$

$$\|V\|_2 \leq 1.039, \tag{2.71}$$

$$\begin{aligned} \|D\|_2 &\leq 1.1\|W\|_2 \\ &\leq 1.906. \end{aligned} \tag{2.72}$$

We combine (2.69)-(2.72) with Lemma 1.7, Lemma 2.3, (2.7), (2.35) and similar steps in [21], we can bound $\|E_3\|_F, \|E_4\|_F$ and $\|E_4\|_g$ in (2.59) and (2.60) as

$$\begin{aligned} \|E_3\|_F &\leq \|V\|_F\cdot\|E_{Di}\|_2 \\ &\leq 1.039\cdot\sqrt{n}\cdot 2.079n\sqrt{n}\mathbf{u} \\ &\leq 2.16n^2\mathbf{u}, \end{aligned} \tag{2.73}$$

$$\begin{aligned} \|E_4\|_F &\leq \gamma_n(\|D\|_F\cdot\|Y\|_F) \\ &\leq \gamma_n(\sqrt{n}\cdot\|D\|_2\cdot\sqrt{n}\cdot\|Y\|_g) \\ &\leq 1.1n\sqrt{n}\mathbf{u}\cdot 1.906\cdot 1.006\|X\|_c \\ &\leq 2.11jn\sqrt{n}\mathbf{u}\|X\|_2, \end{aligned} \tag{2.74}$$

$$\begin{aligned}\|E_4\|_g &\le \gamma_n(\|D\|_F \cdot \|Y\|_g) \\ &\le \gamma_n(\sqrt{n}\|D\|_2 \cdot \|Y\|_g) \\ &\le 1.1n\mathbf{u} \cdot 1.906 \cdot 1.006\|X\|_c \\ &\le 2.11jn\mathbf{u}\|X\|_2 .\end{aligned} \tag{2.75}$$

Moreover, based on Lemma 2.3, Lemma 2.2, (2.35), (2.69), (2.74) and (2.75), $\|N\|_2$ and $\|N\|_g$ in (2.60) can be bounded as

$$\begin{aligned}\|N\|_2 &\le \|D\|_2\|Y\|_2 + \|E_4\|_2 \\ &\le 1.906 \cdot 1.006\|X\|_2 + 2.11jn\sqrt{n}\mathbf{u}\|X\|_2 \\ &\le 1.95\|X\|_2 ,\end{aligned} \tag{2.76}$$

$$\begin{aligned}\|N\|_g &\le \|D\|_2\|Y\|_g + \|E_4\|_g \\ &\le 1.906 \cdot \frac{1.006j}{\sqrt{n}} \cdot \|X\|_2 + 2.11jn\mathbf{u}\|X\|_2 \\ &\le \frac{1.95j}{\sqrt{n}} \cdot \|X\|_2 .\end{aligned} \tag{2.77}$$

Similar to (2.16), we write (2.61) in each row as

$$q_i^\top = v_i^\top (J + E_{Ji})^{-1},$$

where $q_i^\top$ and $v_i^\top$ represent the i-th rows of Q and V. Similar to (2.70)-(2.72) and with (1.1), (1.2) and (2.12), we can get

$$\|Q\|_2 \le 1.1, \tag{2.78}$$

$$\begin{aligned}\|E_{Ji}\|_2 &\le 1.2n\sqrt{n}\|V\|_2 \\ &\le 1.2n\sqrt{n}\mathbf{u} \cdot 1.039 \\ &\le 1.246n\sqrt{n}\mathbf{u},\end{aligned} \tag{2.79}$$

$$\begin{aligned}\|J\|_2 &\le 1.1\|V\|_2 \\ &\le 1.143.\end{aligned} \tag{2.80}$$

With Lemma 1.7 and (2.77)-(2.80), we can bound $\|E_7\|_F$ and $\|E_8\|_F$ in (2.61) and (2.62) as

$$\begin{aligned}\|E_7\|_F &\le \|Q\|_F \cdot \|E_{Ji}\|_2 , \\ &\le 1.1\sqrt{n} \cdot 1.246n\sqrt{n}\mathbf{u} \\ &\le 1.38n^2\mathbf{u},\end{aligned} \tag{2.81}$$

$$\begin{aligned}\|E_8\|_F &\le \gamma_n(\|J\|_F \cdot \|N\|_F) \\ &\le \gamma_n(\sqrt{n} \cdot \|J\|_2 \cdot \sqrt{n} \cdot \|N\|_g) \\ &\le 1.1n\sqrt{n}\mathbf{u} \cdot 1.143 \cdot 1.95j\|X\|_2 \\ &\le 2.46jn\sqrt{n}\mathbf{u}\|X\|_2 .\end{aligned} \tag{2.82}$$

Table 2.1: 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

Therefore, we put (2.9), (2.69), (2.71), (2.73), (2.74), (2.76), (2.78), (2.81) and (2.82) into (2.68) and we can get (2.13). In all, Theorem 2.3 is proved. □

Remark 2.5. *Based on* (2.13)*, we find that we obtain a sharper upper bound of the residual of the algorithm compared to that in [21], utilizing properties of* $\|\cdot\|_c$ *and* $\|\cdot\|_g$*. This represents a theoretical advancement in rounding error analysis. The steps leading to* (2.77) *highlight the effectiveness of Lemma 2.3 and Lemma 2.1. Although the second inequality of* (2.3) *appears weaker than the first inequality of* (2.3)*, it cannot be dismissed in estimating* $\|\cdot\|_g$ *of the error matrix in terms of its absolute value. This lays a solid foundation for* (2.77) *and* (2.82)*, marking advancements in estimation methods for problems related to matrix multiplications.*

2.4 Numerical experiments

In this section, 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 $\left\|Q^\top Q - I\right\|_F$ and residual $\|QR - X\|_F$ 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 l_1-value, defined as $l_1 = \frac{j}{\sqrt{n}}$ for $X \in \mathbb{R}^{m\times n}$, to illustrate the extent of improvement brought by our reduced s compared to the original method in [21]. As a comparison group, we also evaluate the properties of HouseholderQR, which is considered one of the most stable numerical algorithms, to demonstrate the effectiveness and advantages of our improved Shifted CholeskyQR3. The specifications of our computer used for these experiments are provided in Table 2.1. We assess the performance of our method in multi-core CPU environments.

2.4.1 Numerical examples

In this part, we introduce the numerical examples, specifically the test matrix X utilized in this chapter. The primary test matrix $X \in \mathbb{R}^{m\times n}$ is similar to that used in [21, 68] 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.

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 [21, 68]. We control $\kappa_2(X)$ through $\sigma_{min}(X)$. Specifically, we set

$$X = O\Sigma H^T.$$

Here, $O \in \mathbb{R}^{m\times m}, H \in \mathbb{R}^{n\times n}$ 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 $\kappa_2(X) = \frac{1}{\sigma}$.

In our numerical experiments, we will focus on some large matrices. We construct large matrices in a block version. We can construct some small $X_1 \in \mathbb{R}^{n\times n}$ based on SVD and build $X \in \mathbb{R}^{m\times n}$ as

$$X = \begin{pmatrix} X_1 \\ X_1 \\ \vdots \\ X_1 \end{pmatrix}.$$

The Hilbert matrix

The Hilbert matrix is a well-known ill-conditioned square matrix. It is widely used in many applications, including numerical approximation theory and solving linear systems, seeing [6, 9, 34] and the references therein. For a Hilbert matrix T, as n increases, $\kappa_2(T)$ also increases. The element of Hilbert matrix X is shown as below:

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

For $X \in \mathbb{R}^{m\times n}$, we take $m = 10n$. We form X through

$$X = \begin{pmatrix} T \\ T \\ \vdots \\ T \end{pmatrix}.$$

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, seeing [7, 45, 48, 49, 60] 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 chapter, we take two vectors, $e_1 = (1, 0, 0\cdots 0, 0)^\top \in \mathbb{R}^n$ and $g = (1, 1, 1\cdots 1, 1)^\top \in \mathbb{R}^n$. We define a diagonal matrix $M = \mathrm{diag}(y) \in \mathbb{R}^{n\times n}$, where $y = (y_1, y_2, \cdots, y_{n-1}, y_n)$ and $y_i = \begin{cases} 0, & \text{if } i = 1 \\ 10, & \text{if } i = 2, \cdots, n-1 \\ y, & \text{if } i = n \end{cases}$. We build an arrowhead matrix P through

$$P = 30e_1 \cdot g^\top + M.$$

Similar to the previous section, we take $m = 5n$ and construct X through

$$X = \begin{pmatrix} P \\ P \\ \vdots \\ P \end{pmatrix}.$$

We vary v to modify $\kappa_2(X)$.

2.4.2 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 with $s = 11(m\mathbf{u}+(n+1)\mathbf{u})\|X\|_c^2$ as '*Improved*', while the original Shifted CholeskyQR3 with $s = 11(mn\mathbf{u}+(n+1)n\mathbf{u})\|X\|_2^2$ 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 Table 2.2 and Table 2.3. We also carry out numerical experiments for a large $X \in \mathbb{R}^{16384\times 1024}$. We construct a small $X_1 \in \mathbb{R}^{1024\times 1024}$ based on SVD with $\|X_1\|_2 = 1$

and $\kappa_2(X) = \frac{1}{\sigma}$. X is build with 16 X_1 from the up to the bottom. We vary σ from 10^{-6}, 10^{-8}, 10^{-10}, 10^{-12} to 2×10^{-13}. The numerical results are listed in Table 2.4 and Table 2.5. Table 2.2 and Table 2.3 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 (1.10) and (2.11). We have similar results for large matrices according to Table 2.4 and Table 2.5. When m and n get increasing, the computational range of Shifted CholeskyQR3 will decrease, which corresponds to the theoretical results. In our real example based on the Hilbert matrix, we vary n from 9, 10, 11 to 12 and $\kappa_2(X)$ is also varying. In the example based on the arrowhead matrix, we take $n = 64$ and vary y from 10^{-11}, 10^{-12}, 10^{-13} to 10^{-14} to modify $\kappa_2(X)$. The numerical results are shown in Table 2.6-Table 2.9. 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 Table 2.10- 2.13. 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.

Table 2.2: Orthogonality of 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 2.3: Residual of 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 2.4: Orthogonality of the algorithms with $\kappa_2(X)$ varying when $m = 16384$ and $n = 1024$

$\kappa_2(X)$	$1.00e+6$	$1.00e+8$	$1.00e+10$	$1.00e+12$	$5.00e+12$
Improved	$1.73e-14$	$1.90e-14$	$1.99e-14$	$2.10e-14$	$2.05e-14$
Original	$1.88e-14$	$1.97e-14$	$2.04e-14$	$2.10e-14$	-

Table 2.5: Residual of the algorithms with $\kappa_2(X)$ varying when $m = 16384$ and $n = 1024$

$\kappa_2(X)$	$1.00e+6$	$1.00e+8$	$1.00e+10$	$1.00e+12$	$5.00e+12$
Improved	$2.23e-14$	$2.02e-14$	$1.86e-14$	$1.74e-14$	$1.70e-14$
Original	$2.23e-14$	$2.02e-14$	$1.87e-14$	$1.75e-14$	-

Table 2.6: Orthogonality of the algorithm for the Hilbert matrix with different n

n	9	10	11	12
Original	$1.43e-15$	$1.59e-15$	$1.64e-15$	–
Improved	$9.29e-16$	$9.59e-16$	$1.90e-15$	$1.96e-12$

Table 2.7: Residual of the algorithm for the Hilbert matrix with different n

n	9	10	11	12
Original	$9.45e-16$	$1.16e-15$	$6.69e-16$	–
Improved	$8.15e-16$	$1.05e-15$	$5.78e-16$	$1.15e-15$

Table 2.8: Orthogonality of the algorithm for the arrowhead matrix when $n = 64$

$\kappa_2(X)$	$3.40e+13$	$3.40e+14$	$3.40e+15$	$3.24e+16$
Original	$1.11e-15$	$1.11e-15$	$1.13e-15$	–
Improved	$1.75e-15$	$1.80e-15$	$1.80e-15$	$1.80e-15$

Table 2.9: Residual of the algorithm for the arrowhead matrix when $n = 64$

$\kappa_2(X)$	$3.40e+13$	$3.40e+14$	$3.40e+15$	$3.24e+16$
Original	$1.49e-13$	$1.49e-13$	$1.49e-13$	–
Improved	$7.08e-14$	$7.08e-14$	$7.08e-14$	$7.08e-14$

Table 2.10: Orthogonality of 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 2.11: Residual of 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 2.12: Orthogonality of 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 2.13: Residual of 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	$1.27e-15$	$1.76e-15$	$2.55e-15$	$3.62e-15$	$5.00e-15$

2.4.3 Comparison between the theoretical bounds and real performances

In this part, we make a comparison between the theoretical bounds of Shifted CholeskyQR3 and its real performances. In the beginning, we test the accuracy. For the input $X \in \mathbb{R}^{m\times n}$ based on SVD, we fix $\|X\|_2 = 1$ and $\kappa_2(X) = 10^{12}$. We denote $6(mn\mathbf{u} + n(n+1)\mathbf{u})$ in (2.12) as the '*Theoretical bound*' in orthogonality. Moreover, $(6.57 \cdot \frac{j}{\sqrt{n}} + 4.87)n^2\mathbf{u}\|X\|_2$ in (2.13) is the '*Theoretical bound*' in residual. To test the influence of m, we fix $n = 64$ and vary m. To test the influence of n, we fix $m = 2048$ and vary n. Comparisons of orthogonality and residual with different m and n are shown in Table 2.14-Table 2.17. Regarding the conditions of $\kappa_2(X)$, we denote $\frac{1}{86j(m\sqrt{n}\mathbf{u}+(n+1)\sqrt{n}\mathbf{u})}$ in Table 1.2 as the '*Sufficient condition*' of $\kappa_2(X)$ and $\frac{1}{4.89jn\sqrt{n}\mathbf{u}}$ as the '*Upper bound*' of $\kappa_2(X)$. We vary m and n and comparisons of conditions of $\kappa_2(X)$ are shown in Table 2.18 and Table 2.19.

Table 2.14: Comparison of orthogonality with the improved s when $\kappa_2(X) = 10^{12}$ and $n = 64$

m	128	256	512	1024	2048
Real error	$3.29e-15$	$3.66e-15$	$2.64e-15$	$2.28e-15$	$1.89e-15$
Theoretical bound	$8.23e-12$	$1.37e-11$	$2.46e-11$	$4.64e-11$	$9.01e-11$

Table 2.15: Comparison of orthogonality with the improved s when $\kappa_2(X) = 10^{12}$ and $m = 2048$

n	64	128	256	512	1024
Real error	$1.89e-15$	$2.99e-15$	$5.08e-15$	$9.27e-15$	$1.74e-14$
Theoretical bound	$9.01e-11$	$1.86e-10$	$3.93e-10$	$8.73e-10$	$2.10e-09$

Table 2.16: Comparison of residual with the improved s when $\kappa_2(X) = 10^{12}$ and $n = 64$

m	128	256	512	1024	2048
Real error	$5.90e-16$	$5.97e-16$	$5.56e-16$	$5.76e-16$	$5.67e-16$
Theoretical bound	$2.89e-12$	$2.89e-12$	$2.89e-12$	$2.89e-12$	$2.89e-12$

Table 2.17: Comparison of residual with the improved s when $\kappa_2(X) = 10^{12}$ and $m = 2048$

n	64	128	256	512	1024
Real error	$5.66e-16$	$1.07e-15$	$2.00e-15$	$3.08e-15$	$4.35e-15$
Theoretical bound	$2.89e-12$	$1.16e-11$	$4.64e-11$	$1.82e-10$	$7.21e-10$

Table 2.18: Comparison of $\kappa_2(X)$ with the improved s when $\kappa_2(X) = 10^{12}$ and $n = 128$

m	256	512	1024	2048	4096
Real case	$\geq 10^{12}$	$\geq 10^{12}$	$\geq 10^{12}$	$\geq 10^{12}$	$\geq 10^{12}$
Upper bound	$4.68e+11$	$4.68e+11$	$4.68e+11$	$4.68e+11$	$4.68e+11$
Sufficient condition	$8.85e+09$	$5.31e+09$	$2.95e+09$	$1.56e+09$	$8.06e+08$

Table 2.19: Comparison of $\kappa_2(X)$ with the improved s when $\kappa_2(X) = 10^{12}$ and $m = 4096$

n	128	256	512	1024	2048
Real case	$\geq 10^{12}$	$\geq 10^{12}$	$\geq 10^{12}$	$\geq 10^{12}$	$\geq 10^{12}$
Upper bound	$4.68e+11$	$1.39e+11$	$3.81e+10$	$9.23e+09$	$2.62e+09$
Sufficient condition	$8.06e+08$	$4.66e+08$	$2.41e+08$	$1.05e+08$	$4.96e+07$

According to Table 2.14-Table 2.19, we can find that the theoretical results of $\kappa_2(X)$ and accuracy, including orthogonality and residual, are worse than the real results after computation on the laptop. It shows that the deterministic models for rounding error analysis have the problem of overestimation, which has distance from the real cases.

2.4.4 $\kappa_2(Q)$ 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 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 2.20. From Table 2.20, we can see that $\kappa_2(X)$ 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 (1.9) and (2.10).

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 Table 2.21 and Table 2.22. 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 Table 2.20-Table 2.22, 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 chapter 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.

Table 2.20: $\kappa_2(Q)$ 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 2.21: $\kappa_2(Q)$ 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	$3.88e+06$	$5.01e+06$	$6.72e+06$	$9.23e+06$	$1.29e+07$

Table 2.22: $\kappa_2(Q)$ 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	$1.29e+07$	$1.84e+07$	$2.68e+07$	$4.00e+07$	$6.20e+07$

2.4.5 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.

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 Table 2.23 and Table 2.24. 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. In fact, the computational costs of HouseholderQR and CholeskyQR are all in the level of mn^2 for the input matrix $X \in \mathbb{R}^{m\times n}$. HouseholderQR is not so efficient in implementation because it primarily uses BLAS2 routines, while CholeskyQR uses

BLAS3 due to its structure. 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.

Table 2.23: 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 2.24: 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	$1.12e+01$	$2.59e+01$	$5.66e+01$	$1.16e+02$	$3.11e+02$

2.4.6 The improvement of s

Here, we aim to show the l_1-values in this chapter by using some examples since $l_1 = \frac{\|X\|_g}{\|X\|_2} = \frac{j}{\sqrt{n}} = \sqrt{\frac{11(m\mathbf{u}+(n+1)\mathbf{u})\|X\|_c^2}{11(mn\mathbf{u}+n(n+1)\mathbf{u})\|X\|_2^2}}$ for $X \in \mathbb{R}^{m\times n}$. Therefore, the l_1-value reflects how much the shifted item s is reduced according to our definition of $\|X\|_c$. In the future, we will investigate how to estimate l_1 in different cases.

In the beginning, we test the l_1-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 Table 2.25 and Table 2.26. Moreover, we test the l_1-value with varying m and n for the Hilbert matrix $X \in \mathbb{R}^{m\times n}$, where $m = 10n$. The experimental results are listed in Table 2.27. The numerical results indicate that l_1 is relatively small compared to 1. Notably, n significantly influences l_1 more than m. With n increasing, l_1 decreases markedly, which aligns with the theoretical lower bound of the l_1-value. This observation suggests that our improved s is likely more effective for relatively large matrices.

Table 2.25: l_1 with m varying when $\kappa_2(X) = 10^{12}$ and $n = 64$ for $X \in \mathbb{R}^{m\times n}$ based on SVD

m	128	256	512	1024	2048
l_1	0.2824	0.2762	0.2386	0.2453	0.2498

Table 2.26: l_1 with n varying when $\kappa_2(X) = 10^{12}$ and $m = 2048$ for $X \in \mathbb{R}^{m\times n}$ based on SVD

n	64	128	256	512	1024
l_1	0.2498	0.2396	0.2127	0.2024	0.1726

Table 2.27: l_1 with n varying for the Hilbert matrix $X \in \mathbb{R}^{m\times n}$ with $m = 10n$

n	9	10	11	12
l_1	0.7190	0.7106	0.7033	0.6968

2.5 Conclusions

This chapter focuses on the improvement of Shifted Cholesky3. We define a new $\|X\|_c$ for the input matrix X and construct a new shifted item s based on $\|X\|_c$ for Shifted CholeskyQR3. We prove theoretically that this s can improve the applicability of Shifted CholeskyQR3 while maintaining numerical stability. Numerical experiments verify our findings and show that our improved Shifted CholeskyQR3 with $\|X\|_c$ is as efficient as the original Shifted CholeskyQR3.

CHAPTER 3.
SHIFTED CHOLESKYQR FOR SPARSE MATRICES

In this chapter, we focus on Shifted CholeskyQR for sparse matrices. We provide a new model for sparse matrices and divide sparse matrices into two types, T_1 matrices and T_2 matrices, based on the presence of dense columns. We introduce an alternative choice of the shifted item s based on the structure and the key element of the input $X \in \mathbb{R}^{m\times n}$. We prove that such an s is superior to that mentioned in Chapter 2 for T_1 matrices with the certain element-conditions(ENCs) since it improves the applicability of the algorithm. Shifted CholeskyQR3 is also numerical stable with this s in these cases. Numerical experiments demonstrate the effectiveness of such an alternative choice of s in improving the applicability and maintaining numerical stability for T_1 matrices. For T_2 matrices, Shifted CholeskyQR3 exhibits new properties compared to dense cases. Furthermore, our alternative choice of s remains as efficient as it is with the improved s from Chapter 2. This chapter is organized as follows. Our contributions and primary theoretical results are outlined in Section 3.1. In Section 3.2, we conduct a theoretical analysis of Shifted CholeskyQR3 for sparse matrices and prove Theorems 3.3 through 3.4, which were proposed in Section 3.1. This analysis constitutes a key part of this chapter. Following the theoretical analysis, we perform numerical experiments using typical examples from real-world problems, and we present the results in Section 3.3. Section 3.4 shows the conclusions of this chapter.

3.1 Our contributions and theoretical results

In this part, we introduce our new model for sparse matrices along with its corresponding divisions. With these new concepts and general settings, we present several theoretical results related to Shifted CholeskyQR3 for sparse matrices.

3.1.1 Our new divisions of sparse matrices

In the beginning, we introduce a new model of sparse matrices based on column sparsity and provide the definitions of T_1 and T_2 matrices in Definition 3.1.

Definition 3.1. *A sparse matrix $X \in \mathbb{R}^{m\times n}$ has v dense columns, $0 \leq v << n$, with each dense column containing at most t_1 non-zero elements, where t_1 is relatively close to m. For the remaining sparse columns, each column has at most t_2 non-zero elements, where $0 < t_2 << t_1$. When $v > 0$, we refer to such a sparse matrix X as a T_1 matrix. When $v = 0$, we call X a T_2 matrix. Moreover, we*

define

$$c = \max\left|x_{ij}\right|, 1 \leq i \leq m, 1 \leq j \leq n.$$

as the element with the largest absolute value in X*.*

3.1.2 General settings and Shifted CholeskyQR3 for sparse matrices

When $X \in \mathbb{R}^{m\times n}$ is a sparse matrix which follows Definition 3.1, except (1.1) and (1.2), we give some settings below.

$$j_s \leq s \leq j_b, \tag{3.1}$$

$$\kappa_2(X) \leq F. \tag{3.2}$$

In (3.1) and (3.2), we take

$$j_s = \min(11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2, 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2),$$

$$j_b = \begin{cases} \phi, & \text{if } j_s = 11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2 \\ \frac{1}{100n}\|X\|_c^2, & \text{if } j_s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2 \end{cases},$$

$$F = \begin{cases} \frac{1}{4n^2\mathbf{u}hl}, & \text{if } j_s = 11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2 \\ \frac{1}{4.89jn\sqrt{n}\mathbf{u}}, & \text{if } j_s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2 \end{cases}.$$

Here, j is defined in (2.7) and we let

$$\begin{aligned} \phi &= \min(\frac{1}{100n} \cdot (vt_1 + nt_2)c^2, \frac{1}{100}t_1c^2), \\ l &= \frac{c\sqrt{t_1}}{\|X\|_2}, \\ r &= \frac{n\sqrt{n}}{m\sqrt{v}}, \\ h &= \sqrt{2.23 + 0.34r + 0.013r^2}. \end{aligned}$$

c, v, t_1 and t_2 are defined in Definition 3.1.

In the general settings described above, we utilize the definition of $\|\cdot\|_g$ from Chapter 2, which is presented in Definition 2.2. (3.4) and (3.1) are similar to those in [21] for the original Shifted CholeskyQR3. (3.2) outlines the requirements for $\kappa_2(X)$ in Shifted CholeskyQR3. Shifted CholeskyQR and Shifted CholeskyQR3 for sparse matrices are detailed in Algorithm 11 and Algorithm 12, with $s = j_s$ as specified in (3.1). This demonstrates that an alternative s can be utilized in Shifted CholeskyQR3 for sparse cases, which is a key innovative aspect of this chapter.

Algorithm 11: $[Q, R] = \text{SCholeskyQR}(X)$ for sparse matrices

Input: $X \in \mathbb{R}^{m\times n}$.

Output: Orthogonal factor $Q \in \mathbb{R}^{m\times n}$, Upper triangular factor $R \in \mathbb{R}^{n\times n}$.

1: get c, v, t_1, t_2 as defined in Definition 3.1 for the input X,

2: take $s = j_s$ as defined in (4.45),

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

Algorithm 12: $[Q, R] = \text{SCholeskyQR3}(X)$ for sparse matrices

Input: $X \in \mathbb{R}^{m\times n}$.

Output: Orthogonal factor $Q \in \mathbb{R}^{m\times n}$, Upper triangular factor $R \in \mathbb{R}^{n\times n}$.

1: get c, v, t_1, t_2 as defined in Definition 3.1 for the input X,

2: take $s = j_s$ as defined in (4.45),

3: $[Q, R] = \text{SCholeskyQR3}(X)$.

3.1.3 Theoretical results of T_1 matrices

For T_1 matrices, we have already provided detailed analysis under (2.18) and (2.19) in Chapter 2. In this chapter, we primarily focus on the case when

$$11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2 \le s \le \phi, \tag{3.3}$$

$$4n^2\mathbf{u} \cdot hl\kappa_2(X) \le 1, \tag{3.4}$$

where

$$\phi = \min(\frac{1}{100n} \cdot (vt_1 + nt_2)c^2, \frac{1}{100}t_1c^2).$$

In the following, we show the properties of Shifted CholeskyQR3 for T_1 matrices in Theorem 3.1-Theorem 3.3 under (3.3) and (3.4).

Theorem 3.1. *If $X \in \mathbb{R}^{m\times n}$ is a T_1 matrix and $[W, Y] = SCholeskyQR(X)$, when (3.3) and (3.4) are satisfied, we have*

$$\kappa_2(Q) \le 2h \cdot \sqrt{1 + \alpha_0(\kappa_2(X))^2}, \tag{3.5}$$

if $\alpha_0 = \frac{s}{\|X\|_2^2} = 11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot k$ and $k = \frac{(vt_1+nt_2)c^2}{\|X\|_2^2}$. For $[Q, R] = SCholeskyQR3(X)$ with $s = 11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2$, if $\kappa_2(X)$ is large enough, the sufficient condition of $\kappa_2(X)$ is

$$\kappa_2(X) \le \frac{1}{16\sqrt{11nk} \cdot (m\mathbf{u} + (n+1)\mathbf{u})h}. \tag{3.6}$$

Here, h is utilized and defined in (3.2).

Theorem 3.2. *Under* (3.6)*, if* $X \in \mathbb{R}^{m\times n}$ *is a* T_1 *matrix and* $[Q,R] = SCholeskyQR3(X)$*, when* $s = 11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2$*, we have*

$$\left\|Q^\top Q - I\right\|_F \leq 6(mn\mathbf{u} + n(n+1)\mathbf{u}), \tag{3.7}$$

$$\|QR - X\|_F \leq (2.79 + 3.97l)hn^2\mathbf{u}\|X\|_2 . \tag{3.8}$$

Here, l *is utilized and defined in* (3.2)*.*

In Theorem 3.3, when we take

$$s = 11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2, \tag{3.9}$$

we provide a corresponding element-norm condition (ENC) under which s in (3.9) is optimal, which differs significantly from s in [21] and Chapter 2. The ENC is not unique, and we present a typical example in the following theoretical results.

Theorem 3.3. *If* T_1 *matrix* $X \in \mathbb{R}^{m\times n}$ *is a* T_1 *matrix and* $[Q,R] = SCholeskyQR3(X)$*, if* X *satisfies the ENC:* $c = \sqrt{\frac{\beta}{m}} \cdot \|X\|_2$ *and* $\beta \leq \frac{mj^2}{vt_1+nt_2}$*, then*

$$j_s = 11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2. \tag{3.10}$$

Here, j *and* j_s *are utilized and defined in* (2.7) *and* (3.1)*. Therefore, the sufficient condition of* $\kappa_2(X)$ *is*

$$\kappa_2(X) \leq \frac{1}{16\sqrt{11n\epsilon} \cdot (m\mathbf{u} + (n+1)\mathbf{u})h}, \tag{3.11}$$

when we define $\epsilon = \frac{\beta(vt_1+nt_2)}{m}$*.*

Remark 3.1. *Theorem 3.1 is one of the most important results of this chapter. It demonstrates that when* X *is a* T_1 *matrix,* s *in* (3.9) *can be taken for Shifted CholeskyQR3. Theorem 3.2 shows that such an* s *maintains numerical stability. These two theorems indicate that we can leverage the structure of the sparse* X *to construct a new shifted item* s*, which is superior to that in Chapter 2 with proper ENCs, such as the one mentioned in Theorem 3.3. With the ENC in Theorem 3.3,* (3.8) *is equivalent to*

$$\|QR - X\|_F \leq (2.79 + 3.97\beta)hn^2\mathbf{u}\|X\|_2 .$$

This shows that, given a suitable ENC and when (3.10) *is satisfied, Shifted CholeskyQR3 is numerically stable with respect to the residual.*

3.1.4 Theoretical results of T_2 matrices

When X is a T_2 matrix under Definition 3.1, the following theorem holds.

Theorem 3.4. *If $X \in \mathbb{R}^{m\times n}$ is a T_2 matrix and $[Q, R] = SCholeskyQR3(X)$, we have*

$$j_s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2 . \tag{3.12}$$

When $s = j_s$, the sufficient condition of $\kappa_2(X)$ and rounding error analysis of Shifted CholeskyQR3 for T_2 matrices follow Theorem 2.3 in Chapter 2.

Remark 3.2. *In the real practice, we can easily obtain c using MATLAB, and determining t_1 and t_2 requires only a few lines of code. Since we have already defined $\|X\|_g$ in Chapter 2, we can conduct theoretical analysis based on the structure of X. In many real-world applications, there are many T_1 matrices with relatively dense columns. The presence of such dense columns can greatly influence $\|X\|_2$, especially when the absolute values of the elements of X are very close to each other. It is acceptable to estimate t_1 and t_2 roughly, as this will not affect the primary results when m is sufficiently large.*

3.2 Proof of Theorem 3.1-Theorem 3.4

In this section, we prove Theorem 3.1-Theorem 3.4 under the assumption that X is sparse, based on Definition 3.1. Among all the theorems, Theorem 3.1 and Theorem 3.2 are the key results.

3.2.1 Lemmas to prove Theorem 3.1-Theorem 3.3 matrices

Before proving Theorem 3.1-Theorem 3.3, we write Shifted CholeskyQR with error matrices below.

$$G = X^\top X + E_A, \tag{3.13}$$

$$Y^\top Y = G + sI + E_B, \tag{3.14}$$

$$w_i^\top = x_i^\top (Y + \Delta Y_i)^{-1}, \tag{3.15}$$

$$WY = X + \Delta X. \tag{3.16}$$

Here, $x_i^\top$ and $w_i^\top$ represent the i-th rows of X and W, respectively. The definitions of E_A in (3.13), E_B in (3.14), ΔY_i in (3.15) and ΔX in (3.16) are the same as those defined in Chapter 2.

To prove these theorems, we first need to establish some lemmas. When $11(m\mathbf{u}+(n+1)\mathbf{u})\|X\|_c^2 \le s \le \frac{1}{100n}\|X\|_c^2$, we have conducted rounding error analysis in Chapter 2. Therefore, we primarily focus on the case when $11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2 \le s \le \phi$, $\phi = \min(\frac{1}{100n} \cdot (vt_1 + nt_2)c^2, \frac{1}{100}t_1c^2)$ and $v > 0$. The general ideas of the theoretical analysis are similar to those in [21] and Chapter 2. However, we integrate the model of sparsity from Definition 3.1 with rounding error analysis, providing different theoretical results compared to existing works.

Lemma 3.1. *For* $\|E_A\|_2$ *and* $\|E_B\|_2$ *in* (3.13) *and* (3.14)*, when* (3.3) *is satisfied, we have*

$$\|E_A\|_2 \leq 1.1m\mathbf{u} \cdot (vt_1 + nt_2)c^2, \tag{3.17}$$

$$\|E_B\|_2 \leq 1.1(n+1)\mathbf{u} \cdot (vt_1 + nt_2)c^2. \tag{3.18}$$

Proof. According to Definition 3.1, it X is a T_1 matrix, it has v dense columns with at most t_1 non-zero elements and sparse columns with at most t_2 non-zero elements, when estimating the ij-th element of E_A, with Lemma 1.7, we can have

$$\begin{aligned} |E_A|_{ij1} &\leq \gamma_m |x_i| \left|x_j\right| \\ &\leq \gamma_m \cdot t_1 \cdot \|x_i\|_2 \left\|x_j\right\|_2 \\ &\leq \gamma_m \cdot t_1 c^2, \end{aligned} \tag{3.19}$$

if both x_i and x_j are dense columns. x_i is the i-th column of X. There are v^2 elements of E_A can be estimated in this way. When at least one of x_i and x_j is sparse, we can have

$$\begin{aligned} |E_A|_{ij2} &\leq \gamma_m |x_i| \left|x_j\right| \\ &\leq \gamma_m \cdot t_2 \cdot \|x_i\|_2 \left\|x_j\right\|_2 \\ &\leq \gamma_m \cdot t_2 c^2. \end{aligned} \tag{3.20}$$

There are $2v(n-v)+(n-v)^2$ elements of E_A can be estimated in this way. Therefore, based on (3.19) and (3.20), we can estimate $\|E_A\|_2$ as

$$\begin{aligned} \|E_A\|_2 &\leq \left\| |E_A| \right\|_F \\ &\leq \sqrt{v^2 \cdot [\gamma_m \cdot t_1 c^2]^2 + (2v(n-v) + (n-v)^2) \cdot [\gamma_m \cdot t_2 c^2]^2} \\ &\leq 1.1m\mathbf{u} \cdot (vt_1 + nt_2)c^2. \end{aligned}$$

(3.17) is proved.

For $\|E_B\|_2$, Lemma 2.2, (3.13) and (3.14), we can get

$$\begin{aligned} \|E_B\|_2 &\leq \left\| |E_B| \right\|_F \\ &\leq \gamma_{n+1} \|Y\|_F^2 . \end{aligned} \tag{3.21}$$

In fact, we have

$$\|Y\|_F^2 = \operatorname{tr}(Y^\top Y), \tag{3.22}$$

which denotes the trace of the gram matrix $Y^\top Y$. With Definition 3.1, (3.13), (3.14) and (3.22), we can get

$$\begin{aligned} \gamma_{n+1}\|Y\|_F^2 &\leq \gamma_{n+1} \operatorname{tr}(Y^\top Y) \\ &\leq \gamma_{n+1} \operatorname{tr}(X^\top X + sI + E_A + E_B) \\ &\leq \gamma_{n+1}(\|X\|_F^2 + sn + n\|E_A\|_2 + n\|E_B\|_2) \\ &\leq \gamma_{n+1}((vt_1 + nt_2)c^2 + sn + n\|E_A\|_F + n\|E_B\|_F). \end{aligned} \tag{3.23}$$

If we set

$$z = \frac{s}{(vt_1 + nt_2)c^2},$$

with (3.1), we can have

$$11(m\mathbf{u} + (n+1)\mathbf{u}) \leq z \leq \frac{1}{100n}. \tag{3.24}$$

With (3.24), we combine (3.21) and (3.23) with (1.1), (1.2), (3.3) and (3.17), and we can bound $\|E_B\|_2$ as

$$\begin{aligned}
\|E_B\|_2 &\leq \frac{\gamma_{n+1} \cdot (1 + 1.1mn\mathbf{u} + zn)}{1 - \gamma_{n+1} \cdot n} \cdot (vt_1 + nt_2)c^2 \\
&\leq \frac{1.02(n+1)\mathbf{u} \cdot (1 + 1.1mn\mathbf{u} + 0.01)}{1 - 1.02(n+1)\mathbf{u} \cdot n} \cdot (vt_1 + nt_2)c^2 \\
&\leq \frac{1.02(n+1)\mathbf{u} \cdot (1 + 1.1 \cdot \frac{1}{64} + 0.01)}{1 - \frac{1.02}{64}} \cdot (vt_1 + nt_2)c^2 \\
&\leq 1.1(n+1)\mathbf{u} \cdot (vt_1 + nt_2)c^2.
\end{aligned}$$

(3.18) is proved. □

Remark 3.3. *The steps to prove* (3.18) *contains a step utilizing the properties of traces in* (3.23)*. This idea of proof has not occurred in the works of CholeskyQR before. Although* (3.18) *seems similar to the corresponding results in [21, 68] and Chapter 2, our ideas in the theoretical analysis are distinguished from those in the previous works, which is an innovative point of this chapter.*

Lemma 3.2. *For* $\left\|Y^{-1}\right\|_2$ *and* $\left\|XY^{-1}\right\|_2$ *in* (3.15)*, when* (3.3) *is satisfied, we have*

$$\left\|Y^{-1}\right\|_2 \leq \frac{1}{\sqrt{(\sigma_{min}(X))^2 + 0.9s}}, \tag{3.25}$$

$$\left\|XY^{-1}\right\|_2 \leq 1.5. \tag{3.26}$$

Proof. The steps to prove (3.25) and (3.26) are the same as those in [21] and Chapter 2. □

Lemma 3.3. *For* $\|\Delta Y_i\|_2$ *in* (3.15)*, when* (3.3) *is satisfied, we have*

$$\|\Delta Y_i\|_2 \leq 1.03n\sqrt{n}\mathbf{u} \cdot c\sqrt{t_1}. \tag{3.27}$$

Proof. For (3.15), based on Lemma 1.9, we can have

$$\begin{aligned}
\|\Delta Y_i\|_2 &\leq \gamma_n \cdot \left\|\|Y|\right\|_F \\
&\leq \gamma_n \cdot \sqrt{n}\|Y\|_g \\
&\leq 1.02n\sqrt{n}\mathbf{u} \cdot \|Y\|_g .
\end{aligned} \tag{3.28}$$

With (3.3), (3.17) and (3.18), we can have

$$\begin{aligned}
\|E_A\|_2 + \|E_B\|_2 &\leq 1.1 \cdot (m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2) \cdot c^2 \\
&\leq 0.1s.
\end{aligned} \tag{3.29}$$

For $\|Y\|_g$, similar to the steps in Chapter 2 and based on (3.3), (3.13), (3.14) and (3.29), we can have

$$\begin{aligned} \|Y\|_g^2 \leq & \|X\|_g^2 + s + (\|E_A\|_2 + \|E_B\|_2) \\ \leq & 1.011 t_1 c^2. \end{aligned} \tag{3.30}$$

Therefore, with (3.30), it is easy to see that

$$\|Y\|_g \leq 1.006 c\sqrt{t_1}. \tag{3.31}$$

We put (3.31) into (3.28) and we can have (3.27). □

Lemma 3.4. *For* $\|\Delta X\|_2$ *in* (3.16)*, when* (3.3) *is satisfied, we have*

$$\|\Delta X\|_2 \leq \frac{1.09n\sqrt{n}\mathbf{u} \cdot \sqrt{t_1} \cdot \sqrt{(vt_1 + nt_2)} \cdot c^2}{\sqrt{(\sigma_{min}(X))^2 + 0.9s}}. \tag{3.32}$$

Proof. Similar to the approach in [21] and Chapter 2, we can express (3.15) as

$$\begin{aligned} w_i^\top &= x_i^\top (Y + \Delta Y_i)^{-1} \\ &= x_i^\top (I + Y^{-1}\Delta Y_i)^{-1} Y^{-1}. \end{aligned} \tag{3.33}$$

When we define

$$(I + Y^{-1}\Delta Y_i)^{-1} = I + \theta_i, \tag{3.34}$$

where

$$\theta_i := \sum_{j=1}^{\infty} (-Y^{-1}\Delta Y_i)^j, \tag{3.35}$$

based on (3.15) and (3.16), we can have

$$\Delta {x_i}^\top = x_i^\top \theta_i. \tag{3.36}$$

Δx_i is the i-th row of ΔX. Based on (1.2), (3.3), (3.25) and (3.27), when (3.3) is satisfied and v is a small positive integer, we can have

$$\begin{aligned} \left\|Y^{-1}\Delta Y_i\right\|_2 \leq & \left\|Y^{-1}\right\|_2 \|\Delta Y_i\|_2 \\ \leq & \frac{1.03n\sqrt{n}\mathbf{u}\|X\|_g}{\sqrt{(\sigma_{min}(X))^2 + 0.9s}} \\ \leq & \frac{1.03n\sqrt{n}\mathbf{u} \cdot c\sqrt{t_1}}{\sqrt{0.9s}} \\ \leq & \frac{1.03n\sqrt{n}\mathbf{u} \cdot c\sqrt{t_1}}{\sqrt{9.9(m\mathbf{u} + (n+1)\mathbf{u})} \cdot \sqrt{(vt_1 + nt_2)c^2}} \\ \leq & \frac{1.03}{\sqrt{9.9}} \cdot n\sqrt{\mathbf{u}} \cdot \frac{1}{\sqrt{v}} \\ \leq & 0.05. \end{aligned} \tag{3.37}$$

For (3.35), with (3.25), (3.27) and (3.37), we can have

$$\begin{aligned}\|\theta_i\|_2 &\leq \sum_{j=1}^{\infty}(\left\|Y^{-1}\right\|_2\|\Delta Y_i\|_2)^j \\ &= \frac{\left\|Y^{-1}\right\|_2\|\Delta Y_i\|_2}{1-\left\|Y^{-1}\right\|_2\|\Delta Y_i\|_2} \\ &\leq \frac{1}{0.95}\cdot\frac{1.03n\sqrt{n}\mathbf{u}\cdot c\sqrt{t_1}}{\sqrt{(\sigma_{min}(X))^2+0.9s}} \\ &\leq \frac{1.09n\sqrt{n}\mathbf{u}\cdot c\sqrt{t_1}}{\sqrt{(\sigma_{min}(X))^2+0.9s}}.\end{aligned} \tag{3.38}$$

Based on (3.36), it is easy to see that

$$\left\|\Delta x_i^\top\right\|_2 \leq \left\|x_i^\top\right\|_2\|\theta_i\|_2\,. \tag{3.39}$$

According to Definition 3.1, when X is a T_1 matrix, we have

$$\|X\|_F \leq \sqrt{vt_1+nt_2}\cdot c. \tag{3.40}$$

Therefore, similar to the step in [21], with (3.39), we can have

$$\begin{aligned}\|\Delta X\|_2 &\leq \|\Delta X\|_F \\ &\leq \|X\|_F\|\theta_i\|_2\,.\end{aligned} \tag{3.41}$$

We put (3.39) and (3.40) into (3.41) and we can have (3.32). □

3.2.2 Proof of Theorem 3.1

Here, we prove Theorem 3.1 with Lemma 3.2-Lemma 3.4.

Proof. The general approach to proving Theorem 3.1 is similar to those in [21] and Chapter 2. However, we establish connections between the structure of X and QR factorization. Our proof will be divided into three parts: estimating $\left\|W^\top W-I\right\|_F$, estimating $\|\Delta X\|_F$, and analyzing the relationship between $\kappa_2(X)$ and $\kappa_2(W)$.

Estimating $\left\|W^\top W-I\right\|_2$

With (3.13)-(3.16), we can have

$$\begin{aligned}W^\top W &= Y^{-\top}(X+\Delta X)^\top(X+\Delta X)Y^{-1} \\ &= Y^{-\top}X^\top XY^{-1}+Y^{-\top}X^\top\Delta XY^{-1} \\ &+ Y^{-\top}\Delta X^\top XY^{-1}+Y^{-\top}\Delta X^\top\Delta XY^{-1} \\ &= I-Y^{-\top}(sI+E_1+E_2)Y^{-1}+(XY^{-1})^\top\Delta XY^{-1} \\ &+ Y^{-\top}\Delta X^\top(XY^{-1})+Y^{-\top}\Delta X^\top\Delta XY^{-1}.\end{aligned}$$

Therefore, we can have

$$\begin{aligned}\left\|W^\top W - I\right\|_2 \leq &\left\|Y^{-1}\right\|_2^2 (\|E_A\|_2 + \|E_B\|_2 + s) + 2\left\|Y^{-1}\right\|_2 \left\|XY^{-1}\right\|_2 \|\Delta X\|_2 \\ &+ \left\|Y^{-1}\right\|_2^2 \|\Delta X\|_2^2 .\end{aligned} \tag{3.42}$$

According to (3.25) and (3.29), we can have

$$\begin{aligned}\left\|Y^{-1}\right\|_2^2 (\|E_A\|_2 + \|E_B\|_2 + s) &\leq \frac{1.1s}{(\sigma_{min}(X))^2 + 0.9s} \\ &\leq 1.23.\end{aligned} \tag{3.43}$$

Based on (3.25), (3.26) and (3.32), when v is a small positive integer, we can have

$$\begin{aligned}2\left\|Y^{-1}\right\|_2 \left\|XY^{-1}\right\|_2 \|\Delta X\|_2 &\leq 2 \cdot \frac{1}{\sqrt{(\sigma_{min}(X))^2 + 0.9s}} \cdot 1.5 \cdot \frac{1.09n\sqrt{n}\mathbf{u} \cdot \sqrt{t_1} \cdot \sqrt{(vt_1 + nt_2)} \cdot c^2}{\sqrt{(\sigma_{min}(X))^2 + 0.9s}} \\ &\leq 3.27 \cdot \frac{n\sqrt{n}\mathbf{u} \cdot \sqrt{t_1} \cdot \sqrt{(vt_1 + nt_2)} \cdot c^2}{(\sigma_{min}(X))^2 + 0.9s} \\ &\leq 3.27 \cdot \frac{n\sqrt{n}\mathbf{u} \cdot \sqrt{t_1} \cdot \sqrt{(vt_1 + nt_2)} \cdot c^2}{9.9(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2} \\ &\leq 0.34 \cdot \frac{n\sqrt{t_1 n}}{\sqrt{vt_1 + nt_2} \cdot (m + (n+1))} \\ &\leq 0.34 \cdot \frac{n\sqrt{n}}{m\sqrt{v}}.\end{aligned} \tag{3.44}$$

With (3.25) and (3.32), if v is a small positive integer, we can have

$$\begin{aligned}\left\|Y^{-1}\right\|_2^2 \|\Delta X\|_2^2 &\leq \frac{1}{(\sigma_{min}(X))^2 + 0.9s} \cdot \frac{(1.09n\sqrt{n}\mathbf{u} \cdot \sqrt{t_1} \cdot \sqrt{(vt_1 + nt_2)} \cdot c^2)^2}{(\sigma_{min}(X))^2 + 0.9s} \\ &\leq \frac{(1.09n\sqrt{n}\mathbf{u} \cdot \sqrt{t_1} \cdot \sqrt{(vt_1 + nt_2)} \cdot c^2)^2}{[9.9(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2]^2} \\ &\leq 0.013 \cdot \frac{n^3 t_1}{(vt_1 + nt_2)[m + (n+1)]^2} \\ &\leq 0.013 \cdot \frac{n^3}{m^2 v}.\end{aligned} \tag{3.45}$$

Therefore, we put (3.43)-(3.45) into (3.42) and we can have

$$\left\|W^\top W - I\right\|_2 \leq 1.23 + 0.34r + 0.013r^2, \tag{3.46}$$

where $r = \frac{n\sqrt{n}}{m\sqrt{v}}$. With (3.46), we can have

$$\|W\|_2 \leq h, \tag{3.47}$$

if $h = \sqrt{2.3 + 0.37r + 0.015r^2}$. From (3.47), we can see that $\|W\|_2$ is influenced by the size of X and the number of dense columns when X is a T_1 matrix. When $X \in \mathbb{R}^{m \times n}$ is very tall and skinny, *e.g.*, $m \geq n\sqrt{n}$, $\|W\|_2$ can be bounded by a small constant since v is a small positive number. Moreover, when $m \geq n$ and v is a small positive integer, we have $r < \sqrt{n}$. Therefore, it is easy to see that h can be bounded by $\sqrt{3n}$, which is very meaningful in estimating the residual of Shifted CholeskyQR3 in the following.

Estimating $\|\Delta X\|_F$

Regarding $\|\Delta X\|_F$ in (3.16), similar to the results in [21] and Chapter 2, when $l = \frac{c\sqrt{t_1}}{\|X\|_2}$, based on (3.27) and (3.47), we can have

$$\begin{aligned}\|\Delta X\|_F &= \|QR - X\|_F \\ &\leq \|Q\|_F \cdot \|\Delta Y_i\|_2 \\ &\leq h\sqrt{n} \cdot 1.03n\sqrt{n}\mathbf{u} \cdot c\sqrt{t_1} \\ &\leq 1.03n^2\mathbf{u} \cdot hc\sqrt{t_1} \\ &= 1.03hln^2\mathbf{u}\|X\|_2 .\end{aligned} \tag{3.48}$$

This is an upper bound based on the settings of T_1 matrices.

The relationship between $\kappa_2(X)$ and $\kappa_2(W)$

In order to estimate $\kappa_2(W)$, since we have already estimated $\|W\|_2$, we only need to estimate $\sigma_{min}(W)$. Based on Lemma 1.6, we can have

$$\sigma_{min}(W) \geq \sigma_{min}(XY^{-1}) - \left\|\Delta XY^{-1}\right\|_2 . \tag{3.49}$$

Based on (3.25) and (3.48), we can have

$$\begin{aligned}\left\|\Delta XY^{-1}\right\|_2 &\leq \|\Delta X\|_2 \left\|Y^{-1}\right\|_2 \\ &\leq \frac{1.03n^2\mathbf{u} \cdot hc\sqrt{t_1}}{\sqrt{(\sigma_{min}(X))^2 + 0.9s}}.\end{aligned} \tag{3.50}$$

Based on the result in [21], we can have

$$\sigma_{min}(XY^{-1}) \geq \frac{\sigma_{min}(X)}{\sqrt{(\sigma_{min}(X))^2 + s}} \cdot 0.9. \tag{3.51}$$

Therefore, we put (3.50) and (3.51) into (3.49) and based on (3.4), we can have

$$\begin{aligned}\sigma_{min}(W) &\geq \frac{0.9\sigma_{min}(X)}{\sqrt{(\sigma_{min}(X))^2 + s}} - \frac{1.1n^2\mathbf{u} \cdot hl\|X\|_2}{\sqrt{(\sigma_{min}(X))^2 + 0.9s}} \\ &\geq \frac{0.9}{\sqrt{(\sigma_{min}(X))^2 + s}}(\sigma_{min}(X) - \frac{1.1}{0.9 \cdot \sqrt{0.9}} \cdot n^2\mathbf{u} \cdot hl\|X\|_2) \\ &\geq \frac{\sigma_{min}(X)}{2\sqrt{(\sigma_{min}(X))^2 + s}} \\ &= \frac{1}{2\sqrt{1 + \alpha_0(\kappa_2(X))^2}},\end{aligned} \tag{3.52}$$

where $\alpha_0 = \frac{s}{\|X\|_2^2} = 11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot k$, $k = \frac{(vt_1 + nt_2)c^2}{\|X\|_2^2}$. Based on (3.47) and (3.52), we can have

$$\kappa_2(W) \leq 2h \cdot \sqrt{1 + \alpha_0(\kappa_2(X))^2}.$$

Here, (3.5) is proved.

With (3.40), we can have

$$k = \frac{(vt_1 + nt_2)c^2}{\|X\|_2^2} \geq 1. \tag{3.53}$$

When (3.9) is satisfied, similar to the steps in Chapter 2, when $\kappa_2(X)$ is large, it is easy to see that $\alpha_0(\kappa_2(X))^2 \geq mk >> 1$ with (3.53). Therefore, we can get

$$2h \cdot \sqrt{1 + \alpha_0(\kappa_2(W))^2} \approx 2h \cdot \sqrt{\alpha_0} \cdot \kappa_2(X).$$

With (3.5), we can have

$$\kappa_2(W) \leq 2h \cdot \sqrt{\alpha_0} \cdot \kappa_2(X).$$

Using the similar method as that in [21] and Chapter 2, in order to receive a sufficient condition for Shifted CholeskyQR3, we only need to have

$$\begin{aligned} \kappa_2(W) &\leq 2h \cdot \sqrt{\alpha_0} \cdot \kappa_2(X) \\ &\leq \frac{1}{8(mn\mathbf{u} + n(n+1)\mathbf{u})}. \end{aligned} \tag{3.54}$$

We put $\alpha_0 = \frac{s}{\|X\|_2^2} = 11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot k$ into (3.54) and we can have (3.6). □

3.2.3 Proof of Theorem 3.2

In this section, we prove Theorem 3.2 based on Theorem 3.1 and the properties of $\|\cdot\|_g$. Our approach to prove Theorem 3.2 is inspired by that in Chapter 2.

Proof. When $s = 11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2$, $\kappa_2(X)$ satisfies (3.6). We can easily derive (3.7) with $\kappa_2(X)$, which is similar to that in [68].

For the residual of Shifted CholeskyQR3, $\|QR - X\|_F$, we express the CholeskyQR2 after Shifted CholeskyQR with the error matrices as follows.

$$\begin{aligned} C - W^\top W &= E_1, \\ D^\top D - C &= E_2, \\ VD - W &= E_3, && (3.55) \\ DY - N &= E_4. && (3.56) \\ B - V^\top V &= E_5, \\ J^\top J - B &= E_6, \\ QJ - V &= E_7, && (3.57) \\ JN - R &= E_8. && (3.58) \end{aligned}$$

The same as that in Chapter 2, we divide the last step of calculating R in Algorithm 10 into (3.56) and (3.58). Based on (3.55)-(3.58), we can have

$$\begin{aligned} QR &= (V + E_7)J^{-1}(JN - E_8) \\ &= (V + E_7)N - (V + E_7)J^{-1}E_8 \\ &= VN + E_7N - QE_8 \\ &= (W + E_3)D^{-1}(DY - E_4) + E_7N - QE_8 \\ &= (W + E_3)Y - (W + E_3)D^{-1}E_4 + E_7N - QE_8 \\ &= WY + E_3Y - VE_4 + E_7N - QE_8. \end{aligned} \tag{3.59}$$

Therefore, based on (3.59), we can get

$$\begin{aligned} \|QR - X\|_F \leq &\|WY - X\|_F + \|E_3\|_F\|Y\|_2 + \|V\|_2\|E_4\|_F \\ &+ \|E_7\|_F\|N\|_2 + \|Q\|_2\|E_8\|_F . \end{aligned} \tag{3.60}$$

Similar to (3.15), we rewrite (3.55) through rows as

$$v_i^\top = w_i^\top (D + \Delta D_i)^{-1},$$

where $v_i^\top$ and $w_i^\top$ represent the i-th rows of V and W. Based on the results in [21, 68] and (3.47), we can have

$$\begin{aligned} \|\Delta D_i\|_2 &\leq 1.03n\sqrt{n}\mathbf{u} \cdot \|W\|_2 \\ &\leq 1.03hn\sqrt{n}\mathbf{u}, \end{aligned} \tag{3.61}$$

$$\|Y\|_2 \leq 1.006\|X\|_2 , \tag{3.62}$$

$$\|V\|_2 \leq \frac{\sqrt{69}}{8}, \tag{3.63}$$

$$\begin{aligned} \|D\|_2 &\leq 1.02\|W\|_2 \\ &\leq 1.02h. \end{aligned} \tag{3.64}$$

With Lemma 1.7, Lemma 2.3, (3.31) and (3.61)-(3.64), we can bound $\|E_3\|_F, \|E_4\|_F$ and $\|E_4\|_g$ as

$$\begin{aligned} \|E_3\|_F &\leq \|V\|_F \cdot \|\Delta D_i\|_2 \\ &\leq \frac{\sqrt{69n}}{8} \cdot 1.03hn\sqrt{n}\mathbf{u} \\ &\leq 1.07hn^2\mathbf{u}, \end{aligned} \tag{3.65}$$

$$\begin{aligned} \|E_4\|_F &\leq \gamma_n(\|D\|_F \cdot \|Y\|_F) \\ &\leq \gamma_n(\sqrt{n}\|D\|_2 \cdot \sqrt{n}\|Y\|_g) \\ &\leq 1.02n^2\mathbf{u} \cdot 1.02h \cdot 1.006c\sqrt{t_1} \\ &\leq 1.05hln^2\mathbf{u}\|X\|_2 , \end{aligned} \tag{3.66}$$

$$\begin{aligned}\|E_4\|_g &\leq \gamma_n(\|D\|_F \cdot \|Y\|_g) \\ &\leq \gamma_n(\sqrt{n}\|D\|_2 \cdot \|Y\|_g) \\ &\leq 1.02n\sqrt{n}\mathbf{u} \cdot 1.02h \cdot 1.006c\sqrt{t_1} \\ &\leq 1.05hln\sqrt{n}\mathbf{u}\|X\|_2 .\end{aligned} \tag{3.67}$$

Moreover, when $l = \frac{c\sqrt{t_1}}{\|X\|_2}$, based on Lemma 2.2, Lemma 2.1, (3.31), (3.62), (3.64) and (3.67), $\|N\|_2$ and $\|N\|_g$ can be bounded as

$$\begin{aligned}\|N\|_2 &\leq \|D\|_2\|Y\|_2 + \|E_4\|_2 \\ &\leq 1.02h \cdot 1.006\|X\|_2 + 1.05hln^2\mathbf{u}\|X\|_2 \\ &= (1.03h + 0.02hl)\|X\|_2 ,\end{aligned} \tag{3.68}$$

$$\begin{aligned}\|N\|_g &\leq \|D\|_2\|Y\|_g + \|E_4\|_g \\ &\leq 1.02h \cdot 1.006c\sqrt{t_1} + 1.05hln\sqrt{n}\mathbf{u}\|X\|_2 \\ &\leq 1.05hl\|X\|_2 .\end{aligned} \tag{3.69}$$

If we rewrite (3.57) through rows as

$$q_i^\top = v_i^\top (J + \Delta J_i)^{-1},$$

where $q_i^\top$ and $v_i^\top$ represent the i-th rows of Q and V, based on the results in [21, 68], we can have

$$\begin{aligned}\|\Delta J_i\|_2 &\leq 1.03n\sqrt{n}\|V\|_2 \\ &\leq 1.03n\sqrt{n}\mathbf{u} \cdot \frac{\sqrt{69}}{8} \\ &\leq 1.07n\sqrt{n}\mathbf{u},\end{aligned} \tag{3.70}$$

$$\|Q\|_2 \leq 1.1, \tag{3.71}$$

$$\begin{aligned}\|J\|_2 &\leq 1.02\|V\|_2 \\ &\leq 1.06.\end{aligned} \tag{3.72}$$

With Lemma 1.7 and (3.69)-(3.72), we can bound $\|E_7\|_F$ and $\|E_8\|_F$ as

$$\begin{aligned}\|E_7\|_F &\leq \|Q\|_F \cdot \|\Delta J_i\|_2 \\ &\leq 1.1\sqrt{n} \cdot 1.07n\sqrt{n}\mathbf{u} \\ &\leq 1.18n^2\mathbf{u},\end{aligned} \tag{3.73}$$

$$\begin{aligned}\|E_8\|_F &\leq \gamma_n(\|J\|_F \cdot \|N\|_F) \\ &\leq \gamma_n(\sqrt{n}\|J\|_2 \cdot \sqrt{n}\|N\|_g) \\ &\leq 1.02n^2\mathbf{u} \cdot 1.06 \cdot 1.05hc\sqrt{t_1} \\ &\leq 1.14hln^2\mathbf{u}\|X\|_2 .\end{aligned} \tag{3.74}$$

Therefore, we put (3.48), (3.62), (3.63), (3.65), (3.66), (3.68), (3.71), (3.73) and (3.74) into (3.60) and we can have (3.8). Theorem 3.2 is proved. □

3.2.4 Proof of Theorem 3.3

In this part, we prove Theorem 3.3 based on the proper ENC provided.

Proof. When we have the ENC: $c = \sqrt{\frac{\beta}{m}} \cdot \|X\|_2$ and $\beta \leq \frac{mj^2}{vt_1+nt_2}$, we just need to put the ENC into j_s as defined in (3.1) and we can have (3.10). When $s = 11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2$ and $c = \sqrt{\frac{\beta}{m}} \cdot \|X\|_2$, α_0 in Theorem 3.1 satisfies $\alpha_0 = 11(m\mathbf{u}+(n+1)\mathbf{u}) \cdot \epsilon$, where $\epsilon = \frac{\beta(vt_1+nt_2)}{m}$. Therefore, we only to replace k in (3.6) with ϵ and we can receive (3.11). Therefore, Theorem 3.3 is proved. □

3.2.5 Proof of Theorem 3.4

After proving Theorem 3.1-Theorem 3.3 matrices, we prove the special case of Definition 3.1 when X is a T_2 matrix.

Proof. According to Definition 3.1, if the input $X \in \mathbb{R}^{m\times n}$ is a T_2 matrix, then $v = 0$. In (3.1), j_s becomes $\min(11(mn\mathbf{u} + n(n+1)\mathbf{u}) \cdot t_2c^2, 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2)$. From (2.1) and Definition 3.1, we have $t_2c^2 \geq \|X\|_g^2$. Therefore, we can derive (3.12). When (3.12) is satisfied, it is Theorem 2.3 in Chapter 2. □

Remark 3.4. *Among all the lemmas used to prove Theorem 3.1-Theorem 3.4, Lemma 3.1 is one of the most crucial. We build connections between the model of sparse matrices in Definition 3.1 and the estimation of* $\|E_A\|_2$ *and* $\|E_B\|_2$. *Our alternative s is based on* (3.17) *and* (3.18). *The proof of Lemma 3.1 lays a solid foundation for the subsequent analysis.* (3.30) *demonstrates the advantage of* $\|X\|_g$ *over* $\|X\|_2$ *for sparse matrices. For the sparse* X, *estimating* $\|X\|_2$ *through the element and structure of* X *is challenging. The traditional* $\|\cdot\|_2$ *is not the best to reflect the properties of the sparse matrix. We often need to estimate* $\|X\|_F$ *to replace* $\|X\|_2$, *which will influences the required ENCs and error bounds. In fact,* $\|\cdot\|_g$ *plays a significant role in rounding error analysis for sparse matrices, particularly in the steps of proving Theorem 3.2 to get tighter error bounds of residual. Although we do not calculate* $\|X\|_g$ *directly in the proof of these theorems, its connection to the structure and the element of* X *greatly simplifies our analysis, leveraging the relationship between the columns of the input* X *and CholeskyQR-type algorithms, as also mentioned in Chapter 2.*

3.3 Numerical experiments

In this section, we conduct numerical experiments to examine the properties of Shifted CholeskyQR3 for sparse matrices. We primarily focus on the applicability, numerical stability, and CPU time(s) of Shifted CholeskyQR3 with our alternative s. The experiments are performed on our own laptop using MATLAB R2022a, and the specifications of the computer are listed in Table 2.1.

3.3.1 T_1 matrices

In real applications, T_1 matrices are very common in graph theory, control theory, and certain eigenvalue problems, see [7, 45, 49] and their references. One of the most well-known T_1 matrices is the arrowhead matrix, which features a dense column and a dense row. In this section, we focus on the arrowhead matrix and conduct numerical experiments. Two different types of numerical examples are shown below.

A medium-size X in the block version

For the medium-size X in the block version, we take $m = 2048$ and $n = 64$. We build a group of orthogonal basis in R^{64} in the form of $e_1 = (1, 0, 0 \cdots 0, 0)^\top$, $e_2 = (0, 1, 0 \cdots 0, 0)^\top$, $\cdots$, $e_{64} = (0, 0, 0 \cdots 0, 1)^\top$. We take a vector $f \in \mathbb{R}^{64}$, which satisfies $f = (f_1, f_2, \cdots, f_{n-1}, f_n)^\top$ and $f_i = \begin{cases} 0, & \text{if } i = 1 \\ 1, & \text{if } i = 2, 3, \cdots, 64 \end{cases}$. We define a diagonal matrix $P = \text{diag}(u) \in \mathbb{R}^{64\times 64}$, where $u = (u_1, u_2, \cdots, u_{63}, u_{64})$ and $u_i = \begin{cases} 3, & \text{if } i = 1, 2, \cdots, 32 \\ 3 \cdot (\frac{a}{3})^{\frac{i-33}{31}}, & \text{if } i = 33, 34, \cdots, 64 \end{cases}$. Here, a is a small positive constant. We form $K \in \mathbb{R}^{64\times 64}$ as

$$K = -5e_1 \cdot f^\top - 10f \cdot e_1^\top + P.$$

We build $X \in \mathbb{R}^{2048\times 64}$ with 32 K as

$$X = \begin{pmatrix} K \\ K \\ \vdots \\ K \end{pmatrix}.$$

As a comparison group, we construct a common dense matrix U using the same method described in [21, 68] and Chapter 2. U is constructed using Singular Value Decomposition (SVD), and we control $\kappa_2(U)$ through $\sigma_{min}(U)$. We set

$$U = O\Sigma H^\top.$$

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

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

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

In the beginning, we test the applicability and accuracy of Shifted CholeskyQR3 between different s for such a medium-size T_1 matrix X in the block version. Our X satisfies the ENC in Theorem 3.3 with $c = 10$, $v = 1$, $t_1 = 2048$ and $t_2 = 64$. We choose $s = j_s = \min(11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2, 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2)$ based on (3.1). Here, $j_s = 11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2$ with the ENC. We vary a from 3×10^{-6}, 3×10^{-8}, 3×10^{-10}, 3×10^{-12} to 3×10^{-14} to adjust $\kappa_2(X)$. The σ of U is also varying to ensure $\kappa_2(U) \approx \kappa_2(X)$. For U, we use $s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$ in Chapter 2. We test the applicability and accuracy of Shifted CholeskyQR3 with different s in the cases of X and U. All results are listed in Table 3.1–Table 3.3. We refer to our alternative $s = j_s = \min(11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2, 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2)$ as '*the alternative s*' and $s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$ as '*the improved s*'.

Table 3.1: Shifted CholeskyQR3 with the alternative s for the medium-size X

$\kappa_2(X)$	$2.18e+07$	$1.99e+09$	$1.81e+11$	$1.63e+13$	$1.46e+15$
Orthogonality	$2.92e-15$	$3.52e-15$	$4.43e-15$	$3.80e-15$	$3.84e-15$
Residual	$1.08e-13$	$1.07e-13$	$1.00e-13$	$1.16e-13$	$8.83e-14$

Table 3.2: Shifted CholeskyQR3 with the improved s for the medium-size X

$\kappa_2(X)$	$2.18e+07$	$1.99e+09$	$1.81e+11$	$1.63e+13$	$1.46e+15$
Orthogonality	$3.02e-15$	$3.60e-15$	$5.67e-15$	$4.08e-15$	–
Residual	$1.10e-13$	$1.09e-13$	$1.00e-13$	$1.04e-13$	–

Table 3.3: Shifted CholeskyQR3 with the improved s for the medium-size U

$\kappa_2(X)$	$2.18e+07$	$1.99e+09$	$1.81e+11$	$1.63e+13$	$1.46e+15$
Orthogonality	$1.96e-15$	$1.83e-15$	$2.13e-15$	$1.86e-15$	–
Residual	$6.95e-16$	$6.47e-16$	$6.10e-16$	$5.69e-16$	–

According to Table 3.1 and Table 3.2, we find that Shifted CholeskyQR3 with our alternative s can handle more ill-conditioned T_1 matrices than with the improved s in Chapter 2 in this medium-size case, demonstrating the improvement of our new s for T_1 matrices in terms of applicability with

appropriate ENCs. When $\kappa_2(X) \geq 10^{14}$, our alternative s remains applicable, while the improved s does not. The comparison between Table 3.1 and Table 3.3 highlights the effectiveness of designing a different choice of s for sparse cases, which corresponds to the comparison in Table 1.4. Furthermore, Shifted CholeskyQR3 maintains a similar level of numerical stability in this case with our alternative s compared to both the case with the improved s and the case of dense matrices, as indicated by the comparison of orthogonality and residuals in Table 3.1–Table 3.3. This aligns with the theoretical results presented in Table 1.5.

In addition to testing applicability and numerical stability, we also evaluate the CPU time(s) of Shifted CholeskyQR3 with different s in this case with respect to X in our numerical experiments. The corresponding results of CPU times for the various s values are listed in Table 3.4.

Table 3.4: Comparison of CPU time(s) with different s for the medium-size X

$\kappa_2(X)$	$2.18e+07$	$1.99e+09$	$1.81e+11$	$1.63e+13$	$1.46e+15$
The alternative s	0.007	0.006	0.006	0.009	0.006
The improved s	0.008	0.007	0.005	0.008	–

Table 3.4 shows that the CPU time(s) of Shifted CholeskyQR3 with different s are almost in the same level for the medium-size X, which indicates that our alternative choice s can keep the efficiency of Shifted CholeskyQR3 for such a T_1 matrix.

A large-size X in the general form

In this part, we form a large-size X in the general form. We take $m = 16384$ and $n = 1024$. We define some vectors in the beginning: $e_{1ns} = (1, 0, 0, \cdots, 0, 0)^\top \in \mathbb{R}^{1024}$, $e_{1zs} = (0, 1, 1, \cdots, 1, 1)^\top \in \mathbb{R}^{1024}$, $e_{1nb} = (1, 0, 0, \cdots, 0, 0)^\top \in \mathbb{R}^{16384}$ and $e_{1zs} = (0, 1, 1, \cdots, 1, 1)^\top \in \mathbb{R}^{16384}$, together with a diagonal matrix $E = \mathrm{diag}(1, \beta^{\frac{1}{1023}}, \cdots, \beta^{\frac{1022}{1023}}, \beta) \in \mathbb{R}^{1024\times1024}$. Moreover, a large matrix $\mathbb{O}_{15360\times1024}$ is formed with all the elements 0. Therefore, a matrix $P_{sparse} \in \mathbb{R}^{16384\times1024}$ is formed as

$$P_{sparse} = \begin{pmatrix} E \\ \mathbb{O}_{15360\times1024} \end{pmatrix}.$$

We build $X \in \mathbb{R}^{16384\times1024}$ as

$$X = -5e_{1nb} \cdot e_{1zs}^\top - 10e_{1zs} \cdot e_{1ns}^\top + P_{sparse} \tag{3.75}$$

Similar to the previous part, we build a comparison group with a common dense matrix $U_b \in \mathbb{R}^{16384\times1024}$. It is constructed in the same way as that in the previous section with $\sigma_1(U_b) = \|U_b\|_2 = 1$ and $\kappa_2(U_b) = \frac{1}{\sigma}$. Here, σ is a positive constant.

In the beginning, we test the applicability and accuracy of Shifted CholeskyQR3 between different s for such a large-size T_1 matrix X in the general form. Our X satisfies the ENC in Theorem 3.3 with $c = 10$, $v = 1$, $t_1 = 16384$ and $t_2 = 2$. We choose $s = j_s = \min(11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2, 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2)$ based on (3.1). Here, $j_s = 11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2$ with the ENC. We vary β from 10^{-6}, 10^{-7}, 10^{-8}, 10^{-9} to $\times 10^{-10}$ to adjust $\kappa_2(X)$. σ of U_b is also varying to ensure $\kappa_2(U_b) \approx \kappa_2(X)$. For U_b, we use $s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$ in Chapter 2. We test the applicability and accuracy of Shifted CholeskyQR3 with different s in the cases of X and U_b. Moreover, CPU time(s) is also tested for Shifted CholeskyQR3 with different s. All results are listed in Table 3.5–Table 3.8. The same as the previous section, we still refer to our alternative $s = j_s = \min(11(m\mathbf{u} + (n+1)\mathbf{u}) \cdot (vt_1 + nt_2)c^2, 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2)$ as '*the alternative s*' and $s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$ as '*the improved s*'.

Table 3.5: Shifted CholeskyQR3 with the alternative s for the large-size X

$\kappa_2(X)$	$1.28e+09$	$1.28e+10$	$1.28e+11$	$1.27e+12$	$1.27e+13$
Orthogonality	$2.67e-14$	$6.37e-14$	$9.19e-14$	$1.04e-13$	$1.19e-13$
Residual	$3.07e-13$	$2.92e-13$	$2.98e-13$	$2.82e-13$	$3.26e-13$

Table 3.6: Shifted CholeskyQR3 with the improved s for the large-size X

$\kappa_2(X)$	$1.28e+09$	$1.28e+10$	$1.28e+11$	$1.27e+12$	$1.27e+13$
Orthogonality	$8.95e-14$	$1.14e-13$	$1.24e-13$	$1.37e-13$	–
Residual	$2.93e-13$	$2.98e-13$	$3.16e-13$	$2.92e-13$	–

Table 3.7: Shifted CholeskyQR3 with the improved s for the large-size U_b

$\kappa_2(X)$	$1.28e+09$	$1.28e+10$	$1.28e+11$	$1.27e+12$	$1.27e+13$
Orthogonality	$1.96e-14$	$2.00e-14$	$2.06e-14$	$2.05e-14$	–
Residual	$1.93e-14$	$1.85e-14$	$1.79e-14$	$1.73e-14$	–

Table 3.8: Comparison of CPU time(s) with different s for the large-size X

$\kappa_2(X)$	$2.18e+07$	$1.99e+09$	$1.81e+11$	$1.63e+13$	$1.46e+15$
The alternative s	1.63	1.71	1.73	1.69	1.70
The improved s	1.61	1.70	1.71	1.72	–

According to Table 3.5-Table 3.8, similar findings hold for the large-size X in the general form as

those of the medium-size X in the block version. We can say that our Shifted CholeskyQR3 with the alternative s for the sparse matrices exhibits the advantages compared to case with the improved s in Chapter 2 under certain ENCs, showing that such an alternative s is an optimal one for T_1 matrices.

3.3.2 T_2 matrices

T_2 matrices with all columns being sparse are also very common in real applications, such as scientific computing, machine learning, and image processing [54, 59, 63]. Similar to the case of T_1 matrices, we do two groups of numerical experiments with a medium size X in the block version and a large-size X in the general form.

A medium-size X in the block version

For the medium-size X, we still take $m = 2048$ and $n = 64$. We form matrices $X \in \mathbb{R}^{2048\times 64}$ with 32 $K \in \mathbb{R}^{64\times 64}$ as

$$X = \begin{pmatrix} K \\ K \\ \vdots \\ K \end{pmatrix}.$$

Similar to the construction of a diagonal matrix P, the diagonal matrix $P = \mathrm{diag}(u) \in \mathbb{R}^{64\times 64}$, where $u = (u_1, u_2, \cdots, u_{63}, u_{64})$ and $u_i = \begin{cases} 10, & \text{if } i = 1, 2, \cdots, 32 \\ 10 \cdot (\frac{b}{10})^{\frac{i-33}{31}}, & \text{if } i = 33, 34, \cdots, 64 \end{cases}$. Here, b is a small positive constant. We utilize the definition of the orthogonal basis and form K as

$$K = 10e_{32} \cdot d^\top + 10e_{33} \cdot d^\top + P.$$

Here, $d \in \mathbb{R}^{64}$ is a vector with all the elements 1. Therefore, X is also formed. The comparison group of the common dense matrix U is built in the same way as the part of T_1 matrices.

In the beginning, we make comparison of the applicability and accuracy between different s for such a medium-size T_2 matrix. We choose $s = j_s = \min(11(mn\mathbf{u}+n(n+1)\mathbf{u})\cdot t_2c^2, 11(m\mathbf{u}+(n+1)\mathbf{u})\|X\|_c^2)$ based on (3.1). According to Theorem 3.4, we have $s = j_s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$. We vary b from 10^{-5}, 10^{-7}, 10^{-9}, 10^{-11} to 10^{-13} to adjust $\kappa_2(X_s)$ and $\kappa_2(X)$. Meanwhile, we vary the σ of U to ensure $\kappa_2(U) \approx \kappa_2(X)$. For U, we use $s = 11(m\mathbf{u} + (n+1)\mathbf{u})\|X\|_c^2$. We test the applicability of Shifted CholeskyQR3 with different s for both X and U. The corresponding results are listed in Table 3.9 and Table 3.10.

Table 3.9: Shifted CholeskyQR3 with the alternative s for the medium-size X

$\kappa_2(X)$	$1.30e+07$	$1.29e+09$	$1.28e+11$	$1.28e+13$	$1.28e+15$
Orthogonality	$2.05e-15$	$2.06e-15$	$2.20e-15$	$2.05e-15$	$2.22e-15$
Residual	$3.42e-13$	$3.51e-13$	$1.65e-13$	$3.32e-13$	$3.47e-13$

Table 3.10: Shifted CholeskyQR3 with the improved s for U

$\kappa_2(X)$	$1.30e+07$	$1.29e+09$	$1.28e+11$	$1.28e+13$	$1.28e+15$
Orthogonality	$2.13e-15$	$1.98e-15$	$1.94e-15$	$2.07e-15$	–
Residual	$6.95e-16$	$6.56e-16$	$6.19e-16$	$5.74e-16$	–

According to Table 3.9 and Table 3.10, we observe that similar results hold for such a medium-size T_2 matrix in the block version as for T_1 matrices. With the alternative s and appropriate ENCs, Shifted CholeskyQR3 can handle cases with larger $\kappa_2(X)$ compared to the dense cases in this example. This highlights the difference between the sparse and the dense cases for Shifted CholeskyQR3. Furthermore, with the alternative s, Shifted CholeskyQR3 remains numerically stable for such a medium-size T_2 matrix in the block version, as indicated by Chapter 2 and Theorem 3.4.

In the following, we show the CPU time(s) of Shifted CholeskyQR3 with different s when X is a medium-size T_2 matrix in the block version. We do hundreds of tests and take the average of the CPU time(s) of different s. They are presented in Table 3.11.

Table 3.11: Comparison of CPU time(s) with different s for the medium size X

$\kappa_2(X)$	$1.30e+07$	$1.29e+09$	$1.28e+11$	$1.28e+13$	$1.28e+15$
The alternative s	0.009	0.008	0.010	0.007	0.009
The improved s	0.007	0.007	0.011	0.009	–

According to Table 3.11, we find that Shifted CholeskyQR3 exhibits similar CPU times for Shifted CholeskyQR3 with different s values in this example, which aligns with the conclusion drawn when X is a T_1 matrix. Although j_s for the T_2 matrix is equivalent to the improved s from Chapter 2, we can still use $s = j_s$ because j_s in (3.1) represents a common form applicable to all the sparse matrices.

A large-size X in the general form

For T_2 matrices, we also form a large-size X in the general form. We take $m = 16384$ and $n = 1024$. We define a vector $u_t \in \mathbb{R}^{1024}$ as $u_t = (10, 10, \cdots, 10, 10)^\top$. We define a diagonal matrix

$E_s = \mathrm{diag}(1, b_1^{\frac{1}{1023}}, \cdots, b_1^{\frac{1022}{1023}}, b_1) \in \mathbb{R}^{1024\times1024}$. This is the same as that in Chapter 2. Moreover, we build two matrices with all the elements 0, $\mathbb{O}_{15360\times1024}$ and $\mathbb{O}_{8191\times1024}$. Therefore, a matrix $P_{sparse} \in \mathbb{R}^{16384\times1024}$ is formed as

$$D_{sparse} = \begin{pmatrix} E_s \\ \mathbb{O}_{15360\times1024} \end{pmatrix}.$$

Another matrix $C_{sparse} \in \mathbb{R}^{16384\times1024}$ is defined as

$$C_{sparse} = \begin{pmatrix} \mathbb{O}_{8191\times1024} \\ u_t \\ u_t \\ \mathbb{O}_{8191\times1024} \end{pmatrix}.$$

We build $X \in \mathbb{R}^{16384\times1024}$ as

$$X = C_{sparse} + D_{sparse}.$$

In the beginning, we make comparison of the applicability and accuracy between the cases with different s for such a large-size T_2 matrix. We choose $s = j_s = \min(11(mn\mathbf{u}+n(n+1)\mathbf{u})\cdot t_2c^2, 11(m\mathbf{u}+(n+1)\mathbf{u})\|X\|_c^2)$ based on (3.1). We vary b_1 from 10^{-7}, 10^{-8}, 10^{-9}, 10^{-10} to 10^{-11} to adjust $\kappa_2(X)$. For the comparison group, we take the same U_b based on SVD as that in the previous section for T_1 matrices. We vary σ of U_b to ensure $\kappa_2(U) \approx \kappa_2(X)$. For U_b, we use $s = 11(m\mathbf{u}+(n+1)\mathbf{u})\|X\|_c^2$. We test the applicability of Shifted CholeskyQR3 with different s for both X and U_b. The same as that of T_1 matrices, CPU time(s) is also tested for Shifted CholeskyQR3 with different s. The corresponding results are listed in Table 3.12-Table 3.14.

Table 3.12: Shifted CholeskyQR3 with the alternative s for the large-size X

$\kappa_2(X)$	$1.85e+10$	$1.73e+11$	$1.64e+12$	$1.56e+13$	$1.49e+14$
Orthogonality	$2.67e-14$	$6.37e-14$	$9.19e-14$	$1.04e-13$	$1.19e-13$
Residual	$3.07e-13$	$2.92e-13$	$2.98e-13$	$2.83e-13$	$3.26e-13$

Table 3.13: Shifted CholeskyQR3 with the improved s for U_b

$\kappa_2(X)$	$1.85e+10$	$1.73e+11$	$1.64e+12$	$1.56e+13$	$1.49e+14$
Orthogonality	$8.95e-14$	$1.14e-13$	$1.24e-13$	$1.37e-13$	–
Residual	$2.93e-13$	$2.98e-13$	$3.16e-13$	$2.92e-13$	–

Table 3.14: Comparison of CPU time(s) with different s for the large-size X

$\kappa_2(X)$	$1.85e+10$	$1.73e+11$	$1.64e+12$	$1.56e+13$	$1.49e+14$
The alternative s	1.66	1.74	1.72	1.75	1.69
The improved s	1.64	1.73	1.74	1.73	1.69

According to Table 3.12-Table 3.14, we find that Shifted CholeskyQR3 with the alternative s exhibits good properties in the applicability, accuracy and efficiency for such a large-size X in the general form, which is not worse than the case with the improved s proposed in Chapter 2. Generally speaking, our Shifted CholeskyQR3 with the alternative s performs well for T_2 matrices. Combing with the theoretical results and numerical experiments for both T_1 and T_2 matrices, we can say that our alternative s is an optimal one compared to the improved s for Shifted CholeskyQR3 in sparse cases.

3.4 Conclusions

This chapter focuses on the theoretical analysis of Shifted CholeskyQR3 for sparse matrices. We divide sparse matrices into two types: T_1 matrices and T_2 matrices based on the presence of dense columns. We propose an alternative choice of the shifted item s based on the structure and the key element of the input X, which is a novel approach compared to the existing works. Our rounding error analysis demonstrates that this alternative s is optimal for T_1 matrices and can ensure numerical stability of Shifted CholeskyQR3 with certain element-norm conditions(ENCs). Numerical experiments verify our theoretical results for T_1 matrices. Furthermore, Shifted CholeskyQR3 exhibits new properties for T_2 matrices compared to dense cases, and it remains as efficient with our alternative s as with the improved s from Chapter 2.

CHAPTER 4.
PROBABILISTIC ERROR ANALYSIS OF CHOLESKYQR BASED ON COLUMNS

In this chapter, we do probabilistic error analysis of CholeskyQR-type algorithms with the randomized models in [33] and $\|X\|_c$ defined in Chapter 2 for the input matrix X. Different from other works of probabilistic error analysis, all the steps of CholeskyQR-type algorithms are matrix multiplications and matrix factorization. Therefore, we set that all the steps of CholeskyQR-type algorithms in this chapter follow Lemma 1.10-Lemma 1.12 independently. We receive tighter upper bounds for both orthogonality and residual for Shifted CholeskyQR3 and Shifted CholeskyQR2, together with an improved probabilistic shifted item s for Shifted CholeskyQR3 compared to that in Chapter 2. Numerical experiments demonstrate that the improvement of such a probabilistic s on the applicability and show its robustness in ill-conditioned cases. This chapter is organized as follows. We present the probabilistic error analysis for CholeskyQR2 in Section 4.1 and for Shifted CholeskyQR3 in Section 4.2. Detailed numerical experiments are provided in Section 4.3.

4.1 Probabilistic error analysis of CholeskyQR2

In this section, we aim to utilize the randomized models to conduct a probabilistic error analysis of CholeskyQR2. The same as Chapter 2, $\|\cdot\|_g$ and its properties are utilized in the theoretical analysis.

4.1.1 General settings

In the beginning, we present CholeskyQR2 step by step, accompanied by the corresponding error matrices below.

$$G - X^\top X = E_A, \tag{4.1}$$

$$Y^\top Y - G = E_B, \tag{4.2}$$

$$WY = X + E_{WY}, \tag{4.3}$$

$$C - W^\top W = E_1,$$

$$Z^\top Z - C = E_2,$$

$$QZ - W = E_3, \tag{4.4}$$

$$ZY - R = E_4. \tag{4.5}$$

For the input matrix $X \in \mathbb{R}^{m\times n}$, we provide some general settings for all the algorithms in this chapter below.

$$\max(\eta\sqrt{mn}\mathbf{u}, mn\mathbf{u}) \leq \frac{1}{64}, \tag{4.6}$$

$$\max(\eta\sqrt{n+1}n\mathbf{u}, (n+1)n\mathbf{u}) \leq \frac{1}{64}. \tag{4.7}$$

Here, η occurs in (1.20) and (1.21). For CholeskyQR2, when (4.6) and (4.7) are satisfied, if we want to have $Q(\eta, mn^2)$, $Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3})$ and $Q(\eta, n^3)$ to be all positive, we can choose η as a positive constant not exceeding 10 in numerical experiments. The same as that in Chapter 2, we keep $j_1 = \frac{\|X\|_c}{\|X\|_2}$. Moreover, we define $j_2 = \frac{\|W\|_c}{\|W\|_2}, 1 \leq j_i \leq \sqrt{n}, i = 1, 2$.

4.1.2 Probabilistic error analysis of CholeskyQR2

In this section, we present some theoretical results related to the probabilistic error analysis of CholeskyQR2.

Theorem 4.1. *With Lemma 1.10-Lemma 1.12, for $X \in \mathbb{R}^{m\times n}$ and $[Q, R] = CholeskyQR2(X)$, with* (4.6), (4.7) *and*

$$k_1 = 8j_1\kappa_2(X)\sqrt{\eta(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u})} \leq 1, \tag{4.8}$$

we have

$$\left\|Q^\top Q - I\right\|_F \leq 6\eta \cdot j_2^2(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u}), \tag{4.9}$$

$$\|QR - X\|_F \leq (1.1j_1 + 1.23j_2 + 1.19 \cdot \frac{j_1 j_2}{\sqrt{n}})\eta \cdot n\mathbf{u}\|X\|_2, \tag{4.10}$$

with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3}))^2 Q(\eta, n^3)$.

4.1.3 Lemmas for proving Theorem 4.1

Before proving Theorem 4.1, we present some lemmas related to it. The analytical steps of these lemmas in this chapter are similar to those in [21, 68], Chapter 2 and Chapter 3. However, we utilize the randomized models, allowing us to obtain sharper upper bounds with minimal probabilities for all results, which are fundamentally different from existing works.

Lemma 4.1. *For E_A and E_B in* (4.1) *and* (4.2), *we have*

$$\|E_A\|_2 \leq 1.1\eta\sqrt{m}\mathbf{u}\|X\|_c^2, \tag{4.11}$$

$$\|E_B\|_2 \leq 1.1\eta\sqrt{n+1}\mathbf{u}\|X\|_c^2, \tag{4.12}$$

with probability at least $Q(\eta, mn^2)Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3})$.

Proof. Regarding $\|E_A\|_2$, with Lemma 1.7 and (4.1), we can have

$$\begin{aligned} |E_A| =& \left|G - X^\top X\right| \\ &\leq \tilde{\gamma}_m(\eta)\left|X^\top\right||X| \\ &\leq 1.1\eta\sqrt{m}\mathbf{u}\cdot\left|X^\top\right||X|, \end{aligned} \tag{4.13}$$

with probability at least $Q(\eta, mn^2)$. Similar to those in [21, 69] and Chapter 2, we can bound $\|E_A\|_2$ as

$$\begin{aligned} \|E_A\|_2 \leq \left\||E_A|\right\|_F &\leq \tilde{\gamma}_m(\eta)\cdot\|X\|_F^2 \\ &\leq \tilde{\gamma}_m(\eta)\cdot\|X\|_c^2 \\ &\leq 1.1\eta\sqrt{m}\mathbf{u}\cdot\|X\|_c^2, \end{aligned}$$

with probability at least $Q(\eta, mn^2)$. Here, x_i denotes the i-th column of X. (4.11) is proved.

Regarding $\|E_B\|_2$, we use similar ideas in Chapter 2 with $\|\cdot\|_g$ and its properties. With Lemma 1.12, (4.1) and (4.2), we can have

$$\begin{aligned} \|E_B\|_2 \leq& \left\||E_B|\right\|_F \\ &\leq \tilde{\gamma}_{n+1}(\eta)\|Y\|_F^2 \\ &\leq \tilde{\gamma}_{n+1}(\eta)\cdot n\|Y\|_g^2 \\ &\leq \tilde{\gamma}_{n+1}(\eta)\cdot n(\|X\|_g^2 + \|E_A\|_2 + \|E_B\|_2), \end{aligned} \tag{4.14}$$

with probability at least $Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. Based on (2.20), we can get an deterministic upper bound of $\|E_A\|_2$ as $1.1mn\mathbf{u}\|X\|_g^2$. Therefore, with (4.6), (4.7), (4.11) and (4.14), we can get

$$\begin{aligned} \|E_B\|_2 &\leq \frac{\tilde{\gamma}_{n+1}(\eta)\cdot n(1+1.1mn\mathbf{u})}{1-\tilde{\gamma}_{n+1}(\eta)\cdot n}\|X\|_g^2 \\ &\leq \frac{1.02\eta\sqrt{n+1}\cdot n\mathbf{u}(1+1.1mn\mathbf{u})}{1-1.02\eta\sqrt{n+1}\cdot n\mathbf{u}}\|X\|_g^2 \\ &\leq \frac{1.02\eta\sqrt{n+1}\cdot n\mathbf{u}\cdot(1+1.1\cdot\frac{1}{64})}{1-\frac{1.02}{64}}\|X\|_g^2 \\ &\leq 1.1\eta\sqrt{n+1}\cdot n\mathbf{u}\|X\|_g^2 \\ &= 1.1\eta\sqrt{n+1}\mathbf{u}\|X\|_c^2, \end{aligned}$$

with probability at least $Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. (4.12) is proved. Therefore, Lemma 4.1 holds. □

Lemma 4.2. *For Y^{-1} and XY^{-1} in (4.3), we have*

$$\left\|Y^{-1}\right\|_2 \leq \frac{1.1}{\sigma_{min}(X)}, \tag{4.15}$$

$$\left\|XY^{-1}\right\|_2 \leq 1.1, \tag{4.16}$$

with probability at least $Q(\eta, mn^2)Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$.

Proof. The idea to prove Lemma 4.2 is the same as that in [68]. Based on Lemma 1.6, (4.1) and (4.2), we can have

$$(\sigma_{min}(Y))^2 \geq (\sigma_{min}(X))^2 - (\|E_A\|_2 + \|E_B\|_2). \tag{4.17}$$

Based on (4.8), (4.11) and (4.12), we can have

$$\begin{aligned} \|E_A\|_2 + \|E_B\|_2 &\leq \frac{1.1}{64}(\sigma_{min}(X))^2 \\ &\leq (1 - \frac{1}{1.1^2})(\sigma_{min}(X))^2, \end{aligned} \tag{4.18}$$

with probability at least $Q(\eta, mn^2)Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. We combine (4.17) with (4.18) and we can have

$$\frac{1}{1.1^2} \cdot (\sigma_{min}(X))^2 \leq (\sigma_{min}(Y))^2, \tag{4.19}$$

with probability at least $Q(\eta, mn^2)Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. Therefore, we can easily get (4.15). Similar to [68], we can have (4.16). Lemma 4.2 holds. □

Lemma 4.3. *For E_{WY} in* (4.3)*, we have*

$$\|E_{WY}\|_2 \leq 1.05\eta n\mathbf{u} \cdot \|W\|_2 \|X\|_c \,, \tag{4.20}$$

with probability at least $Q(\eta, mn^2)$.

Proof. With Lemma 1.11 and (4.3), we can have

$$\begin{aligned} \|E_{WY}\|_2 &\leq 1.02\eta\sqrt{n}\mathbf{u} \cdot (\|W\|_F \cdot \|Y\|_F) \\ &\leq 1.02\eta n\sqrt{n}\mathbf{u}\|W\|_2\|Y\|_g \,, \end{aligned} \tag{4.21}$$

with probability at least $Q(\eta, mn^2)$. In Chapter 2, we show the deterministic upper bounds of both $\|E_A\|_2$ and $\|E_B\|_2$ in (4.6) and (4.7). Based on (4.1), (4.2), (4.6) and (4.7), we can have

$$\begin{aligned} \|Y\|_g^2 &\leq \|X\|_g^2 + \|E_A\|_2 + \|E_B\|_2) \\ &\leq 1.04\|X\|_g^2 \,. \end{aligned} \tag{4.22}$$

Based on (4.22), we can have

$$\|Y\|_g \leq 1.02\|X\|_g \,. \tag{4.23}$$

Therefore, we put (4.23) into (4.21) and we can get (4.20). Lemma 4.3 holds. □

Lemma 4.4. *For W in* (4.3)*, we have*

$$\|W\|_2 \leq 1.13, \tag{4.24}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$.

Proof. Based on (4.3), we can have

$$\begin{aligned}\|W\|_2 \leq & \left\|XY^{-1}\right\|_2 + \left\|E_{WY}Y^{-1}\right\|_2 \\ \leq & \left\|XY^{-1}\right\|_2 + \|E_{WY}\|_2\left\|Y^{-1}\right\|_2 .\end{aligned} \tag{4.25}$$

With (4.7), (4.15) and (4.20), we can have

$$\begin{aligned}\|E_{WY}\|_2\left\|Y^{-1}\right\|_2 \leq & \frac{1.16\eta n\mathbf{u}\cdot\|W\|_2\|X\|_c}{\sigma_{min}(X)} \\ \leq & \frac{1.16j_1\eta n\mathbf{u}\|X\|_2}{8j_1\sqrt{\eta(\sqrt{m}\mathbf{u}+\sqrt{n+1}\mathbf{u})}}\cdot\|W\|_2 \\ \leq & \frac{1.06}{8}\cdot\sqrt{\eta n\sqrt{n+1}\mathbf{u}}\cdot\|W\|_2 \\ \leq & 0.02\|W\|_2 ,\end{aligned} \tag{4.26}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3})$. Therefore, we put (4.60) and (4.26) into (4.25) and we can have

$$\|W\|_2 \leq 1.1 + 0.02\|W\|_2 , \tag{4.27}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3})$. With (4.27), we can have (4.24). Lemma 4.4 holds. □

4.1.4 Proof of Theorem 4.1

With Lemma 4.1-Lemma 4.4, we begin to prove Theorem 4.1.

Proof. The proof of Theorem 4.1 is divided into two parts, orthogonality and residual.

The upper bound of orthogonality

First, we consider the orthogonality. Based on (4.1), (4.2) and (4.3), it is easy to get

$$\begin{aligned}W^\top W = & Y^{-\top}(X+E_{WY})^\top(X+E_{WY})Y^{-1} \\ = & Y^{-\top}X^\top XY^{-1} + Y^{-\top}X^\top E_{WY}Y^{-1} \\ & + Y^{-\top}E_{WY}^\top XY^{-1} + Y^{-\top}E_{WY}^\top E_{WY}Y^{-1} \\ = & I - Y^{-\top}(E_A+E_B)Y^{-1} + (XY^{-1})^\top E_{WY}Y^{-1} \\ & + Y^{-\top}E_{WY}^\top(XR^{-1}) + Y^{-\top}E_{WY}^\top E_{WY}Y^{-1}.\end{aligned}$$

Therefore, we can have

$$\begin{aligned}\left\|W^\top W - I\right\|_2 \leq & \left\|Y^{-1}\right\|_2^2(\|E_A\|_2 + \|E_B\|_2) + 2\left\|Y^{-1}\right\|_2\left\|XY^{-1}\right\|_2\|E_{WY}\|_2 \\ & + \left\|Y^{-1}\right\|_2^2\|E_{WY}\|_2^2 .\end{aligned} \tag{4.28}$$

Based on (4.8), (4.11), (4.12) and (4.15), when $j_1 = \frac{\|X\|_c}{\|X\|_2}$, we can have

$$\begin{aligned}\left\|Y^{-1}\right\|_2^2 (\|E_A\|_2 + \|E_B\|_2) &\leq \frac{1.21 \cdot (1.1 j_1^2 \cdot \eta(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u})\|X\|_2^2)}{(\sigma_{min}(X))^2} \\ &\leq \frac{1.34}{64} k_1^2,\end{aligned} \tag{4.29}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. Based on (4.8), (4.15), (4.16), (4.20) and (4.33), we can have

$$\begin{aligned}2\left\|Y^{-1}\right\|_2 \left\|XY^{-1}\right\|_2 \|E_{WY}\|_2 &\leq 2 \cdot \frac{1.1}{\sigma_{min}(X)} \cdot 1.1 \cdot (1.05 j_1 \cdot \eta n \mathbf{u} \|X\|_2 \cdot 1.13) \\ &\leq \frac{3}{64} k_1,\end{aligned} \tag{4.30}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. With (4.8), (4.15), (4.20) and (4.33), we can have

$$\begin{aligned}\left\|Y^{-1}\right\|_2^2 \|E_{WY}\|_2^2 &\leq \frac{1.21}{(\sigma_{min}(X))^2} \cdot (1.05 j_1 \cdot \eta n \mathbf{u} \|X\|_2 \cdot 1.13)^2 \\ &\leq \frac{2}{4096} k_1^2,\end{aligned} \tag{4.31}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. Therefore, we put (4.29)-(4.31) into (4.28) and with (4.8), we can have

$$\left\|W^\top W - I\right\|_2 \leq \frac{5}{64}, \tag{4.32}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. With (4.32), it is easy to have

$$\|W\|_2 \leq \frac{\sqrt{69}}{8}, \tag{4.33}$$

$$\sigma_{min}(W) \geq \frac{\sqrt{59}}{8}, \tag{4.34}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. (4.33) is an improved upper bound of $\|W\|_2$ compared to (4.24). With (4.33) and (4.34), we can get

$$\kappa_2(W) \leq \sqrt{\frac{69}{59}}, \tag{4.35}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. Based on (4.6), (4.7) and (4.35), when $j_2 \leq 1$, we can get

$$k_2 = 8 j_2 \kappa_2(W) \sqrt{\eta(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u}} \leq 1. \tag{4.36}$$

With (4.36) and similar to the previous steps to get (4.32), we can have

$$\begin{aligned}\left\|Q^\top Q - I\right\|_F &\leq \frac{5}{64} k_2^2 \\ &\leq 6 j_2^2 \cdot \eta(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u}),\end{aligned}$$

with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3}))^2$. (4.9) holds.

The upper bound of residual

Regarding the residual, according to (4.4) and (4.5), we can have

$$\begin{aligned} QR - X &= (W + E_3)Z^{-1}(ZY - E_4) - X \\ &= (W + E_3)Y - (W + E_3)Z^{-1}E_4 - X \\ &= WY - X + E_3Y - QE_4. \end{aligned}$$

Therefore, it is easy to have

$$\|QR - X\|_F \leq \|WY - X\|_F + \|E_3\|_F \|Y\|_2 + \|Q\|_2 \|E_4\|_F . \tag{4.37}$$

Based on (4.20) and (4.33), we can have

$$\begin{aligned} \|WY - X\|_F &\leq 1.05\eta n\mathbf{u} \cdot \|W\|_2 \|X\|_c \\ &\leq \frac{\sqrt{69}}{8} \cdot 1.05 j_1 \cdot \eta n\mathbf{u} \|X\|_2 \\ &\leq 1.1 j_1 \cdot \eta n\mathbf{u} \|X\|_2 , \end{aligned} \tag{4.38}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. We replace $\|Y\|_g$ and $\|X\|_g$ in (4.22) with $\|Y\|_2$ and $\|X\|_2$ and we can get

$$\|Y\|_2 \leq 1.02 \|X\|_2 . \tag{4.39}$$

With (4.6), (4.7) and (4.9), we can have

$$\|Q\|_2 \leq 1.1, \tag{4.40}$$

with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3}))^2$. Similar to the steps of (4.38), with (4.33) and (4.40), we can bound $\|E_3\|_F$ as

$$\begin{aligned} \|E_3\|_F &\leq 1.05\eta n\mathbf{u} \cdot \|Q\|_2 \|W\|_c \\ &\leq 1.05\eta n\mathbf{u} \cdot 1.1 \cdot \frac{\sqrt{69}}{8} j_2 \\ &\leq 1.2 j_2 \cdot \eta n\mathbf{u}, \end{aligned} \tag{4.41}$$

with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3}))^2$. Similar to the step in [68], Chapter 2 and Chapter 3, with (4.23) and (4.33), we can get

$$\begin{aligned} \|Z\|_g &\leq 1.02 \|W\|_g \\ &\leq \frac{1.02 j_2}{\sqrt{n}} \cdot \frac{\sqrt{69}}{8} \\ &\leq \frac{1.06 j_2}{\sqrt{n}}, \end{aligned} \tag{4.42}$$

with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3}))^2$. With Lemma 1.11, (4.23) and (4.42), we can bound $\|E_4\|_F$ as

$$\begin{aligned} \|E_4\|_F &\leq \tilde{\gamma}_n(\eta)(\|Z\|_F \cdot \|Y\|_F) \\ &\leq \tilde{\gamma}_n(\eta)(\sqrt{n}\|Z\|_g \cdot \sqrt{n}\|Y\|_g) \\ &\leq 1.02\eta\sqrt{n}\mathbf{u} \cdot 1.06 j_2 \cdot 1.02 j_1 \|X\|_2 \\ &\leq 1.08 j_1 j_2 \cdot \eta\sqrt{n}\mathbf{u}\|X\|_2 , \end{aligned} \tag{4.43}$$

with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3}))^2 Q(\eta, n^3)$. We put (4.38)-(4.40), (4.41) and (4.43) into (4.37) and we can have (4.10) with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3}))^2 Q(\eta, n^3)$. Therefore, Theorem 4.1 holds. □

Remark 4.1. *In fact, our theoretical analysis of CholeskyQR is very different from those of [21, 68] and Chapter 2. Regarding probabilistic analysis, the original analysis of CholeskyQR may lead to a very limited least probability because of some lemmas using the way of solving linear systems through each row. The analysis in this part is a more direct way and can avoid the problem. We utilize deterministic bounds of* $\|E_A\|_2$ *and* $\|E_B\|_2$ *to derive* (4.22)*, demonstrating the connection between the deterministic and the the probabilistic results, which is a significant innovation in this chapter. Theorem 4.1 provides sharper theoretical upper bounds for CholeskyQR2 with the randomized models compared to Lemma 1.1 when* n *is large. Furthermore, the sufficient condition for* $\kappa_2(X)$ *is also significantly better than that in [68] when* m *is large.*

4.2 Probabilistic error analysis for Shifted CholeskyQR3

In this part, we provide probabilistic error analysis of Shifted CholeskyQR3 with an alternative shifted item s based on the randomized models.

4.2.1 General settings and algorithms

In the beginning, we write Shifted CholeskyQR3 with error matrices step by step below. It is the same as that in Chapter 2 and Chapter 3.

$$G - X^\top X = E_A, \tag{4.44}$$

$$Y^\top Y = G + sI + E_B, \tag{4.45}$$

$$WY = X + E_{WY}, \tag{4.46}$$

$$C - W^\top W = E_1,$$

$$D^\top D - C = E_2,$$

$$VD - W = E_3, \tag{4.47}$$

$$DY - N = E_4, \tag{4.48}$$

$$B - V^\top V = E_5,$$

$$J^\top J - B = E_6,$$

$$QJ - V = E_7, \tag{4.49}$$

$$JN - R = E_8. \tag{4.50}$$

For Shifted CholeskyQR3, (4.6) and (4.7) still hold. We present more general settings below.

$$\kappa_2(X) \le L, \tag{4.51}$$

$$11\eta(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u})\|X\|_c^2 \le s \le \frac{1}{100n}\|X\|_c^2. \tag{4.52}$$

The same as before, we have $j_1 = \frac{\|X\|_c}{\|X\|_2}, j_2 = \frac{\|W\|_c}{\|W\|_2}$. Furthermore, we let $j_3 = \frac{\|V\|_c}{\|V\|_2}$. Here, $1 \le j_i \le \sqrt{n}, i = 1, 2, 3$. We define

$$L = \min(\frac{1}{4.89 j_1 \cdot \eta m \mathbf{u}}, \Phi),$$
$$\Phi = \frac{1}{86 j_1 j_2 \cdot \eta(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u})}.$$

4.2.2 Probabilistic error analysis of Shifted CholeskyQR3

In this section, we present theoretical results of the probabilistic error analysis for Shifted CholeskyQR3 based on Lemma 1.10-Lemma 1.12.

The same as the corresponding steps in Chapter 2 and Chapter 3, we divide the calculation of R in the last step of Shifted CholeskyQR3 into (4.48) and (4.50). In the following, we show some theoretical results of probabilistic error analysis of Shifted CholeskyQR3 below.

Theorem 4.2. *With Lemma 1.10-Lemma 1.12, for $X \in \mathbb{R}^{m\times n}$ and $[W,Y] = SCholeskyQR(X)$, when* (4.51) *and* (4.52) *are satisfied, we have*

$$\kappa_2(W) \le 3.24\sqrt{1+t(\kappa_2(X))^2}, \tag{4.53}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3})$. Here, $t = \frac{s}{\|X\|_2^2}$.

Theorem 4.3. *With Lemma 1.10-Lemma 1.12, for $X \in \mathbb{R}^{m\times n}$ and $[Q,R] = SCholeskyQR3(X)$, when $\kappa_2(X)$ is large enough, if we take $s = 11\eta(\sqrt{m}\mathbf{u}+\sqrt{n+1}\mathbf{u})\|X\|_c^2$ and* (4.51) *is satisfied, we have*

$$\left\|Q^\top Q - I\right\|_F \le 6\eta \cdot j_3^2(\sqrt{m}\mathbf{u}+\sqrt{n+1}\mathbf{u}), \tag{4.54}$$

$$\|QR - X\|_F \le \phi_1(j_1,j_2,j_3)\eta \cdot n\mathbf{u}\|X\|_2, \tag{4.55}$$

with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3}))^3 (Q(\eta, n^3))^2$. Here, $\phi_1(j_1,j_2,j_3,n) = (1.66j_1 + 1.71j_2 + 1.78j_3 + 1.71\cdot\frac{j_1 j_2}{\sqrt{n}} + 1.70\cdot\frac{j_1 j_3}{\sqrt{n}})$.

4.2.3 Lemmas for proving Theorem 4.2 and Theorem 4.3

To prove Theorem 4.2 and Theorem 4.3, we present the following lemmas.

Lemma 4.5. *For E_A and E_B in* (4.44) *and* (4.45)*, when* (4.52) *is satisfied, we have*

$$\|E_A\|_2 \le 1.1\eta\sqrt{m}\mathbf{u}\|X\|_c^2, \tag{4.56}$$

$$\|E_B\|_2 \le 1.1\eta\sqrt{n+1}\mathbf{u}\|X\|_c^2, \tag{4.57}$$

with probability at least $Q(\eta, mn^2)Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3})$.

Proof. When estimating $\|E_A\|_2$, the same as Lemma 4.1, we can get (4.56) with probability at least $Q(\eta, mn^2)$.

Regarding $\|E_B\|_2$, with Lemma 2.2 and similar to (4.14), we can get

$$\begin{aligned}\|E_B\|_2 \le \left\||E_B|\right\|_F &\le \tilde{\gamma}_{n+1}(\eta)\left\||Y|\right\|_F^2 \\ &\le \tilde{\gamma}_{n+1}(\eta)\cdot n\|Y\|_g^2 \\ &\le \tilde{\gamma}_{n+1}(\eta)\cdot n(\|X\|_g^2 + s + \|E_A\|_2 + \|E_B\|_2),\end{aligned} \tag{4.58}$$

with probability at least $Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3})$. Since the deterministic upper bound of $\|E_A\|_2$ can be taken as $1.1m\mathbf{u}\|X\|_c^2$ according to (2.20), with Lemma 1.12, (4.6), (4.7), (4.56) and (4.58), when

$t_1 = \frac{s}{\|X\|_g^2} \leq \frac{1}{100}$, we can get

$$\begin{aligned}
\|E_B\|_2 &\leq \frac{\tilde{\gamma}_{n+1}(\eta)n((1+\gamma_m \cdot n + t_1)}{1-\tilde{\gamma}_{n+1}(\eta)n}\|X\|_g^2 \\
&\leq \frac{1.02\eta\sqrt{n+1}n\mathbf{u}\cdot(1+1.1mn\mathbf{u}+t_1)}{1-1.02\eta\sqrt{n+1}n\mathbf{u}}\|X\|_g^2 \\
&\leq \frac{1.02\eta\sqrt{n+1}n\mathbf{u}\cdot(1+1.1\cdot\frac{1}{64}+0.01)}{1-\frac{1.02}{64}}\|X\|_g^2 \\
&\leq 1.1\eta\sqrt{n+1}n\mathbf{u}\|X\|_g^2 \\
&\leq 1.1\eta\sqrt{n+1}\mathbf{u}\|X\|_c^2,
\end{aligned}$$

with probability at least $Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3})$. Therefore, (4.57) holds. Based on the results above, Lemma 4.5 holds. □

Lemma 4.6. *For Y^{-1} and XY^{-1}, we have*

$$\left\|Y^{-1}\right\|_2 \leq \frac{1}{\sqrt{(\sigma_{min}(X))^2+0.9s}}, \tag{4.59}$$

$$\left\|XY^{-1}\right\|_2 \leq 1.5, \tag{4.60}$$

with probability at least $Q(\eta, mn^2)Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3})$.

Proof. The proofs of (4.59) and (4.60) follow the same approach as that in [21], Chapter 2 and Chapter 3. Since (4.56) and (4.57) are used in the proof, (4.59) and (4.60) hold with a probability of at least $Q(\eta, mn^2)Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3})$. Thus, Lemma 4.6 holds. □

Lemma 4.7. *For E_{WY} in* (4.46)*, we have*

$$\|E_{WY}\|_2 \leq 1.03\eta n\mathbf{u}\cdot\|W\|_2\|X\|_c, \tag{4.61}$$

with probability at least $Q(\eta, mn^2)$.

Proof. With Lemma 1.11 and (4.46), we can have

$$\begin{aligned}
\|E_{WY}\|_2 &\leq 1.02\eta\sqrt{n}\mathbf{u}\cdot(\|W\|_F\cdot\|Y\|_F) \\
&\leq 1.02\eta n\sqrt{n}\mathbf{u}\|W\|_2\|Y\|_g,
\end{aligned} \tag{4.62}$$

with probability at least $Q(\eta, mn^2)$. Similar to (4.22), we utilize the deterministic bounds of $\|E_A\|_2$ and $\|E_B\|_2$ in (4.6) and (4.7) in Chapter 2. Based on (4.6), (4.7), (4.44), (4.45) and (4.52), we can have

$$\begin{aligned}
\|Y\|_g^2 &\leq \|X\|_g^2 + (s + \|E_A\|_2 + \|E_B\|_2) \\
&\leq 1.011\|X\|_g^2.
\end{aligned} \tag{4.63}$$

Based on (4.63), we can have

$$\|Y\|_g \leq 1.006\|X\|_g . \tag{4.64}$$

Therefore, we put (4.64) into (4.62) and we can get (4.61). Lemma 4.7 holds. □

Lemma 4.8. *For W in (4.46), we have*

$$\|W\|_2 \leq 1.58, \tag{4.65}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$.

Proof. Based on (4.46), we can have

$$\begin{aligned} \|W\|_2 &\leq \left\|XY^{-1}\right\|_2 + \left\|E_{WY}Y^{-1}\right\|_2 \\ &\leq \left\|XY^{-1}\right\|_2 + \|E_{WY}\|_2 \left\|Y^{-1}\right\|_2 . \end{aligned} \tag{4.66}$$

With (4.7), (4.59) and (4.61), we can have

$$\begin{aligned} \|E_{WY}\|_2 \left\|Y^{-1}\right\|_2 &\leq \frac{1.03\eta n\mathbf{u} \cdot \|W\|_2 \|X\|_c}{\sqrt{(\sigma_{min}(X))^2 + 0.9s}} \\ &\leq \frac{1.03\eta n\mathbf{u}\|X\|_c}{\sqrt{9.9\eta\sqrt{n+1}\mathbf{u}\|X\|_c^2}} \cdot \|W\|_2 \\ &\leq \frac{1.06}{\sqrt{9.9}} \cdot \sqrt{\eta n\sqrt{n}\mathbf{u}} \cdot \|W\|_2 \\ &\leq 0.05\|W\|_2 , \end{aligned} \tag{4.67}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. Therefore, we put (4.60) and (4.67) into (4.66) and we can have

$$\|W\|_2 \leq 1.5 + 0.05\|W\|_2 , \tag{4.68}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. With (4.68), we can have (4.65). Lemma 4.8 holds. □

4.2.4 Proof of Theorem 4.2

In this part, we proof Theorem 4.2 regarding $\kappa_2(X)$ and $\kappa_2(W)$.

Proof. For $\|E_{WY}\|_F = \|WY - X\|_F$, we put (4.65) into (4.61) and we can have

$$\begin{aligned} \|E_{WY}\|_F &= \|WY - X\|_F \\ &\leq 1.03\eta n\mathbf{u} \cdot \|W\|_2 \|X\|_c \\ &\leq 1.66 j_1 \eta \cdot n\mathbf{u}\|X\|_2 , \end{aligned} \tag{4.69}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. Since we have already estimated $\|W\|_2$, we still need to estimate $\sigma_{min}(W)$ in order to evaluate $\kappa_2(X)$. Using Lemma 1.6 and (4.46), we can derive

$$\sigma_{min}(W) \geq \sigma_{min}(XY^{-1}) - \left\|E_{WY}Y^{-1}\right\|_2. \tag{4.70}$$

According to (4.59) and (4.69), we can have

$$\begin{aligned} \left\|E_{WY}Y^{-1}\right\|_2 &\leq \|E_{WY}\|_2 \left\|Y^{-1}\right\|_2 \\ &\leq \frac{1.66\eta \cdot n\mathbf{u}\|X\|_c}{\sqrt{(\sigma_{min}(X))^2 + 0.9s}}, \end{aligned} \tag{4.71}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. Using the same method in [21], we can have

$$\sigma_{min}(XY^{-1}) \geq \frac{\sigma_{min}(X)}{\sqrt{(\sigma_{min}(X))^2 + s}} \cdot 0.9, \tag{4.72}$$

with probability at least $Q(\eta, mn^2) Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. Therefore, we put (4.71) and (4.72) into (4.70) and when $\kappa_2(X) \leq \frac{1}{4.89 j_1 \cdot \eta n\sqrt{n}\mathbf{u}}$, we can have

$$\begin{aligned} \sigma_{min}(W) &\geq \frac{0.9\sigma_{min}(X)}{\sqrt{(\sigma_{min}(X))^2 + s}} - \frac{1.66\eta \cdot n\mathbf{u}\|X\|_c}{\sqrt{(\sigma_{min}(X))^2 + 0.9s}} \\ &\geq \frac{0.9}{\sqrt{(\sigma_{min}(X))^2 + s}} \cdot (\sigma_{min}(X) - \frac{1.66}{0.9\sqrt{0.9}} j_1 \cdot \eta n\mathbf{u}\|X\|_2) \\ &\geq \frac{\sigma_{min}(X)}{2\sqrt{(\sigma_{min}(X))^2 + s}} \\ &= \frac{1}{2\sqrt{1 + t(\kappa_2(X))^2}}, \end{aligned} \tag{4.73}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. Here, $t = \frac{s}{\|X\|_2^2}$. With (4.65) and (4.73), we can have

$$\kappa_2(W) \leq 3.24\sqrt{1 + t(\kappa_2(X))^2},$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. (4.53) is proved. Therefore, Theorem 4.2 holds. □

4.2.5 Proof of Theorem 4.3

In this part, we prove Theorem 4.3.

Proof. When we take

$$s = 11\eta(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u})\|X\|_c^2,$$

we have

$$\begin{aligned} t &= \frac{s}{\|X\|_2^2} \\ &= 11 j_1^2 \cdot \eta(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u}). \end{aligned} \tag{4.74}$$

Similar to the steps in Chapter 2 and Chapter 3, when X is ill-conditioned, *e.g.*, $\kappa_2(X) \geq \mathbf{u}^{-\frac{1}{2}}$, with (4.74), we can have $t(\kappa_2(X))^2 \geq 11j_1^2 \cdot \eta(\sqrt{m}+\sqrt{n+1}) >> 1$. Therefore, we can get

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

With (4.53), it is clear to see that

$$\kappa_2(W) \leq 3.24\sqrt{t} \cdot \kappa_2(X),$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3})$. Based on the results in [21, 68], Chapter 2, Chapter 3 and (4.8), the sufficient condition of $\kappa_2(X)$ satisfies

$$\begin{aligned} \kappa_2(W) &\leq 3.24\sqrt{t} \cdot \kappa_2(X) \\ &\leq \frac{1}{8j_2 \cdot \sqrt{\eta(\sqrt{m}\mathbf{u}+\sqrt{n+1}\mathbf{u})}}, \end{aligned} \tag{4.75}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3})$. When (4.74) is satisfied, based on (4.75), we can have

$$\begin{aligned} \kappa_2(X) &\leq \Phi \\ &= \frac{1}{86j_1j_2 \cdot \eta(\sqrt{m}\mathbf{u}+\sqrt{n+1}\mathbf{u})}, \end{aligned}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3})$. Combining Φ with the required condition for $\kappa_2(X)$ in (4.73), we obtain the requirement for $\kappa_2(X)$ for Shifted CholeskyQR3 with the randomized models as stated in (4.51). Under the condition given in (4.51), we proceed to prove Theorem 4.3. The proof is divided into two parts, orthogonality and residual.

Orthogonality of Shifted CholeskyQR3

First, we consider the orthogonality. For Shifted CholeskyQR3, the same as (4.35) , we can have

$$\kappa_2(V) \leq \sqrt{\frac{69}{59}}, \tag{4.76}$$

with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3}))^2$. Here, the same as (4.9) and the steps in Chapter 2, we can have (4.54) with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3}))^3$.

For the residual, with (4.47)-(4.50), we can have

$$\begin{aligned} QR &= (V + E_7)J^{-1}(JN - E_8) \\ &= (V + E_7)N - (V + E_7)J^{-1}E_8 \\ &= VN + E_7N - QE_8 \\ &= (W + E_3)D^{-1}(DY - E_4) + E_7N - QE_8 \\ &= (W + E_3)Y - (W + E_3)D^{-1}E_4 + E_7N - QE_8 \\ &= WY + E_3Y - VE_4 + E_7N - QE_8. \end{aligned}$$

So it is obvious that

$$\begin{aligned} \|QR - X\|_F \leq &\|WY - X\|_F + \|E_3\|_F \|Y\|_2 + \|V\|_2 \|E_4\|_F \\ &+ \|E_7\|_F \|N\|_2 + \|Q\|_2 \|E_8\|_F . \end{aligned} \tag{4.77}$$

Based on the results in [21], we can have

$$\|Y\|_2 \leq 1.006 \|X\|_2 . \tag{4.78}$$

Similar to (4.23) and (4.39), with (4.52) and (4.65), we can estimate $\|D\|_2$ and $\|D\|_g$ as

$$\begin{aligned} \|D\|_2 &\leq 1.005 \|W\|_2 \\ &\leq 1.59, \end{aligned} \tag{4.79}$$

$$\begin{aligned} \|D\|_g &\leq 1.005 \|W\|_g \\ &\leq \frac{1.59 j_2}{\sqrt{n}}, \end{aligned} \tag{4.80}$$

with probability at least $(Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3})$. The same as (4.33), we can get

$$\|V\|_2 \leq \frac{\sqrt{69}}{8}, \tag{4.81}$$

with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3}))^2$. We follow the steps to get (4.61), together with (4.65) and (4.81), we can bound $\|E_3\|_F$ as

$$\begin{aligned} \|E_3\|_F &\leq 1.03 \eta n \mathbf{u} \cdot \|V\|_2 \|W\|_c \\ &\leq \frac{\sqrt{69n}}{8} \cdot 1.03 j_2 \eta n \mathbf{u} \cdot 1.58 \\ &\leq 1.69 j_2 \cdot \eta n \mathbf{u}, \end{aligned} \tag{4.82}$$

with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3}))^2$. Using Lemma 1.11, (2.3), (4.64) and (4.80), we can estimate $\|E_4\|_F$ and $\|E_4\|_g$ as

$$\begin{aligned}\|E_4\|_F &\leq \tilde{\gamma}_n(\eta)(\|D\|_F \cdot \|Y\|_F) \\ &\leq \tilde{\gamma}_n(\eta)(\sqrt{n}\|D\|_g \cdot \sqrt{n}\|Y\|_g) \\ &\leq 1.02\eta \cdot \sqrt{n}\mathbf{u} \cdot 1.59 j_2 \cdot 1.006 j_1 \|X\|_2 \\ &\leq 1.64 j_1 j_2 \cdot \eta\sqrt{n}\mathbf{u}\|X\|_2 \,,\end{aligned} \tag{4.83}$$

$$\begin{aligned}\|E_4\|_g &\leq \tilde{\gamma}_n(\eta)(\|D\|_F \cdot \|Y\|_g) \\ &\leq \tilde{\gamma}_n(\eta)(\sqrt{n}\|D\|_g \cdot \|Y\|_g) \\ &\leq 1.02\eta \cdot \sqrt{n}\mathbf{u} \cdot 1.59 j_2 \cdot \frac{1.006 j_1}{\sqrt{n}} \cdot \|X\|_2 \\ &\leq 1.64 j_1 j_2 \cdot \eta\mathbf{u}\|X\|_2 \,,\end{aligned} \tag{4.84}$$

with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3}))^2 Q(\eta, n^3)$. Moreover, based on Lemma 2.2, Lemma 2.3, (4.7), (4.64), (4.78), (4.79), (4.83) and (4.84), $\|N\|_2$ and $\|N\|_g$ can be bounded as

$$\begin{aligned}\|N\|_2 &\leq \|D\|_2\|Y\|_2 + \|E_4\|_2 \\ &\leq 1.59 \cdot 1.006\|X\|_2 + 1.64 j_1 j_2 \cdot \eta\sqrt{n}\mathbf{u}\|X\|_2 \\ &\leq 1.63\|X\|_2 \,,\end{aligned} \tag{4.85}$$

$$\begin{aligned}\|N\|_g &\leq \|D\|_2\|Y\|_g + \|E_4\|_g \\ &\leq 1.59 \cdot \frac{1.006 j_1}{\sqrt{n}} \cdot \|X\|_2 + 1.64 j_1 j_2 \cdot \eta\mathbf{u}\|X\|_2 \\ &\leq \frac{1.63 j_1}{\sqrt{n}} \cdot \|X\|_2 \,,\end{aligned} \tag{4.86}$$

with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3}))^2 Q(\eta, n^3)$. Based on (4.52) and (4.54), we can have

$$\|Q\|_2 \leq 1.01, \tag{4.87}$$

with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3}))^3$. Similar to (4.80) and with (4.81), we can get

$$\begin{aligned}\|J\|_g &\leq 1.005\|V\|_g \\ &\leq \frac{1.005}{\sqrt{n}} \cdot j_3,\end{aligned} \tag{4.88}$$

with probability at least $((Q(\eta, mn^2))^2 Q(\eta, \frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3}))^2$. Similar to (4.82), we can bound $\|E_7\|_F$

with (4.81) and (4.87) as

$$\begin{aligned}\|E_7\|_F &\leq 1.03\eta n\mathbf{u}\cdot\|Q\|_2\|V\|_c \\ &\leq 1.03\eta n\mathbf{u}\cdot 1.01\cdot 1.005j_3\cdot\frac{\sqrt{69}}{8} \\ &\leq 1.09j_3\cdot\eta n\mathbf{u},\end{aligned} \tag{4.89}$$

with probability at least $((Q(\eta,mn^2))^2Q(\eta,\frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3}))^3$. Based on Lemma 1.11, (4.86) and (4.88), we can bound $\|E_8\|_F$ as

$$\begin{aligned}\|E_8\|_F &\leq \tilde{\gamma}_n(\eta)(\|J\|_F\cdot\|N\|_F) \\ &\leq \tilde{\gamma}_n(\eta)(\sqrt{n}\|J\|_g\cdot\sqrt{n}\|N\|_g) \\ &\leq 1.02\eta\cdot\sqrt{n}\mathbf{u}\cdot 1.005j_3\cdot 1.63j_1\|X\|_2 \\ &\leq 1.68j_1j_3\cdot\eta\sqrt{n}\mathbf{u}\|X\|_2,\end{aligned} \tag{4.90}$$

with probability at least $((Q(\eta,mn^2))^2Q(\eta,\frac{n^3}{6}+\frac{n^2}{2}+\frac{n}{3}))^3(Q(\eta,n^3))^2$. Therefore, we put (4.69), (4.78), (4.81)-(4.83), (4.85), (4.87), (4.89) and (4.90) into (4.77) and we can have (4.55). Therefore, Theorem 4.3 holds. □

Remark 4.2. *Theorem 4.2 is the key theoretical result of this chapter. The primary advantage of the probabilistic error analysis of Shifted CholeskyQR3 is that it provides a better shifted item s, which can significantly enhance the properties of Shifted CholeskyQR under certain probabilities. We can observe these advantages in Section 4.3. Furthermore, Theorem 4.3 offers improved upper bounds of orthogonality and residual in Shifted CholeskyQR3 compared to those in [21] and Chapter 2.*

Actually, there is another approach for rounding analysis of Shifted CholeskyQR3. We can provide a weaker assumption only for the first Shifted CholeskyQR in this work with the deterministic models for other steps of analysis in this work to achieve a higher probability for the upper bounds theoretically. CholeskyQR2 after Shifted ChoelskyQR can be taken as the idea in Remark 4.1.

4.3 Numerical experiments

In this section, we present several groups of numerical experiments. We primarily focus on the numerical experiments of Shifted CholeskyQR3 conducted with the probabilistic s in this work. We test the numerical stability, the p-values and the robustness of the algorithm in the following. All the experiments are implemented using MATLAB R2022A on our laptop. Specifications of our computer are in Table 2.1.

4.3.1 Applicability and accuracy of Shifted CholeskyQR3 with the probabilistic s

In this section, we focus on the applicability and accuracy of Shifted CholeskyQR3 with different s. We take two different s, that is, the probabilistic s used in this chapter and the improved s in Chapter 2. For the input matrix $X \in \mathbb{R}^{m\times n}$, we primarily focus on the potential influence of $\kappa_2(X)$, m and n. We construct X using SVD, as described in [21, 68] and Chapter 2. The methods in [17, 24] are also applicable. We control $\kappa_2(X)$ through $\sigma_{min}(X)$. We set

$$X = O\Sigma H^\top,$$

where $O \in \mathbb{R}^{m\times m}$, $H \in \mathbb{R}^{n\times n}$ 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 positive constant. Therefore, we can have $\sigma_1(X) = \|X\|_2 = 1$ and $\kappa_2(X) = \frac{1}{\sigma}$. Similar to the setting in Chapter 2, we can build the large $X \in \mathbb{R}^{m\times n}$ using $X_1 \in \mathbb{R}^{n\times n}$ based on SVD as

$$X = \begin{pmatrix} X_1 \\ X_1 \\ \vdots \\ X_1 \end{pmatrix}.$$

To test the influence of $\kappa_2(X)$, we vary $\kappa_2(X)$ while fixing $m = 1024$, $n = 32$ and $\eta = 6$. When varying n, we fix $m = 4096$, $\kappa_2(X) = 10^{12}$ and $\eta = 8$. When varying m, we fix $n = 128$, $\kappa_2(X) = 10^{12}$ and $\eta = 8$. To assess the influence of $\kappa_2(X)$, we compare the accuracy of Shifted CholeskyQR3 using the improved s in Chapter 2. Numerical experiments for a large $X \in \mathbb{R}^{16384\times 1024}$ are taken and X is built in a block version as mentioned above. Here, we take $\eta = 10$. We define the probabilistic s as $s = 11\eta(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u})\|X\|_c^2$ and the improved s as $s = 11\eta(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u})\|X\|_c^2$. The results of the numerical experiments are presented in Table 4.1–Table 4.6.

Table 4.1: Shifted CholeskyQR3 with the probabilistic s for $X \in \mathbb{R}^{1032\times 32}$

$\kappa_2(X)$	10^8	10^{10}	10^{12}	10^{14}	10^{15}
Orthogonality	$1.40e-15$	$1.58e-15$	$1.58e-15$	$1.62e-15$	$1.84e-15$
Residual	$4.00e-16$	$3.95e-16$	$3.30e-16$	$3.20e-16$	$3.20e-16$

Table 4.2: Shifted CholeskyQR3 with the improved s for $X \in \mathbb{R}^{1032\times 32}$

$\kappa_2(X)$	10^8	10^{10}	10^{12}	10^{14}	10^{15}
Orthogonality	$1.30e-15$	$1.69e-15$	$1.38e-15$	$1.49e-15$	–
Residual	$3.72e-16$	$3.52e-16$	$3.55e-16$	$3.28e-16$	–

Table 4.3: Shifted CholeskyQR3 with the probabilistic s for $X \in \mathbb{R}^{16384\times 1024}$

$\kappa_2(X)$	10^6	10^8	10^{10}	10^{12}	10^{13}
Orthogonality	$1.69e-14$	$1.86e-14$	$1.98e-14$	$2.07e-14$	$2.10e-14$
Residual	$2.24e-14$	$2.02e-14$	$1.87e-14$	$1.74e-14$	$1.69e-14$

Table 4.4: Shifted CholeskyQR3 with the improved s for $X \in \mathbb{R}^{16384\times 1024}$

$\kappa_2(X)$	10^6	10^8	10^{10}	10^{12}	10^{13}
Orthogonality	$1.73e-14$	$1.90e-14$	$1.99e-14$	$2.10e-14$	–
Residual	$2.23e-14$	$2.02e-14$	$1.86e-14$	$1.74e-14$	–

Table 4.5: Shifted CholeskyQR3 with the probabilistic s under different n

n	128	256	512	1024	2048
Orthogonality	$2.75e-15$	$4.16e-15$	$8.27e-15$	$1.40e-14$	$2.53e-14$
Residual	$1.07e-15$	$2.00e-15$	$3.08e-15$	$4.35e-15$	$5.81e-15$

Table 4.6: Shifted CholeskyQR3 with the probabilistic s under different m

m	256	512	1024	2048	4096
Orthogonality	$6.08e-15$	$4.39e-15$	$3.82e-15$	$3.03e-15$	$2.75e-15$
Residual	$1.08e-15$	$1.10e-15$	$1.08e-15$	$1.07e-15$	$1.07e-15$

According to Table 4.1 and Table 4.2, we find that both Shifted CholeskyQR3 with the probabilistic s and the improved s are numerically stable in terms of both orthogonality and residual, as indicated by (4.54), (4.55), and the results in Chapter 2. Shifted CholeskyQR3 with the probabilistic s shows better applicability for ill-conditioned matrices. Similar results hold for the large X according to Table 4.3 and Table 4.4. This highlights the significance of probabilistic error analysis of CholeskyQR algorithms. Comparing Table 4.1 and Table 4.2 with Table 4.3 and Table 4.4, we find that the

increasing m and n will decrease the accuracy and applicability of the algorithm, which corresponds to the theoretical results of CholeskyQR-type algorithms. Furthermore, Table 4.5 shows that both m and n do not influence the numerical stability of Shifted CholeskyQR3 with the probabilistic s.

4.3.2 Comparison between the theoretical bounds and real performances

In this part, we make a comparison between the theoretical bounds of Shifted CholeskyQR3 and its real performances with the probabilistic s. Similar to that in Chapter 2, we primarily focus on the accuracy. For the input $X \in \mathbb{R}^{m\times n}$ based on SVD, we fix $\|X\|_2 = 1$ and $\kappa_2(X) = 10^{12}$. We denote $6\eta \cdot j_3^2(\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u})$ in (4.54) as the '*Theoretical bound*' in orthogonality. Moreover, $\phi_1(j_1, j_2, j_3, n)\eta \cdot n\mathbf{u}\|X\|_2$ in (4.55) is the '*Theoretical bound*' in residual. To test the influence of m, we fix $n = 128$ and $\eta = 8$. To test the influence of n, we fix $m = 2048$ and $\eta = 8$. Comparisons of orthogonality and residual with different m and n are shown in Table 4.7-Table 4.10. Regarding the conditions of $\kappa_2(X)$, we denote L in Table 1.8 as the '*Sufficient condition*' of $\kappa_2(X)$ and $\frac{1}{4.89j_1\cdot\eta n\mathbf{u}}$ as the '*Upper bound*' of $\kappa_2(X)$. We vary m and n and comparisons of conditions of $\kappa_2(X)$ are shown in Table 4.11 and Table 4.12.

Table 4.7: Comparison of orthogonality with the probabilistic s when $\kappa_2(X) = 10^{12}$, $n = 128$ and $\eta = 8$

m	256	512	1024	2048	4096
Real error	$6.04e-15$	$4.55e-15$	$3.61e-15$	$3.16e-15$	$2.74e-15$
Theoretical bound	$1.87e-11$	$2.32e-11$	$2.96e-11$	$3.86e-11$	$5.14e-11$

Table 4.8: Comparison of orthogonality with the probabilistic s when $\kappa_2(X) = 10^{12}$, $m = 4096$ and $\eta = 8$

n	128	256	512	1024	2048
Real error	$2.74e-15$	$4.10e-15$	$7.88e-15$	$1.42e-14$	$2.51e-14$
Theoretical bound	$5.14e-11$	$1.09e-10$	$2.36e-10$	$5.24e-10$	$1.19e-09$

Table 4.9: Comparison of residual with the probabilistic s when $\kappa_2(X) = 10^{12}$, $n = 128$ and $\eta = 8$

m	256	512	1024	2048	4096
Real error	$1.08e-15$	$1.08e-15$	$1.05e-15$	$1.09e-15$	$1.08e-15$
Theoretical bound	$6.09e-12$	$6.09e-12$	$6.09e-12$	$6.09e-12$	$6.09e-12$

Table 4.10: Comparison of residual with the probabilistic s when $\kappa_2(X) = 10^{12}$, $m = 4096$ and $\eta = 8$

n	128	256	512	1024	2048
Real error	$1.08e-15$	$2.04e-15$	$3.09e-15$	$4.35e-15$	$5.84e-15$
Theoretical bound	$6.09e-12$	$1.65e-11$	$4.62e-11$	$1.28e-10$	$3.58e-10$

Table 4.11: Comparison of $\kappa_2(X)$ with the probabilistic s when $\kappa_2(X) = 10^{12}$, $n = 128$ and $\eta = 8$

m	256	512	1024	2048	4096
Real case	$\geq 10^{12}$	$\geq 10^{12}$	$\geq 10^{12}$	$\geq 10^{12}$	$\geq 10^{12}$
Upper bound	$6.62e+11$	$6.62e+11$	$6.62e+11$	$6.62e+11$	$6.62e+11$
Sufficient condition	$1.55e+10$	$1.25e+10$	$9.82e+09$	$7.52e+09$	$5.65e+09$

Table 4.12: Comparison of $\kappa_2(X)$ with the probabilistic s when $\kappa_2(X) = 10^{12}$, $m = 4096$ and $\eta = 8$

n	128	256	512	1024	2048
Real case	$\geq 10^{12}$	$\geq 10^{12}$	$\geq 10^{12}$	$\geq 10^{12}$	$\geq 10^{12}$
Upper bound	$6.62e+11$	$2.79e+11$	$1.08e+11$	$3.69e+10$	$1.48e+10$
Sufficient condition	$5.65e+09$	$3.17e+09$	$1.60e+09$	$7.00e+08$	$3.49e+08$

According to Table 4.7-Table 4.12, similar to the results in Chapter 2, we can find that the theoretical results of $\kappa_2(X)$ and accuracy have some distances to the real result after many groups of numerical experiments. However, when comparing Table 4.11 and Table 4.12 with Table 2.18-Table 2.19, we can find that the theoretical bounds of $\kappa_2(X)$ from probabilistic error analysis are closer to the real results than those based on deterministic error analysis, which reflects, to some extent, the advantage of the randomized model for probabilistic error analysis.

4.3.3 Improvements of $\|\cdot\|_c$

In this section, we examine the improvements of $\|\cdot\|_c$ on probabilistic error analysis. Similar to that in Chapter 2, we consider the j-values in probabilistic error analysis of Shifted CholeskyQR3. The same as that in Chapter 2, we take $l_1 = \frac{\|X\|_g}{\|X\|_2}$. Moreover, we define $l_2 = \frac{\|W\|_g}{\|W\|_2}$ and $l_3 = \frac{\|Y\|_g}{\|Y\|_2}$. Here, $l_i = \frac{j_i}{\sqrt{n}}$ and $\frac{1}{\sqrt{n}} \leq l_i \leq 1$, $i = 1, 2, 3$. We construct the input matrix X in the same manner as described in Table 4.3.1. We test the influence of $\kappa_2(X)$, n and m on the l-values. When varying $\kappa_2(X)$, we fix $m = 1024$, $n = 32$ and $\eta = 6$. For varying n, we fix $m = 4096$, $\kappa_2(X) = 10^{12}$ and $\eta = 8$. When varying m, we fix $n = 128$, $\kappa_2(X) = 10^{12}$ and $\eta = 8$. We use $s = 11\eta \cdot (\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u})\|X\|_c^2$ for Shifted CholeskyQR3. The results of the numerical experiments are presented in Table 4.13-Table 4.15.

Table 4.13: l-values with different $\kappa_2(X)$ for Shifted CholeskyQR3

$\kappa_2(X)$	10^8	10^{10}	10^{12}	10^{14}	10^{15}
l_1	0.2590	0.2413	0.2289	0.2225	0.2200
l_2	1.0000	1.0000	1.0000	1.0000	1.0000
l_3	1.0000	1.0000	1.0000	0.9953	0.9978

Table 4.14: l-values with different n for Shifted CholeskyQR3

n	128	256	512	1024	2048
l_1	0.2228	0.2119	0.1899	0.1869	0.1793
l_2	1.0000	1.0000	1.0000	1.0000	1.0000
l_3	1.0000	1.0000	0.9999	0.9998	0.9994

Table 4.15: l-values with different m for Shifted CholeskyQR3

m	256	512	1024	2048	4096
l_1	0.2181	0.2181	0.2181	0.2181	0.2228
l_2	1.0000	1.0000	1.0000	1.0000	1.0000
l_3	1.0000	1.0000	1.0000	1.0000	1.0000

Table 4.13–Table 4.15 show that the l-values are closely related to n. As n increases, both l_1 and l_3 decrease significantly, which aligns with the lower bound of the l-value, $\frac{1}{\sqrt{n}}$. Additionally, l_1 is much smaller than 1. This demonstrates that j_1 is much smaller than $\sqrt{n}$, while j_2 and j_3 tend to be close to $\sqrt{n}$. Such a phenomenon shows the improvement of using $\|X\|_c$ instead of $\|X\|_2$ in s on the applicability of Shifted CholeskyQR3, since $\frac{11\eta(\sqrt{m}\mathbf{u}+\sqrt{n+1}\mathbf{u})\|X\|_c^2}{11\eta(\sqrt{mn}\mathbf{u}+\sqrt{n+1}n\mathbf{u})\|X\|_2^2} = l_1^2 << 1$.

4.3.4 Robustness of Shifted CholeskyQR3 with the probabilistic s

In this section, we demonstrate the robustness of Shifted CholeskyQR3 with the probabilistic s. To the best of our knowledge, this group of experiments has not been conducted in similar works before. We construct the input matrix X in the same manner as described in Table 4.3.1 and examine the potential influence of $\kappa_2(X)$, n and m, which are consistent with those in Table 4.3.3. When varying $\kappa_2(X)$, we fix $m = 1024$, $n = 32$ and $\eta = 6$. When varying n, we fix $m = 4096$, $\kappa_2(X) = 10^{12}$ and $\eta = 8$. When varying m, we fix $n = 128$, $\kappa_2(X) = 10^{12}$ and $\eta = 8$. We use $s = 11\eta \cdot (\sqrt{m}\mathbf{u} + \sqrt{n+1}\mathbf{u})\|X\|_c^2$ for Shifted CholeskyQR3. We record the number of successful outcomes every 30 trials after conducting

several groups and calculate the average. The numerical results are listed in Table 4.16-Table 4.18.

Table 4.16: Times of success with different $\kappa_2(X)$ for Shifted CholeskyQR3

$\kappa_2(X)$	10^8	10^{10}	10^{12}	10^{14}	10^{15}
Times	30	30	30	30	30

Table 4.17: Times of success with different n for Shifted CholeskyQR3

n	128	256	512	1024	2048
Times	30	30	30	30	30

Table 4.18: Times of success with different m for Shifted CholeskyQR3

m	256	512	1024	2048	4096
Times	30	30	30	30	30

Table 4.16-Table4.18 demonstrate that Shifted CholeskyQR3 with the probabilistic s exhibits strong robustness in our numerical examples, which is crucial for the practical application of this improved algorithm.

4.4 Conclusions

In this chapter, we do probabilistic error analysis of Shifted CholeskyQR3 and CholeskyQR2. The new matrix $\|\cdot\|_c$ is utilized. We receive tighter upper bounds of orthogonality and residual for the algorithms and a probabilistic shifted item s for Shifted CholekskyQR3. Numerical experiments show the improvement on applicability of such a probabilistic s and its robustness.

CHAPTER 5.
CONCLUSIONS AND FUTURE WORKS

This thesis presents several improvements on CholeskyQR-type algorithms from different perspectives and with different tools, including the improved shifted item s for Shifted CholeskyQR3 based on $\|\cdot\|_c$, an analysis of Shifted CholeskyQR for sparse matrices and rounding error analysis of CholeskyQR-type algorithms with the randomized model partially. The primary target of these works is to enhancing the applicability of CholeskyQR-type algorithms by improving rounding error analysis. Complete and rigorous theoretical proofs, along with the corresponding numerical experiments, are contained in Chapter 2-Chapter 4.

Chapter 2 focuses on improving the shifted item s for Shifted CholeskyQR3. We introduce a new matrix norm $\|\cdot\|_c$ and propose an improved shifted item s with $\|X\|_c$ for the input matrix $X \in \mathbb{R}^{m\times n}$. Our theoretical analysis and numerical experiments show that such an improved s can guarantee numerical stability and efficiency of Shifted CholeskyQR3, along with better applicability compared to the case with the original s based on $\|X\|_2$. In Chapter 3, we focus on Shifted CholeskyQR for sparse matrices. We introduce a new model for the division of sparse matrices based on the presence of dense columns. Therefore, an alternative choice of s based on the structure and the key element of X can be taken for Shifted CholeskyQR3. We prove that such an alternative choice s can guarantee numerical stability of Shifted CholeskyQR3 with certain element-norm conditions (ENCs), under which our alternative s is optimal compared to s proposed in Chapter 2. Numerical experiments show that our alternative s can improve the applicability of Shifted CholeskyQR3 in sparse cases while maintaining numerical stability and efficiency. In Chapter 4, we present rounding error analysis of CholeskyQR-type algorithms with the randomized model partially under a weak assumption. We receive the improved sufficient condition of $\kappa_2(X)$ for CholeskyQR2 and the best shifted item s for Shifted CholeskyQR3 as far as we know with the randomized model. Shifted CholeskyQR3 with such a probabilistic s is robust after numerous experiments and can still keep numerical stability.

Compared to other algorithms for QR factorization, CholeskyQR strikes a balance between accuracy and efficiency, making it more suitable for parallel computing. In this thesis, we explore a new perspective on CholeskyQR, defining a new $\|X\|_c$ and build connections between CholeskyQR and randomized models. These strategies improve the applicability of these algorithms in both general and special scenarios. Our contributions extend the properties of CholeskyQR-type algorithms to many real-world applications and address several issues in this field, which are significant for the advancement of QR factorization. But the works contained this thesis are not all the works we have done

in the past several years. In fact, the most fragile step of CholeskyQR lies in Cholesky factorization to calculate the R-factor. Except for Shifted CholeskyQR3 to address the problem of applicability, there is another way deserving consideration for CholeskyQR-type algorithms, that is, the mixed preconditioning step with CholeskyQR and other types of algorithms, which is often taken with some randomized techniques. In [3, 20, 31], some researchers tend to use some other structures to replace the steps of calculating the gram matrix and Cholesky factorization in CholeskyQR to generate the R-factor, in order to avoid the numerical breakdown of Cholesky factorization in ill-conditioned cases. Some randomized techniques, such as matrix sketching, are used to accelerate the algorithms. Among all the CholeskyQR-type algorithms, LU-CholeskyQR2 [62] has no requirement on $\kappa_2(X)$ for the input $X \in \mathbb{R}^{m\times n}$, which is a very special but important advantage compared to other algorithms. In [28], we improve LU-CholeskyQR2 by combining LU-CholeskyQR with HouseholderQR and form LHC2. We utilize the recent matrix sketching to accelerate LHC2 and form SLHC3 and SSLHC3. Such new algorithms do not have requirements on $\kappa_2(X)$ for the input X and are very balanced in accuracy, applicability, efficiency and robustness, which are the top CholeskyQR-type algorithms in the real performance.

During our research, we identify several topics for future exploration. We have ongoing and potential projects related to some of these areas. Below is a list of these topics.

1. In Chapter 2, we introduce a new matrix $\|\cdot\|_c$ and demonstrate some of its properties. However, several issues remain to be addressed in the future. Specifically, $\|\cdot\|_c$ of a matrix warrants further exploration. Developing efficient methods to quickly estimate $\|X\|_c$ for the input matrix X is an open topic for future research, particularly for large-scale matrices. Additionally, the properties of calculating $\|\cdot\|_c$ suggests that parallel computing can be employed to obtain $\|\cdot\|_c$ more efficiently. In this thesis, we leverage the connections between $\|\cdot\|_c$ and some other matrix norms to do rounding error analysis. Given that $\|\cdot\|_c$ can be applied to various problems, such as HouseholderQR and Nyström approximation, we aim to explore its relationship with the singular values of matrices and other factors, such as the condition number. We are also focusing on additional properties related to $\|\cdot\|_c$.

2. In Chapter 3, we introduced a new model for dividing sparse matrices into two types based on the presence of dense columns and provided a detailed rounding error analysis of Shifted CholeskyQR3 for sparse matrices. We are curious about whether we can do improvements on this model and provide more accurate error analysis of CholeskyQR-type algorithms for sparse matrices. We are also interested in exploring whether other algorithms for QR factorization, such as HouseholderQR and Modified Gram-Schmidt (MGS), can benefit from our framework or alternative types of models. Focusing on sparse matrices is particularly meaningful, as they are

common in real-world applications. We aim to improve HouseholderQR and MGS by addressing their drawbacks and designing more accurate and efficient methods.

3. In Chapter 4, we utilize the randomized model to do improved analysis of CholeskyQR-type algorithms. According to Lemma 1.10-Lemma 1.13, we observe that the existing randomized models has very strict conditions to be applied, which is not friendly towards CholeskyQR-type algorithms with several different steps of computation. Therefore, in Chapter 4, we only use the randomized model of matrix multiplications in the first step of CholeskyQR, which can only improve the applicability of Shifted CholeskyQR3 and the sufficient condition of CholeskyQR2. In order to provide probabilistic error bounds of CholeskyQR-type algorithms, it is meaningful for us to provide better randomized models of rounding error analysis with weaker conditions and larger probabilities, which can provide more tools for rounding error analysis and matrix perturbations.

4. Among all the CholeskyQR-type algorithms, a key sufficient condition is that the input matrix should be full-rank, which is closely related to the singular values of the input matrix. For the tall skinny matrix $X \in \mathbb{R}^{m\times n}$ with $m \geq n$, it means that rank(X)=n. Additionally, many of the existing Cholesky-type algorithms encounter problems regarding the sufficient condition of $\kappa_2(X)$ of the input matrix X, limiting their practical use in the real applications. To address these challenges, we are exploring new preconditioning steps based on singular value decomposition(SVD) for Shifted CholeskyQR3, which can deal with rank-deficient and ill-conditioned cases. Such an operator aims to strike a balance between speed, accuracy and applicability, thereby enhancing the performance of CholeskyQR-type algorithms.

5. In recent years, problems concerning Quaternion matrices [39, 43, 46, 65] have attracted the attention of many researchers in numerical linear algebra. This area has applications in image processing and signal processing. As one of the most important challenges in numerical linear algebra, QR factorization of Quaternion matrices [40, 41, 44, 64] warrants further exploration. We aim to combine CholeskyQR with Quaternion matrices and design new algorithms based on this integration. Additionally, more theoretical results and properties of CholeskyQR remain to be discovered.

6. Our work in this thesis are primarily implemented on CPU, so do most of the CholeskyQR-type algorithms. In recent years, GPU has played an important role in high-performance computing. We always need to consider the version of many algorithms for parallel computing. There is an existing work [70] regarding CholeskyQR and its analysis on GPU. However, it only focuses on the basic version of CholeskyQR. We are currently focusing on constructing new CholeskyQR-

type algorithms which is suitable for parallel computing. We hope that it can make use of the advantages of GPU.

Bibliography

[1] D. Achlioptas. Database-friendly random projections. In *Proceedings of the Twentieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, PODS '01, page 274–281, New York, NY, USA, 2001. Association for Computing Machinery.

[2] Y. Aizenbud, G. Shabat, and A. Averbuch. Randomized LU decomposition using sparse projections. *Computers and Mathematics with Applications*, 72:2525–2534, 2016.

[3] O. Balabanov. Randomized CholeskyQR factorizations. *arxiv preprint arXiv:2210.09953*, 2022.

[4] O. Balabanov and L. Grigori. Randomized Gram–Schmidt Process with Application to GMRES. *SIAM Journal on Scientific Computing*, 44(3):A1450–A1474, 2022.

[5] G. Ballard, J. Demmel, O. Holtz, and O. Schwartz. Minimizing communication in numerical linear algebra. *SIAM Journal on Matrix Analysis and Applications*, 32(3):866–901, 2011.

[6] B. Beckermann. The condition number of real Vandermonde, Krylov and positive definite Hankel matrices. *Numerische Mathematik*, 85:553–577, 2000.

[7] A. Borobia. Constructing matrices with prescribed main-diagonal submatrix and characteristic polynomial. *Linear Algebra and its Applications*, 418:886–890, 2006.

[8] Yifan Chen, Ethan N. Epperly, Joel A. Tropp, and Robert J. Webber. Randomly pivoted cholesky: Practical approximation of a kernel matrix with few entry evaluations. *Communications on Pure and Applied Mathematics*, 78(5):995–1041, 2025.

[9] M. Choi. Tricks or Treats with the Hilbert Matrix. *The American Mathematical Monthly*, 90(5):301–312, 1983.

[10] M.P. Connolly and N.J. Higham. Probabilistic Rounding Error Analysis of Householder QR Factorization. *SIAM Journal on Matrix Analysis and Applications*, 44:1146–1163, 2023.

[11] M.P. Connolly, N.J. Higham, and T. Mary. Stochastic Rounding and its Probabilistic Backward Error Analysis. *SIAM Journal on Scientific Computing*, 43:566–585, 2021.

[12] P.G. Constantine and D.F. Gleich. Tall and skinny QR factorizations in MapReduce architectures. In *Proceedings of the second international workshop on MapReduce and its applications*, pages 43–50, 2011.

[13] T.A. Davis and W.W. Hager. Modifying a sparse Cholesky factorization. *SIAM Journal on Matrix Analysis and Applications*, 20:606–627, 1999.

[14] T.A. Davis and W.W. Hager. Row modifications of a sparse Cholesky factorization. *SIAM Journal on Matrix Analysis and Applications*, 26:621–639, 2005.

[15] J. Demmel. On floating point errors in Cholesky. *Tech.Report 14, LAPACK working Note*, 1989.

[16] J. Demmel, L. Grigori, M. Hoemmen, and J. Langou. 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.

[17] Z. Drmač and K. Veselić. New fast and accurate Jacobi SVD algorithm II. *SIAM Journal on Matrix Analysis and Applications*, 29:1343–1362, 2008.

[18] J.A. Duersch, M. Shao, C. Yang, and M. Gu. A robust and efficient implementation of LOBPCG. *SIAM Journal on Scientific Computing*, 40(5):C655–C676, 2018.

[19] Y. Fan, H. Guan, and Z. Qiao. An Improved Shifted CholeskyQR Based on Columns. *Journal of Scientific Computing*, 104(86), 2025.

[20] Y. Fan, Y. Guo, and T. Lin. A Novel Randomized XR-Based Preconditioned CholeskyQR Algorithm. *arxiv preprint arXiv:2111.11148*, 2021.

[21] T. Fukaya, R. Kannan, Y. Nakatsukasa, Y. Yamamoto, and Y. Yanagisawa. Shifted Cholesky QR for computing the QR factorization of ill-conditioned matrices. *SIAM Journal on Scientific Computing*, 42(1):A477–A503, 2020.

[22] T. Fukaya, Y. Nakatsukasa, Y. Yanagisawa, and Y. Yamamoto. 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.

[23] J. Gao, W. Ji, F. Chang, S. Han, B. Wei, Z. Liu, and Y. Wang. A systematic survey of General Sparse Matrix-matrix Multiplication. *ACM Computing Surveys*, 55:1–36, 2023.

[24] W. Gao, Y. Ma, and M. Shao. A Mixed Precision Jacobi SVD Algorithm. *ACM Transactions on Mathematical Software*, 51(1), April 2025.

[25] G.H. Golub and C.F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, 4th edition, 2013.

[26] Ming Gu and Stanley C. Eisenstat. Efficient algorithms for computing a strong rank-revealing qr factorization. *SIAM Journal on Scientific Computing*, 17(4):848–869, 1996.

[27] H. Guan and Y. Fan. An improved error analysis of CholeskyQR with the randomized model. *arxiv preprint arXiv:2410.09389*, 2024.

[28] H. Guan and Y. Fan. Deterministic and randomized LU-Householder CholeskyQR. *arxiv preprint arXiv:2412.06551*, 2024.

[29] H. Guan and Y. Fan. Shifted CholeskyQR for sparse matrices. *arxiv preprint arXiv:2410.06525*, 2024.

[30] N. Halko, P.G. Martinsson, and J.A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.

[31] A. Higgins, D. Szyld, E. Boman, and I. Yamazaki. Analysis of Randomized Householder-Cholesky QR Factorization with Multisketching. *arxiv preprint arXiv:2309.05868*, 2023.

[32] N. J. Higham. *Accuracy and Stability of Numerical Algorithms*. SIAM, Philadelphia, PA, USA, second ed. edition, 2002.

[33] N.J. Higham and T. Mary. A New Approach to Probabilistic Rounding Error Analysis. *SIAM Journal on Scientific Computing*, 41:2815–2835, 2019.

[34] D. Hilbert. Ein Beitrag zur Theorie des Legendre'schen Polynoms. *Acta Mathematica*, 18(none):155 – 159, 1900.

[35] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58:13–30, 1963.

[36] M. Hoemmen. *Communication-avoiding Krylov subspace methods*. University of California, Berkeley, 2010.

[37] I.C.F. Ipsen and H. Zhou. Probabilistic Error Analysis for Inner Products. *SIAM Journal on Matrix Analysis and Applications*, 41(4):1726–1741, 2020.

[38] C.P. Jeannerod and S.M. Rump. Improved error bounds for inner products in floating-point arithmetic. *SIAM Journal on Matrix Analysis and Applications*, 34:338–344, 2013.

[39] Z. Jia and Michael K. Ng. Structure Preserving Quaternion Generalized Minimal Residual Method. *SIAM Journal on Matrix Analysis and Applications*, 42:616–634, 2021.

[40] Z. Jia, Michael K. Ng, and G. Song. Lanczos method for large-scale quaternion singular value decomposition. *Numerical Algorithms*, 82(2):699–717, Nov 2018.

[41] Z. Jia, M. Wei, M. Zhao, and Y. Chen. A new real structure-preserving quaternion QR algorithm. *Journal of Computational and Applied Mathematics*, 343:26–48, 2018.

[42] M. Kapralov, V. Potluru, and D. Woodruff. How to Fake Multiply by a Gaussian Matrix. In Balcan, Maria Florina and Weinberger, Kilian Q., editor, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 2101–2110, New York, New York, USA, 20–22 Jun 2016. PMLR.

[43] T. Li and Q. Wang. Structure Preserving Quaternion Biconjugate Gradient Method. *SIAM Journal on Matrix Analysis and Applications*, 45(1):306–326, 2024.

[44] Y. Li, M. Wei, F. Zhang, and Zhao J. Real structure-preserving algorithms of Householder based transformations for quaternion matrices. *Journal of Computational and Applied Mathematics*, 305:82–91, 2016.

[45] Z. Li, Y. Wang, and S. Li. 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.

[46] R. Ma, Z. Jia, and Z. Bai. A structure-preserving Jacobi algorithm for quaternion Hermitian eigenvalue problems. *Computers and Mathematics with Applications*, 75(3):809–820, 2018.

[47] P.G. Martinsson and J.A. Tropp. Randomized numerical linear algebra: Foundations and algorithms. *Acta Numerica*, 29:403 – 572, 2020.

[48] D.P. O'Leary and G.W. Stewart. Computing the eigenvalues and eigenvectors of symmetric arrowhead matrices. *Journal of Computational Physics*, 90(2):497–505, 1990.

[49] J. Peng, X. Hu, and L. Zhang. Two inverse eigenvalue problems for a special kind of matrices. *Linear Algebra and its Applications*, 416:336–347, 2006.

[50] M. Rozložník, M. Tůma, A. Smoktunowicz, and J. Kopal. Numerical stability of orthogonalization methods with a non-standard inner product. *BIT Numerical Mathematics*, pages 1–24, 2012.

[51] S.M. Rump and C.P. Jeannerod. Improved backward error bounds for LU and Cholesky factorization. *SIAM Journal on Matrix Analysis and Applications*, 35:684–698, 2014.

[52] S.M. Rump and T. Ogita. Super-fast vallidated solution of linear systems. *Journal of Computational and Applied Mathematics*, 199:199–206, 2007.

[53] R. Schreiber and C. Van Loan. A storage-efficient WY representation for products of Householder transformations. *SIAM Journal on Scientific and Statistical Computing*, 10(1):53–57, 1989.

[54] J. Scott. *Algorithms for Sparse Linear Systems*. Springer International Publishing, New York, 1st ed. edition, 2023.

[55] A. Smoktunowicz, J.L. Barlow, and J. Langou. A note on the error analysis of classical Gram–Schmidt. *Numerische Mathematik*, 105(2):299–313, Nov 2006.

[56] A. Sobczyk and E. Gallopoulos. Estimating Leverage Scores via Rank Revealing Methods and Randomization. *SIAM Journal on Matrix Analysis and Applications*, 42(3):1199–1228, 2021.

[57] A. Sobczyk and E. Gallopoulos. pylspack: Parallel Algorithms and Data Structures for Sketching, Column Subset Selection, Regression, and Leverage Scores. *ACM Transactions on Mathematical Software*, 48(4), December 2022.

[58] G.W. Stewart and J. Sun. *Matrix perturbation theory*. Academic Press, San Diego, CA, USA, sixth ed. edition, 1990.

[59] J. Stoer and R. Bulirsch. *Introduction to numerical analysis*. Springer, New York, 3rd ed. edition, 2002.

[60] N.J. Stor, I. Slapničar, and J.L. Barlow. Accurate eigenvalue decomposition of real symmetric arrowhead matrices and applications. *Linear Algebra and its Applications*, 464:62–89, 2015. Special issue on eigenvalue problems.

[61] Thomas Strohmer and Roman Vershynin. A randomized Kaczmarz algorithm with exponential convergence. *J. Fourier Anal. Appl.*, 15(2):262–278, 2009.

[62] T. Terao, K. Ozaki, and T. Ogita. LU-Cholesky QR algorithms for thin QR decomposition. *Parallel Computing*, 92:102571, 2020.

[63] R.P. Tewarson. *Sparse matrices*. Academic Press, 1973.

[64] M. Wang and W. Ma. A structure-preserving method for the quaternion LU decomposition in quaternionic quantum theory. *Computer Physics Communications*, 184(9):2182–2186, 2013.

[65] Q. Wang, Z. He, and Y. Zhang. Constrained two-sided coupled Sylvester-type quaternion matrix equations. *Automatica*, 101:207–213, 2019.

[66] J.H. Wilkinson. Error analysis of direct methods of matrix inversion. *Journal of the ACM*, 8:281–330, 1961.

[67] J.H. Wilkinson. *Rounding Errors in Algebraic Processes.* Prentice-Hall, Englewood Cliffs, NJ, 1963.

[68] Y. Yamamoto, Y. Nakatsukasa, Y. Yanagisawa, and T. Fukaya. Roundoff error analysis of the CholeskyQR2 algorithm. *Electronic Transactions on Numerical Analysis*, 44(01), 2015.

[69] Y. Yamamoto, Y. Nakatsukasa, Y. Yanagisawa, and T. Fukaya. Roundoff error analysis of the CholeskyQR2 algorithm in an oblique inner product. *JSIAM Letters*, 8:5–8, 2016.

[70] I. Yamasaki, S. Tomov, and J. Dongarra. Mixed-precision Cholesky QR factorization and its case studies on Multicore CPU with Multiple GPUs. *SIAM Journal on Scientific Computing*, 37:C307–C330, 2015.

[71] Y. Yanagisawa, T. Ogita, and S. Oishi. A modified algorithm for accurate inverse Cholesky factorization. *Nonlinear Theory and Its Applications, IEICE*, 5:35–46, 2014.

[72] S.N. Yeralan, T.A. Davis, W.M. Sid-Lakhdar, and S. Ranka. Algorithm 980:Sparse QR Factorization on the GPU. *ACM Transactions on Mathematical Software*, 44:1–29, 2017.

[73] R. Yuster and U. Zwick. Fast sparse matrix multiplication. *ACM Transactions on Algorithms*, 1:2–13, 2005.

[74] Q. Zou. Probabilistic Rounding Error Analysis of Modified Gram–Schmidt. *SIAM Journal on Matrix Analysis and Applications*, 45:1076–1088, 2024.