RANDOMIZED QUATERNION SINGULAR VALUE DECOMPOSITION FOR LOW-RANK MATRIX APPROXIMATION *

QIAOHUA LIU†, SITAO LING‡, AND ZHIGANG JIA§

Dedicated to Professor Musheng Wei on the occasion of his 75th birthday

Abstract. This paper presents a randomized quaternion singular value decomposition (QSVD) algorithm for low-rank matrix approximation problems, which are widely used in color face recognition, video compression, and signal processing problems. With quaternion normal distribution-based random sampling, the randomized QSVD algorithm projects a high-dimensional data to a low-dimensional subspace and then identifies an approximate range subspace of the quaternion matrix. The key statistical properties of quaternion Wishart distribution are proposed and used to perform the approximation error analysis of the algorithm. Theoretical results show that the randomized QSVD algorithm can trace dominant singular value decomposition triplets of a quaternion matrix with acceptable accuracy. Numerical experiments also indicate the rationality of proposed theories. Applied to color face recognition problems, the randomized QSVD algorithm obtains higher recognition accuracies and behaves more efficient than the known Lanczos-based partial QSVD and a quaternion version of fast frequent directions algorithm.

Key words. randomized quaternion SVD; quaternion Wishart distribution; low-rank approximation; error analysis.

AMS subject classifications. 68W20, 60B20, 15A18

1. Introduction.

Low-rank approximations of quaternion matrices play an important role in color image processing area [16, 17], in which color images are represented by pure quaternion matrices. Based on the color principal component analysis [43], the optimal rank-k approximations preserve the main features and the important low frequency information of original color image samples. The core work of generating low-rank approximations is to compute the dominant quaternion singular value decomposition (QSVD) triplets (i.e., left singular vectors, singular values and right singular vectors). However, there are still few efficient algorithms to do this work when quaternion matrices are of large-scale sizes. No rigorous error analysis of computed approximations have also been given in the literature. In this paper, we present a new randomized QSVD algorithm and propose important theoretical results about the feasibility and the reliability of the algorithm.

In these years, quaternions [12] and quaternion matrices [41] have been more and more attractive in many research fields such as signal processing [6], image data analysis [2, 19], and machine learning [28, 43]. Because of non-commutative multiplication of quaternions, quaternion matrix computations contain more abundant challenging topics than real or complex matrix computations. The algorithms designed for quaternion matrices are also feasible for the real or complex case, but the converse is not always true. As we are concerned on, QSVD triplets can be achieved in three totally different ways. The first one is to call the svd command from Quaternion toolbox for Matlab (QTFM) developed by Sangwine and Bihan in 2005. For the principle of the algorithm, we refer to [32]. The codes in QTFM are based on quaternion arithmetic operations and

* Submitted to the editors DATE.

Funding: The work of the authors is partially supported by National Natural Science Foundation of China grants 12171210, 12090011 and 11771188; the Major Projects of Universities in Jiangsu Province (No. 21KJA110001); the Priority Academic Program Development Project (PAPD); the Top-notch Academic Programs Project (No. PPZY2015A013) of Jiangsu Higher Education Institutions.

†Department of Mathematics, Shanghai University, Shanghai 200444, People's Republic of China (qhliu@shu.edu.cn).

‡ School of Mathematics, China University of Mining and Technology, Xuzhou, Jiangsu, 221116, People's Republic of China (lingsitao2004@163.com).

§Corresponding author. School of Mathematics and Statistics and Research Institute of Mathematical Science, Jiangsu Normal University, Xuzhou 221116, People's Republic of China (zhgjia@jsnu.edu.cn).

is less efficient for large matrices. The second one is to use the real structure-preserving QSVD method [38]. Its main idea is to perform real operations on the real counterparts of quaternion matrices with structure preserving scheme. In practical implementations, only the first block row or column of the real counterpart is explicitly stored and updated, and the other subblocks are implicitly formulated with the aid of the algebraic symmetry structure. The real matrix-matrix multiplication-based BLAS-3 operations make the computation more efficient. The concept of structure-preserving was firstly proposed to solve quaternion eigenvalue problem in [13], and then extended to the computations of quaternion LU [22, 37] and QR [21] factorizations. Recently, Jia et al. [14] developed a new structure-preserving quaternion QR algorithm for eigenvalue problems of general quaternion matrices, by constructing feasible frameworks of calculation for new quaternion Householder reflections and generalized Givens transformations. For more issues about structure-preserving algorithms, we refer to two monographs [38] by Wei et al. and [18] by Jia. The above two ways are based on the truncation of the full QSVD and the computational cost is expensive in computing all singular values and corresponding left and right singular vectors. Thus they are not feasible for large-scale quaternion matrices. Jia et al. [15] proposed a promising iterative algorithm to compute dominant QSVD triplets, based on the Lanczos bidiagonalization [8] with reorthogonalization and thick-restart techniques. This method is referred to as the lansvdQ method. The superiority of lansvdQ method over the full QSVD was revealed in [15], through a number of practical applications such as color face recognition, video compression and color image completion. When the target rank k increases, the matrix-vector products at each iteration of lansvdQ make the computational cost increase. Is there any method with lower computational cost for the quaternion low-rank approximation problem?

In the past decade, randomized algorithms for computing approximations of real matrices have been receiving more and more attention. Randomized projection and randomized sampling are two commonly used techniques to deal with large-scale problems efficiently. Randomized projection combines rows or columns together to produce a small sketch of $M \in \mathbb{R}^{m\times n}(m \geq n)$ [33]. Possible techniques include subspace iterations [10], subspace embedding (SpEmb) [27], frequent directions (FD) [7] and etc. Recently, Teng and Chu [34] implanted SpEmb in FD to develop a fast frequent direction (SpFD) algorithm. Through the experimental results on world datasets and applications in network analysis, the superiority of SpFD over FD is displayed, not only in the efficiency, but also in the effectiveness.

Randomized sampling finds a small subset of rows or columns based on a pre-assigned probability distribution, say, by pre-multiplying M on an $n \times \ell$ ($\ell \ll n$) random Gaussian matrix Ω, and identifies a low-dimensional approximate range subspace of M, after which a small-size matrix approximation is also obtained. The idea of a randomized sampling procedure can be traced to a 2006 technical report of paper [26], and later analyzed and elaborated in [5,10,11,25,31,34,40,42]. They are computationally efficient for large-scale problems and adapt to the case that the numerical rank is known or can be estimated in advance. When the singular values have relatively fast decay rate, the algorithm is inherently stable. For singular values with slow decay rate, the randomized algorithm with power scheme will enhance the stability of the algorithm.

In this paper we consider the randomized sampling algorithm for quaternion low-rank matrix approximations. The targeted randomized QSVD algorithm is expected to have lower computational cost and to be appropriate for choosing a small number of dominant QSVD triplets of large-scale quaternion matrices. It seems natural to utilize the research framework in [11] and generalize the real randomized SVD algorithm to quaternion matrices. Unfortunately, the theoretical analysis is long and arduous. It involves doses of statistics related to quaternion variables and several difficulties block us to go further.

- What kind of quaternion distribution is appropriate for the randomized QSVD algorithm? The proper quaternion distribution should be invariant under unitary transformations, which will bring convenience for approximation error analysis of the proposed algorithm.

However, few studies have been seen on the probability distribution of quaternion variables in the literature.

- What are the distributions of the norms of the pseudoinverse $\mathbf{\Omega}^\dagger$ of quaternion random Gaussian matrix $\mathbf{\Omega}$? Due to the non-commutative multiplication of quaternions, quaternion determinant and integrals could not be defined similar to the real case. Hence, real probability theories could not be directly used to evaluate the norms of quaternion random Gaussian matrices.
- What are statistical evaluations of spectral norms of $\mathbf{\Omega}$ and its real counterpart? The real counter part $\Upsilon_{\mathbf{\Omega}}$ (see (2.1)) is a non-Gaussian random matrix. It is necessary to develop novel techniques to evaluate the expectation and probability bounds of $\|\mathbf{\Omega}\|_2$ and its scaled norms.

Based on the investigations on key features of $\mathbf{\Omega}$, we will give expectation and deviation bounds for approximation errors of the quaternion randomized SVD algorithm. To the best of our knowledge, these results are new and no developments have been made on the proposed algorithm and theories about quaternion matrix approximation problems. With high probability, the theoretical results show that the low rank approximations can be computed quickly for quaternion matrices with rapidly decaying singular values. Through the numerical experiments, the superiority of the proposed algorithm will be displayed, in comparison with the quaternion Lanczos method and a quaternion version of SpFD [34].

The paper is organized as follows. In Section 2, we review some preliminary results about quaternion matrices and randomized SVD for real matrices. The randomized QSVD algorithm and implement details for low-rank approximation problems will be studied in Section 3. In Section 4, the theoretical analysis is provided for the approximation errors. In Section 5, we test the theories and numerical behaviors of the proposed algorithms through several experiments and show their efficiency over Lanczos-based partial QSVD algorithm and quaternion SpFD for color face recognition problems.

Throughout this paper, we denote by $\mathbb{R}^{m\times n}$ and $\mathbb{Q}^{m\times n}$ the spaces of all $m\times n$ real and quaternion matrices, respectively. The norm $\|\cdot\|_a$ denotes either the spectral norm or the Frobenius norm. For quaternion matrix $\mathbf{A}\in\mathbb{Q}^{m\times n}$, $\mathbf{A}^\dagger$ is the pseudoinverse of $\mathbf{A}$, and $\mathcal{R}(\mathbf{A})$ represents the column range space of $\mathbf{A}$. $\mathrm{tr}(\cdot)$ denotes the trace of a quaternion or real square matrix, and $\mathrm{etr}(\cdot)$=$\exp(\mathrm{tr}(\cdot))$ means the exponential operation of the trace. Let $\mathsf{P}\{\cdot\}$ denote the probability of an event and $\mathsf{E}(\cdot)$ denote the expectation of a random variable. For differentials $\mathrm{d}y_1, \mathrm{d}y_2$ of real random variables y_1, y_2, $\mathrm{d}y_1\wedge\mathrm{d}y_2$ denotes the non-commutative exterior product of $\mathrm{d}y_1, \mathrm{d}y_2$, under which $\mathrm{d}y_1\wedge\mathrm{d}y_2 = -\mathrm{d}y_2\wedge\mathrm{d}y_1$ and $\mathrm{d}y_1\wedge\mathrm{d}y_1 = 0$.

2. Preliminaries.

In this section, we first introduce some basic information of quaternion matrices and quaternion SVD. The basic randomized SVD for real matrices is described thereafter.

2.1. Quaternion matrix and QSVD.

The quaternion skew-field $\mathbb{Q}$ is an associative but non-commutative algebra of rank four over $\mathbb{R}$, and any quaternion $\mathbf{q}\in\mathbb{Q}$ has one real part and three imaginary parts given by $\mathbf{q} = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$, where $q_0, q_1, q_2, q_3\in\mathbb{R}$, and $\mathbf{i}$, $\mathbf{j}$ and $\mathbf{k}$ are three imaginary units satisfying $\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1$. The conjugate and modulus of $\mathbf{q}$ are defined by $\mathbf{q}^* = q_0 - q_1\mathbf{i} - q_2\mathbf{j} - q_3\mathbf{k}$ and $|\mathbf{q}| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}$, respectively.

For any quaternion matrices $\mathbf{P} = P_0 + P_1\mathbf{i} + P_2\mathbf{j} + P_3\mathbf{k}\in\mathbb{Q}^{m\times n}$, $\mathbf{Q} = Q_0 + Q_1\mathbf{i} + Q_2\mathbf{j} + Q_3\mathbf{k}\in\mathbb{Q}^{m\times n}$, denote $\mathbf{Q}^* = Q_0^T - Q_1^T\mathbf{i} - Q_2^T\mathbf{j} - Q_3^T\mathbf{k}$ and the sum of $\mathbf{P}, \mathbf{Q}$ as $\mathbf{P}+\mathbf{Q} = (P_0+Q_0) + (P_1+Q_1)\mathbf{i} + (P_2+Q_2)\mathbf{j} + (P_3+Q_3)\mathbf{k}$, and for quaternion matrix $\mathbf{S}\in\mathbb{Q}^{n\times\ell}$, the multiplication $\mathbf{QS}$ is given by

$$(Q_0S_0 - Q_1S_1 - Q_2S_2 - Q_3S_3) + (Q_0S_1 + Q_1S_0 + Q_2S_3 - Q_3S_2)\mathbf{i} +$$
$$(Q_0S_2 - Q_1S_3 + Q_2S_0 + Q_3S_1)\mathbf{j} + (Q_0S_3 + Q_1S_2 - Q_2S_1 + Q_3S_0)\mathbf{k}.$$

For $\mathbf{Q} \in \mathbb{Q}^{m\times n}$, define the real counterpart $\Upsilon_{\mathbf{Q}}$ and the column representation $\mathbf{Q}_{\mathrm{c}}$ as

$$\Upsilon_{\mathbf{Q}} = \begin{bmatrix} Q_0 & -Q_1 & -Q_2 & -Q_3 \\ Q_1 & Q_0 & -Q_3 & Q_2 \\ Q_2 & Q_3 & Q_0 & -Q_1 \\ Q_3 & -Q_2 & Q_1 & Q_0 \end{bmatrix}, \qquad \mathbf{Q}_{\mathrm{c}} = \begin{bmatrix} Q_0 \\ Q_1 \\ Q_2 \\ Q_3 \end{bmatrix}. \tag{2.1}$$

Note that $\Upsilon_{\mathbf{Q}}$ has special real algebraic structure that is preserved under the following operations [13, 21]:

$$\Upsilon_{k_1\mathbf{P}+k_2\mathbf{Q}} = k_1\Upsilon_{\mathbf{P}} + k_2\Upsilon_{\mathbf{Q}} \ (k_1, k_2 \in \mathbb{R}), \quad \Upsilon_{\mathbf{Q}^*} = \Upsilon_{\mathbf{Q}}^T, \qquad \Upsilon_{\mathbf{QS}} = \Upsilon_{\mathbf{Q}}\Upsilon_{\mathbf{S}}. \tag{2.2}$$

For determinants of quaternion square matrices, a variety of definitions have emerged in terms of the complex and real counterparts to avoid the difficulties caused by the non-commutativity of quaternion multiplications; see [20, 30, 41] and reference therein. However these definitions do not coincide with the standard determinant of a real matrix. In this paper, we only consider the determinant of Hermitian quaternion matrices, which was defined by Li [20] as

$$\mathbf{det}(\mathbf{Q}) = \lambda_1\lambda_2\cdots\lambda_n, \quad \mathbf{Q} \in \mathbb{Q}^{n\times n} \text{ is Hermitian}, \tag{2.3}$$

where $\lambda_1, \ldots, \lambda_n$ are eigenvalues of $\mathbf{Q}$, and they are proved to be real [13, 20]. This definition in (2.3) is consistent with the determinant of a real symmetric matrix, but does not adapt to the quaternion non-Hermitian matrices, since a quaternion non-Hermitian matrix has significantly different properties in its left and right eigenvalues, and there is no very close relation between left and right eigenvalues [41]. When $\mathbf{Q}$ is Hermitian, the left and right eigenvalues are coincided to be the same real value. Throughout this paper we use $\mathbf{det}(\mathbf{Q})$ to distinguish it from the real determinant symbol "det". Moreover, if $\mathbf{Q}$ is positive semidefinite so that $\lambda_i \geq 0$, then the quaternion determinant $\mathbf{det}(\mathbf{Q})$ can be represented in terms of a determinant of a real matrix [20] as

$$\mathbf{det}(\mathbf{Q}) = [\det(\Upsilon_{\mathbf{Q}})]^{1/4}, \quad \mathbf{Q} \text{ is Hermitian and positive semidefinite}. \tag{2.4}$$

DEFINITION 2.1. *The spectral norm (2-norm) of a quaternion vector* $\mathbf{x} = [\mathbf{x}_i] \in \mathbb{Q}^n$ *is* $\|\mathbf{x}\|_2 := \sqrt{\sum_i |\mathbf{x}_i|^2}$. *The 2-norm of a quaternion matrix* $\mathbf{A} = [\mathbf{a}_{ij}] \in \mathbb{Q}^{m\times n}$ *are* $\|\mathbf{A}\|_2 := \max \sigma(\mathbf{A})$, *where* $\sigma(\mathbf{A})$ *is the set of singular values of* $\mathbf{A}$. *The Frobenius norm of* $\mathbf{A}$ *is* $\|\mathbf{A}\|_F = \big(\sum_{i,j} |\mathbf{a}_{ij}|^2\big)^{1/2} = [\mathrm{tr}(\mathbf{A}^*\mathbf{A})]^{1/2}$.

QSVD was firstly proposed in [41, Theorem 7.2] and the partial QSVD was presented in [15].

LEMMA 2.2 (QSVD [41]). *Let* $\mathbf{A} \in \mathbb{Q}^{m\times n}$. *Then there exist two quaternion unitary matrices* $\mathbf{U} \in \mathbb{Q}^{m\times m}$ *and* $\mathbf{V} \in \mathbb{Q}^{n\times n}$ *such that* $\mathbf{U}^*\mathbf{AV} = \Sigma$, *where* $\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_l) \in \mathbb{R}^{m\times n}$ *with* $\sigma_i \geq 0$ *denoting the* i*-th largest singular value of* $\mathbf{A}$ *and* $l = \min(m, n)$.

From [15], the optimal rank-k approximation of $\mathbf{A}$ is given by $\mathbf{A}_k = \mathbf{U}_k\Sigma_k\mathbf{V}_k^*$, where $\mathbf{U}_k$ and $\mathbf{V}_k$ are respectively submatrices of $\mathbf{U}$ and $\mathbf{V}$ by taking their first k columns, and $\Sigma_k = \mathrm{diag}(\sigma_1, \ldots, \sigma_k)$. Furthermore, by the real counterpart of QSVD: $\Upsilon_{\mathbf{U}}^T\Upsilon_{\mathbf{A}}\Upsilon_{\mathbf{V}} = \Upsilon_{\Sigma}$, where $\Upsilon_{\mathbf{U}}$ and $\Upsilon_{\mathbf{V}}$ are real orthogonal matrices, and $\Upsilon_{\Sigma} = \mathrm{diag}(\Sigma, \Sigma, \Sigma, \Sigma)$. As a result, spectral and Frobenius norms of a quaternion matrix can be represented by the ones of real matrices as below

$$\|\mathbf{A}\|_2 = \|\Upsilon_{\mathbf{A}}\|_2, \qquad \|\mathbf{A}\|_F = \frac{1}{2}\|\Upsilon_{\mathbf{A}}\|_F = \|\mathbf{A}_{\mathrm{c}}\|_F. \tag{2.5}$$

Moreover, for consistent quaternion matrices $\mathbf{A}$ and $\mathbf{B}$, it is obvious that

$$\|\mathbf{AB}\|_F \leq \|\mathbf{A}\|_2\|\mathbf{B}\|_F, \quad \|\mathbf{AB}\|_F \leq \|\mathbf{A}\|_F\|\mathbf{B}\|_2. \tag{2.6}$$

2.2. Real randomized SVD and low-rank approximation. Given a real matrix $M \in \mathbb{R}^{m\times n}$, randomized sampling methods [11,23,25,26,39] apply the input matrix M onto a diverse set of random sample vectors $\Omega = [\omega_1 \ \dots \ \omega_\ell]$, expecting $M\Omega$ to capture the main information of the range space of M and to maintain safe approximation error bounds with high probability. In [11], a random Gaussian matrix Ω is used. By applying M to Ω, and then computing the orthogonormal basis Q of the range space of $M\Omega$ via skinny QR factorization in Matlab:

$$\Omega = \texttt{randn}(n,\ell), \qquad [Q,\sim] = \ \texttt{qr}(Y,0), \qquad \text{where } Y = M\Omega,$$

one can get an approximate orthogonal range space of M. Here $\ell = k + p$ and p is a small oversampling factor (say, $p = 5$). In this case, the matrix M is approximated by $M \approx QN$, where QQ^T is an orthogonal projector and the matrix $N := Q^T M$ is of small size $\ell \times n$. The problem then reduces to compute the full SVD of N as $N = \hat{U}\hat{S}\hat{V}^T$. Therefore $M \approx QN = Q\hat{U}\hat{S}\hat{V}^T$, and once a suitable rank k has been chosen based on the decay of $\hat{S}$, the low-rank SVD factors can be determined as

$$\bar{U}_k = Q\hat{U}(:,1:k), \qquad \bar{S}_k = \hat{S}(1:k,1:k), \quad \text{and} \quad \bar{V}_k = \hat{V}(:,1:k)$$

such that $M_k \approx \bar{U}_k\bar{S}_k\bar{V}_k^T$. We refer to the above method as the randomized SVD.

The idea is simple, but whether the projection QQ^T can capture the range of M well depends not only on the property of random matrix, but also on the singular values s_i of the matrix M we are dealing with. It was shown in [11, Theorems 10.5 and 10.6] that for $p \geq 2$, the expectation of the approximation error satisfies

$$\begin{aligned} \mathsf{E}\|(I - QQ^T)M\|_2 &\leq \Big(1 + \sqrt{\tfrac{k}{p-1}}\Big)s_{k+1} + \tfrac{\mathrm{e}\sqrt{k+p}}{p}\Big(\sum_{j=k+1}^{\min(m,n)} s_j^2\Big)^{1/2}, \\ \mathsf{E}\|(I - QQ^T)M\|_F &\leq \Big(1 + \tfrac{k}{p-1}\Big)^{1/2}\Big(\sum_{j=k+1}^{\min(m,n)} s_j^2\Big)^{1/2}. \end{aligned} \tag{2.7}$$

It is observed that when the singular values of M decay very slowly, the method fails to work well, because the singular vectors associated with the tail singular values capture a significant fraction of the range of M, and the range of $Y = M\Omega$ as well. Power scheme can be used to enhance the effect of the approximation, i.e., by applying power operation to generate $Y = (MM^T)^q M\Omega$, where $(MM^T)^q M$ has the same singular space as M, but with a faster decay rate in its singular values.

3. Quaternion randomized SVD. In this section, we develop the randomized QSVD (randsvdQ) algorithm in Algorithm 3.1 and present some measures to improve the efficiency of the algorithm in practical implementations.

How to choose the random test matrix in the algorithm? Consider a simple case about the rank-1 approximation $\mathbf{A}_1 = \sigma_1\mathbf{u}_1\mathbf{v}_1^*$ of the quaternion matrix $\mathbf{A}$. It is easy to prove that $\{\mathbf{y}_*, \mathbf{z}_*\} = \{\mathbf{u}_1, \mathbf{v}_1\}$ is the maximizer of $\max\limits_{\|\mathbf{y}\|_2=\|\mathbf{z}\|_2=1} |\mathbf{y}^*\mathbf{A}\mathbf{z}|$, and $|\mathbf{y}^*\tilde{\mathbf{z}}| = |\mathbf{y}^*\mathbf{A}\mathbf{z}|$ approximates σ_1 for $\mathbf{y} = \mathbf{u}_1$ and $\tilde{\mathbf{z}} = \mathbf{A}\mathbf{v}_1 \in \mathcal{R}(\mathbf{A})$, in which the columns of $\mathbf{A}$ are spanned with quaternion coefficients. In order to capture the main information of $\mathcal{R}(\mathbf{A})$ spanned by dominant left singular vectors of $\mathbf{A}$, it is natural to use a set of $n \times 1$ quaternion random vectors $\mathbf{\Omega} = [\boldsymbol{\omega}^{(1)} \ \dots \ \boldsymbol{\omega}^{(\ell)}]$ to span the columns of $\mathbf{A}$, with random standard real Gaussian matrices as the four parts of $\mathbf{\Omega}$. That means the $n \times \ell$ quaternion random test matrix

$$\mathbf{\Omega} = \Omega_0 + \Omega_1\mathbf{i} + \Omega_2\mathbf{j} + \Omega_3\mathbf{k}, \tag{3.1}$$

where the entries of $\Omega_0, \Omega_1, \Omega_2, \Omega_3$ are random and independently drawn from the $N(0,1)$-normal distribution. The detailed description of randomized QSVD is given in Algorithm 3.1.

Algorithm 3.1 (randsvdQ) Randomized QSVD with fixed rank

(1) Given $\mathbf{A} \in \mathbb{Q}^{m\times n}$, choose target rank k, oversampling parameter p and the power scheme parameter q. Set $\ell = k + p$, and draw an $n \times \ell$ quaternion random test matrix $\mathbf{\Omega}$ as in (3.1).
(2) Construct $\mathbf{Y}_0 = \mathbf{A}\mathbf{\Omega}$ and for $i = 1, 2, \ldots, q$, compute

$$\hat{\mathbf{Y}}_i = \mathbf{A}^*\mathbf{Y}_{i-1} \quad \text{and} \quad \mathbf{Y}_i = \mathbf{A}\hat{\mathbf{Y}}_i.$$

(3) Construct an $m \times \ell$ quaternion orthonormal basis $\mathbf{Q}$ for the range of $\mathbf{Y}_q$ by the quaternion QR decomposition and generate $\mathbf{B} = \mathbf{Q}^*\mathbf{A}$.
(4) Compute the QSVD of a small-size matrix $\mathbf{B}$: $\mathbf{B} = \tilde{\mathbf{U}}\tilde{\Sigma}\tilde{\mathbf{V}}^*$.
(5) Form the rank-k approximation of $\mathbf{A}$: $\widehat{\mathbf{A}}_k^{(q)} = \hat{\mathbf{U}}_k\hat{\Sigma}_k\hat{\mathbf{V}}_k^*$, where

$$\hat{\mathbf{U}}_k = \mathbf{Q}\tilde{\mathbf{U}}(:, 1:k), \qquad \hat{\Sigma}_k = \tilde{\Sigma}(1:k, 1:k), \qquad \hat{\mathbf{V}}_k = \tilde{\mathbf{V}}(:, 1:k).$$

To implement Algorithm 3.1 efficiently, we recommend fast structure-preserving quaternion Householder QR [14, 21] and QSVD algorithms [21, 38]. Based on structure-preserving properties (2.2) of the real counterpart of a quaternion matrix, the essence of fast structure-preserving algorithm is to store the four parts Q_0, Q_1, Q_2, Q_3 of a quaternion matrix $\mathbf{Q}$ only. When the left (or right) quaternion matrix transformation $\mathbf{T}_l$ (or $\mathbf{T}_r$) is applied on $\mathbf{Q}$, it is equivalent to implementing the real matrix multiplication $\Upsilon_{\mathbf{T}_l}\Upsilon_{\mathbf{Q}}$ (or $\Upsilon_{\mathbf{Q}}\Upsilon_{\mathbf{T}_r}$). In order to reduce the computational cost, only the first block column (or row) of $\Upsilon_{\mathbf{Q}}$ is updated and stored. Other blocks in the updated matrix are not explicitly stored and formed, and they can be determined according to the real symmetry structure. For example, in Step 2 of Algorithm 3.1, the four parts of quaternion matrices $\mathbf{Y}_0$, $\hat{\mathbf{Y}}_i$ and $\mathbf{Y}_i$ can be found from the computations of matrices

$$(\mathbf{Y}_0)_{\mathrm{c}} = \Upsilon_{\mathbf{A}}\mathbf{\Omega}_{\mathrm{c}}, \quad \big(\hat{\mathbf{Y}}_i\big)_{\mathrm{c}} = \Upsilon_{\mathbf{A}}^T(\mathbf{Y}_{i-1})_{\mathrm{c}}, \quad \big(\mathbf{Y}_i\big)_{\mathrm{c}} = \Upsilon_{\mathbf{A}}\big(\hat{\mathbf{Y}}_i\big)_{\mathrm{c}},$$

respectively, and in Step 3, the four parts of quaternion matrix $\mathbf{B}$ can be found from $\mathbf{B}_{\mathrm{c}} = \Upsilon_{\mathbf{Q}}^T\mathbf{A}_{\mathrm{c}}$. Note that the computations of $(\mathbf{Y}_0)_{\mathrm{c}} := \Upsilon_{\mathbf{A}}\mathbf{\Omega}_{\mathrm{c}}$ and the quaternion matrix multiplication $\mathbf{Y}_0 = \mathbf{A}\mathbf{\Omega}$ have the same real flops, while the former utilizes BLAS-3 based matrix-matrix operations better, and hence leads to efficient computations.

Once $\mathbf{Y}_q$ is obtained, the fast structure-preserving quaternion Householder QR algorithm [21] can be applied to get the orthonormal basis matrix $\mathbf{Q}$. Here the quaternion Householder transformation $\mathbf{H}$ to reduce a vector $\mathbf{u} \in \mathbb{Q}^s$ into $\mathbf{H}\mathbf{u} = \mathbf{a}e_1$ in the QR process takes the form

$$\mathbf{H} = I_s - 2\mathbf{v}\mathbf{v}^*, \quad \text{with} \quad \mathbf{v} = \frac{\mathbf{u} - \mathbf{a}e_1}{\|\mathbf{u} - \mathbf{a}e_1\|_2}, \quad \mathbf{a} = \begin{cases} -\frac{\mathbf{u}_1}{|\mathbf{u}_1|}\|\mathbf{u}\|_2, & \mathbf{u}_1 \neq 0, \\ -\|\mathbf{u}\|_2, & \text{otherwise}, \end{cases}$$

where e_1 is the first column of the identity matrix I_s.

After computing $\mathbf{B} = \mathbf{Q}^*\mathbf{A}$ in Step 3, the structure-preserving QSVD [38] of $\mathbf{B}$ first factorizes $\mathbf{B}$ into a real bidiagonal matrix $\tilde{B}$ [21], with the help of Golub and Reinsch's idea [9] and quaternion Householder transformation $\mathbf{H}_0$ [21]:

$$\mathbf{H}_0\mathbf{u} := \mathrm{diag}\left(\frac{\mathbf{a}^*}{|\mathbf{a}|}, I_{s-1}\right)\mathbf{H}\mathbf{u} = |\mathbf{a}|e_1 = \|\mathbf{u}\|_2 e_1. \tag{3.2}$$

Afterwards, the standard SVD of the real matrix $\tilde{B}$ completes the QSVD algorithm.

Remark 3.1. The basis matrix $\mathbf{Q}$ in the algorithm is designed to approximate the left dominant singular subspace of $\mathbf{A}$. To get $\mathbf{Q}$, the structure-preserving quaternion Householder QR has better

numerical stability through our numerous experiments, but with more computational cost since all columns of a unitary matrix are computed. Structure-preserving quaternion modified Gram-Schmidt (QMGS) [38, Chp. 2.4.3] is an economical alternative for getting the thin orthonormal factor $\mathbf{Q}$, but might lose the accuracy during the orthogonalization process when the input matrix has relatively small singular values. However, when we are dealing with low-rank approximation of a large input matrix, only a small number of dominant SVD triplets are taken into account, and QMGS sometimes is sufficient to get an orthonormal basis with expected accuracy (See Example 5.2 in Section 5).

Remark 3.2. If ℓ is much smaller than n, i.e., $\mathbf{B}$ is a "short-and-wide" matrix, the direct application of QSVD on $\mathbf{B}$ might lead to large computational cost. Alternatively, we recommend implementing the QMGS of $\mathbf{B}^*$ as

$$\mathbf{B}^* = \hat{\mathbf{Q}}_1\hat{\mathbf{R}}_1, \quad \text{for} \quad \hat{\mathbf{Q}}_1 \in \mathbb{Q}^{n\times\ell}, \quad \hat{\mathbf{R}}_1 \in \mathbb{Q}^{\ell\times\ell}, \tag{3.3}$$

and then computing the QSVD of the $\ell\times\ell$ quaternion matrix $\hat{\mathbf{R}}_1$ as $\hat{\mathbf{R}}_1 = \hat{\mathbf{T}}_1\hat{S}_1\hat{\mathbf{Z}}_1^*$, from which the QSVD of $\mathbf{B}$ is given by $\mathbf{B} = \tilde{\mathbf{U}}\tilde{\Sigma}\tilde{\mathbf{V}}^*$ for $\tilde{\mathbf{U}} = \hat{\mathbf{Z}}_1, \tilde{\Sigma} = \hat{S}_1$ and $\tilde{\mathbf{V}} = \hat{\mathbf{Q}}_1\hat{\mathbf{T}}_1$. We call the corresponding method the preconditioned randomized QSVD (prandsvdQ).

Remark 3.3. If $\mathbf{A}$ is Hermitian, it can be approximated as [11, (5.13)]:

$$\mathbf{A} \approx \mathbf{Q}\mathbf{Q}^*\mathbf{A}\mathbf{Q}\mathbf{Q}^*. \tag{3.4}$$

Then we form the matrix $\mathbf{B} = \mathbf{Q}^*\mathbf{A}\mathbf{Q}$, and use the structure-preserving eigQ algorithm in [13] to compute the eigen-decomposition of $\mathbf{B}$. The corresponding algorithm is referred to as the randeigQ algorithm in the context.

Note that both randeigQ and prandsvdQ reduce a large $n\times n$ problem into a smaller $\ell\times\ell$ problem. The essence of randeigQ computes the eigen-decomposition of a Hermitian matrix $\mathbf{Q}^*\mathbf{A}\mathbf{Q}$, while the prandsvdQ algorithm of $\mathbf{A}$ computes the QSVD of $\hat{\mathbf{R}}_1 = \hat{\mathbf{Q}}_1^*\mathbf{A}\mathbf{Q}$. For large problems with $\ell \ll n$, the cost of the two randomized algorithms is dominated by the quaternion QR procedure for getting $\mathbf{Q}$ and $\hat{\mathbf{Q}}_1$, and prandsvdQ will cost more CPU time for the extra computation of $\hat{\mathbf{Q}}_1$, but might be more accurate in estimating the eigenvalues of $\mathbf{A}$. That is because the columns of $\hat{\mathbf{Q}}_1$ span the range space $\mathcal{R}(\mathbf{A}\mathbf{Q})$, and it is exactly $\mathcal{R}(\mathbf{A}^2\mathbf{\Omega})$, while $\mathbf{Q}$ is the low-rank basis of $\mathcal{R}(\mathbf{A}\mathbf{\Omega})$, therefore $\mathcal{R}(\hat{\mathbf{Q}}_1)$ might have a better approximation of the left dominant singular subspace than $\mathcal{R}(\mathbf{Q})$. We will compare the numerical behaviors of the two algorithms in Section 5.

For the error approximation of randeigQ, if for some parameter ε, $\|(I_m - \mathbf{Q}\mathbf{Q}^*)\mathbf{A}\|_a \le \varepsilon$, then by [11, (5.10)], the error of approximating $\mathbf{A}$ is given by $\|\mathbf{A} - \mathbf{Q}\mathbf{Q}^*\mathbf{A}\mathbf{Q}\mathbf{Q}^*\|_a \le 2\varepsilon$, where ε will be evaluated in next section.

Remark 3.4. When the power scheme is not used in Algorithm 3.1 (i.e. $q = 0$), note that the input matrix $\mathbf{A}$ in Algorithm 3.1 is revisited. However, in some circumstance, the matrix is too large to be stored. Using a similar technique to [4], we develop a method that requires just one pass over the matrix. For the input Hermitian matrix $\mathbf{A}$, according to (3.4) and $\mathbf{B} = \mathbf{Q}^*\mathbf{A}\mathbf{Q}$, the sample matrix

$$\mathbf{Y} = \mathbf{A}\mathbf{\Omega} \approx \mathbf{Q}\mathbf{Q}^*\mathbf{A}\mathbf{Q}\mathbf{Q}^*\mathbf{\Omega} = \mathbf{Q}\mathbf{B}\mathbf{Q}^*\mathbf{\Omega},$$

and the approximation of the matrix $\mathbf{B}$ could be obtained by solving $\mathbf{B}\mathbf{Q}^*\mathbf{\Omega} \approx \mathbf{Q}^*\mathbf{Y}$.

If $\mathbf{A}$ is not Hermitian, analogue to [11, (5.14)-(5.15)], the single-pass algorithm can be constructed based on the relation $\mathbf{A} \approx \mathbf{Q}\mathbf{Q}^*\mathbf{A}\tilde{\mathbf{Q}}\tilde{\mathbf{Q}}^*$, where $\tilde{\mathbf{Q}}$ is the low-rank basis of $\mathcal{R}(\mathbf{A}^*)$ by applying $\mathbf{A}^*$ on a random test matrix $\tilde{\mathbf{\Omega}}$. The matrix $\mathbf{B} = \mathbf{Q}^*\mathbf{A}\tilde{\mathbf{Q}}$ can be approximated by finding a minimum-residual solution to the system of relations $\mathbf{B}\tilde{\mathbf{Q}}^*\mathbf{\Omega} = \mathbf{Q}^*\mathbf{Y}$, $\mathbf{B}^*\mathbf{Q}^*\tilde{\mathbf{\Omega}} = \tilde{\mathbf{Q}}^*\tilde{\mathbf{Y}}$ for $\mathbf{Y} = \mathbf{A}\mathbf{\Omega}$ and $\tilde{\mathbf{Y}} = \mathbf{A}^*\tilde{\mathbf{\Omega}}$.

4. Error analysis. The error analysis of Algorithm 3.1 consists of two parts, including the expected values of approximation errors $\|(I-\mathbf{Q}\mathbf{Q}^*)\mathbf{A}\|_a = \|\widehat{\mathbf{A}}_{k+p}^{(q)} - \mathbf{A}\|_a$ in spectral or Frobenius norm, and the probability bounds of a large deviation as well. The argument relies on special statistical properties of quaternion test matrix $\mathbf{\Omega}$. Specially, we need to evaluate the Frobenius and spectral norms of $\mathbf{\Omega}$ and $\mathbf{\Omega}^\dagger$.

Our theories are established based on the framework of [11]. To start the analysis, we require to use the information of quaternion normal distributions, chi-squared and Wishart distributions. Some of results are provided in the literature, e.g. [20, 24], while some other information needs a rather lengthy deduction. In Section 4.1, we first summarize the main results in Theorems 4.1-4.3 to show the properties of quaternion randomized algorithm. After investigating the statistical properties of quaternion distributions in Section 4.2, we will give the detailed proofs of Theorems 4.1-4.3 in Section 4.3.

4.1. Main results.

THEOREM 4.1. *(Average Frobenius error of the* randsvdQ *algorithm) Let the QSVD of the $m\times n$ $(m \geq n)$ quaternion matrix $\mathbf{A}$ be*

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^* = \mathbf{U}\begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix}\begin{bmatrix} \mathbf{V}_1^* \\ \mathbf{V}_2^* \end{bmatrix}, \quad \Sigma_1 \in \mathbb{R}^{k\times k}, \quad \mathbf{V}_1 \in \mathbb{Q}^{n\times k},$$

where the singular value matrix $\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$ with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$, k is the target rank. For oversampling parameter $p \geq 1$, let $q = 0$, $\ell = k + p \leq n$ and the sample matrix $\mathbf{Y}_0 = \mathbf{A}\mathbf{\Omega}$, where $\mathbf{\Omega}$ is an $n \times \ell$ quaternion random test matrix as in (3.1), and $\mathbf{\Omega}_1 = \mathbf{V}_1^*\mathbf{\Omega}$ is assumed to have full row rank, then the expected approximation error for the rank-$(k+p)$ matrix $\widehat{\mathbf{A}}_{k+p}^{(0)}$ via the power scheme-free randsvdQ *algorithm satisfies*

$$\mathsf{E}\|\widehat{\mathbf{A}}_{k+p}^{(0)} - \mathbf{A}\|_F \leq \left(1 + \frac{4k}{4p+2}\right)^{1/2}\left(\sum_{j>k}\sigma_j^2\right)^{1/2}.$$

THEOREM 4.2. *(Average spectral error of the* randsvdQ *algorithm) With the notations in Theorem 4.1, the expected spectral norm of the approximation error in the power scheme-free algorithm satisfies*

$$\mathsf{E}\|\widehat{\mathbf{A}}_{k+p}^{(0)} - \mathbf{A}\|_2 \leq \left(1 + 3\sqrt{\frac{k}{4p+2}}\right)\sigma_{k+1} + \frac{3\mathrm{e}\sqrt{4k+4p+2}}{2p+2}\Big(\sum_{j>k}\sigma_j^2\Big)^{1/2}. \tag{4.1}$$

If $q > 0$ and the power scheme is used, then for the rank-$(k+p)$ matrix $\widehat{\mathbf{A}}_{k+p}^{(q)}$, the spectral error satisfies

$$\mathsf{E}\|\widehat{\mathbf{A}}_{k+p}^{(q)} - \mathbf{A}\|_2 \leq \left[\left(1 + 3\sqrt{\frac{k}{4p+2}}\right)\sigma_{k+1}^{2q+1} + \frac{3\mathrm{e}\sqrt{4k+4p+2}}{2p+2}\left(\sum_{j>k}\sigma_j^{2(2q+1)}\right)^{1/2}\right]^{1/(2q+1)}.$$

THEOREM 4.3. *(Deviation bound for approximation errors of the* randsvdQ *algorithm) With the notations in Theorem 4.1, we have the following estimate for the Frobenius error*

$$\|\widehat{\mathbf{A}}_{k+p}^{(0)} - \mathbf{A}\|_F \leq \Big(1 + t\sqrt{\frac{3k}{p+1}}\Big)\Big(\sum_{j>k}\sigma_j^2\Big)^{1/2} + ut\frac{\mathrm{e}\sqrt{4k+4p+2}}{4p+4}\sigma_{k+1}, \tag{4.2}$$

except with the probability $2t^{-4p}+\mathrm{e}^{-u^2/2}$. *For the spectral error,*

$$\|\widehat{\mathbf{A}}_{k+p}^{(0)}-\mathbf{A}\|_2 \leq \left(1+\frac{3t}{2}\sqrt{\frac{3k}{p+1}}+ut\eta_{k,p}\right)\sigma_{k+1}+3t\eta_{k,p}\left(\sum_{j>k}\sigma_j^2\right)^{1/2}, \tag{4.3}$$

except with the probability $2t^{-4p}+\mathrm{e}^{-u^2/2}$, *in which* $\eta_{k,p}=\frac{\mathrm{e}\sqrt{4k+4p+2}}{4p+4}$.

Theorems 4.1-4.3 reveal that the performance of the randomized algorithm depends strongly on the properties of singular values of $\mathbf{A}$. When the singular values of $\mathbf{A}$ have fast decay rate, it is much easier to identify a good low-rank basis $\mathbf{Q}$ and provide acceptable error bounds. However, when the singular values of $\mathbf{A}$ decay slowly, the constructed basis $\mathbf{Q}$ may have low accuracy, and the power scheme will increase the decay rate of the singular values of $\mathbf{C}=(\mathbf{A}\mathbf{A}^*)^q\mathbf{A}$, and generate a better low-rank basis matrix.

4.2. Statistical analysis of quaternion random test matrix. In this subsection, we aim to investigate Frobenius and spectral norms of the quaternion test matrix $\mathbf{G}$ and its pseudoinverse, where

$$\mathbf{G}=G_0+G_1\mathbf{i}+G_2\mathbf{j}+G_3\mathbf{k}\in\mathbb{Q}^{m\times n},\quad m\leq n, \tag{4.4}$$

and $G_0,\ldots,G_3$ are standard Gaussian matrices whose entries are random and independently drawn from the normal distribution $N(0,1)$. Note that the norms of $\|\mathbf{G}^\dagger\|_a$ for $a=2,F$ are closely related to the measure of $\left(\mathbf{G}\mathbf{G}^*\right)^{-1}$, where the matrix $\mathbf{G}\mathbf{G}^*$ is named as a quaternion Wishart matrix. As a result, we first recall some well known results about the quaternion normal distribution and Wishart distribution.

DEFINITION 4.4 ([35]). *Let* $\mathbf{z}=z_0+z_1\mathbf{i}+z_2\mathbf{j}+z_3\mathbf{k}$ *be a random* $m\times 1$ *quaternion vector with zero mean. Define the quaternion covariance matrix* $\mathbf{\Sigma}_m=\mathbf{cov}(\mathbf{z},\mathbf{z})=\mathsf{E}(\mathbf{z}\mathbf{z}^*)$ *as*

$$\begin{aligned}\mathbf{\Sigma}_m&=\mathsf{E}[(z_0+z_1\mathbf{i}+z_2\mathbf{j}+z_3\mathbf{k})(z_0^T-z_1^T\mathbf{i}-z_2^T\mathbf{j}-z_3^T\mathbf{k})]\\&=\Sigma_{00}+\Sigma_{11}+\Sigma_{22}+\Sigma_{33}+(-\Sigma_{01}+\Sigma_{10}-\Sigma_{23}+\Sigma_{32})\mathbf{i}\\&\quad+(-\Sigma_{02}+\Sigma_{13}+\Sigma_{20}-\Sigma_{31})\mathbf{j}+(-\Sigma_{03}+\Sigma_{30}-\Sigma_{12}+\Sigma_{21})\mathbf{k},\end{aligned}$$

in which $\Sigma_{ij}=\mathrm{cov}(z_i,z_j)\in\mathbb{R}^{m\times m}$ *is the real covariance of random vectors* z_i *and* z_j.

In particular, when the four parts z_0,z_1,z_2,z_3 of the quaternion vector $\mathbf{z}$ are real independent random vectors drawn from the normal distribution $N(0,I_m)$, then the quaternion random vector $\mathbf{z}$ follows the quaternion normal distribution $\mathbf{N}(0,4I_m)$ law, with the possibility density function (pdf) [35]: $\mathsf{pdf}(\mathbf{z})=(2\pi)^{-2m}\mathrm{etr}(-\frac{1}{2}\mathbf{z}^*\mathbf{z})$. We remark that when $\mathbf{z}\sim\mathbf{N}(0,4I_m)$, $\|\mathbf{z}\|_2^2$ represents the sum of $4m$ independent real variables and each variable follows $N(0,1)$ law. Thus by the concept of real chi-squared distribution, $\|\mathbf{z}\|_2^2$ follows real chi-squared distribution χ_{4m}^2 with $4m$ degrees of freedom.

The following lemma indicates that the quaternion normal distribution $\mathbf{N}(0,4I_m)$ is unitarily invariant.

LEMMA 4.5 ([20]). *For an* $m\times 1$ *quaternion random vector* $\mathbf{z}\sim\mathbf{N}(0,4I_m)$, *let* $\mathbf{y}=\mathbf{B}\mathbf{z}+\mathbf{u}$, *where* $\mathbf{B}$ *is an m-by-m nonsingular quaternion matrix, and* $\mathbf{u}$ *is an m-by-1 quaternion vector, then* $\mathbf{y}\sim\mathbf{N}(\mathbf{u},4\mathbf{B}\mathbf{B}^*)$.

The rigorous definition of the Wishart distribution is given as follows.

DEFINITION 4.6 ([35, 36]). *Let* $\mathbf{Z}=[\mathbf{z}_1\ \mathbf{z}_2\ \ldots\ \mathbf{z}_n]$, *where* $\mathbf{z}_1,\ldots,\mathbf{z}_n$ *are* $m\times 1$ *random independent quaternion vectors drawn from the same distribution, i.e.,* $\mathbf{z}_i\sim\mathbf{N}(0,\mathbf{\Sigma})(1\leq i\leq n)$.

Then $\mathbf{A} = \mathbf{Z}\mathbf{Z}^* \in \mathbb{Q}^{m\times m}$ *is said to have the quaternion Wishart distribution with* n *degrees of freedom and covariance matrix* $\mathbf{\Sigma}$*. We will write that* $\mathbf{A} \sim \mathbf{W}_m(n, \mathbf{\Sigma})$*.*

Note that the matrix $\mathbf{\Sigma}$ could be quaternion or real. In this paper, we are only interested in the real case and use the notation Σ for a distinguishment. The matrix $\mathbf{A}$ is singular when $n < m$, and the pdf of $\mathbf{A}$ doesn't exist in this case. When $m \le n$, the pdf [20, 36] (See also [24, Theorem 4.2.1]) of $\mathbf{A}$ exists. Before giving the pdf, we first recall the definitions of exterior products, which are vital for the volume element of a multivariate density function.

DEFINITION 4.7 ([20, 29]). *For any* $m \times n$ *real matrix* X*, let* $\mathrm{d}X = [\mathrm{d}x_{ij}]$ *denote the matrix of differentials, define the* mn*-exterior product* $\{\mathrm{d}X\}$ *of the* mn *distinct and free elements in* X *as* $\{\mathrm{d}X\} \equiv \bigwedge_{i,j} \mathrm{d}x_{ij}$*. For any* $m \times n$ *quaternion matrix* $\mathbf{X} = X_0 + X_1\mathbf{i} + X_2\mathbf{j} + X_3\mathbf{k}$*, denote* $\mathrm{d}\mathbf{X} = \mathrm{d}X_0 + \mathrm{d}X_1\ \mathbf{i} + \mathrm{d}X_2\ \mathbf{j} + \mathrm{d}X_3\ \mathbf{k}$*, and define* $\{\mathrm{d}\mathbf{X}\} = \{\mathrm{d}X_0\} \wedge \{\mathrm{d}X_1\} \wedge \{\mathrm{d}X_2\} \wedge \{\mathrm{d}X_3\}$*. If* $\mathbf{X}$ *is Hermitian, then* X_0 *is symmetric, while* X_2, X_3, X_4 *are skew-symmetric, and* $\{\mathrm{d}\mathbf{X}\}$ *takes the form*

$$\{\mathrm{d}\mathbf{X}\} = \Big(\bigwedge_{i\le j}^{m} \mathrm{d}(X_0)_{ij}\Big) \wedge \Big(\bigwedge_{i<j}^{m} \mathrm{d}(X_1)_{ij}\Big) \wedge \Big(\bigwedge_{i<j}^{m} \mathrm{d}(X_2)_{ij}\Big) \wedge \Big(\bigwedge_{i<j}^{m} \mathrm{d}(X_3)_{ij}\Big).$$

In the definition, the exterior product of differential form in different order might differ by a factor ± 1. Since we are integrating exterior differential forms representing probability density functions, we ignore the sign of exterior differential forms for the sake of convenience. Based on the notation for the exterior product, the pdf of the quaternion Wishart matrix is given as follows.

LEMMA 4.8 ([20, 24]). *Let the quaternion Wishart matrix* $\mathbf{A} \sim \mathbf{W}_m(n, \Sigma)$*, then the* pdf *of* $\mathbf{A}$ *satisfies*

$$\mathsf{pdf}(\mathbf{A})\{\mathrm{d}\mathbf{A}\} = \beta_{m,n}\,[\det(\Sigma)]^{-2n}\,[\mathbf{det}(\mathbf{A})]^{2(n-m)+1}\,\mathrm{etr}(-2\Sigma^{-1}\mathbf{A})\{\mathrm{d}\mathbf{A}\}, \tag{4.5}$$

in which $\{\mathrm{d}\mathbf{A}\}$ *represents the volume element of this multivariate density function, and*

$$\beta_{m,n} = 2^{2mn}\pi^{-m(m-1)}\left(\prod_{i=1}^{m}\Gamma\big(2(n-i+1)\big)\right)^{-1},$$

with the Gamma function $\Gamma(\cdot)$ *defined by* $\Gamma(x) = \int_0^\infty t^{x-1}\mathrm{e}^{-t}dt (x > 0)$*.*

The properties of the quaternion Wishart matrix are given as follows.

THEOREM 4.9. *Given* $\mathbf{A} \sim \mathbf{W}_m(n, \Sigma)$*.*
(i) For $\mathbf{M} \in \mathbb{Q}^{k\times m}$ *with* $\mathrm{rank}(\mathbf{M}) = k$*, we have* $\mathbf{M}\mathbf{A}\mathbf{M}^* \sim \mathbf{W}_k(n, \mathbf{M}\Sigma\mathbf{M}^*)$*.*
(ii) Partition

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{12}^* & \mathbf{A}_{22} \end{bmatrix}, \quad \Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix},$$

in which $\mathbf{A}_{11} \in \mathbb{Q}^{k\times k}$*,* $\Sigma_{11} \in \mathbb{R}^{k\times k}$*. Let* $\mathbf{A}_{11,2} = \mathbf{A}_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{12}^*$*,* $\Sigma_{11,2} = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$*, then*

$$\mathbf{A}_{11,2} \sim \mathbf{W}_k(n - m + k, \Sigma_{11,2}).$$

Proof. (i) Note that $\mathbf{A} = \sum_{i=1}^{n} \mathbf{z}_i\mathbf{z}_i^*$ with $\mathbf{z}_i \sim \mathbf{N}(0, \Sigma)$. It follows that $\hat{\mathbf{z}}_i := 2\Sigma^{-1/2}\mathbf{z}_i \sim \mathbf{N}(0, 4I_m)$ from the definition of quaternion covariance. By applying Lemma 4.5, $\mathbf{M}\mathbf{z}_i = \frac{1}{2}(\mathbf{M}\Sigma^{1/2}\hat{\mathbf{z}}_i) \sim \mathbf{N}(0, \mathbf{M}\Sigma\mathbf{M}^*)$, and hence

$$\mathbf{M}\mathbf{A}\mathbf{M}^* = \sum_{i=1}^{n} \mathbf{M}\mathbf{z}_i(\mathbf{M}\mathbf{z}_i)^* \sim \mathbf{W}_k(n, \mathbf{M}\Sigma\mathbf{M}^*).$$

(ii) Let $\mathbf{Z} = \begin{bmatrix} I_k & 0 \\ -\mathbf{A}_{22}^{-1}\mathbf{A}_{12}^* & I_{m-k} \end{bmatrix}$, and change the variables of $\mathbf{A}$ into $\mathbf{A}_{11,2}$, $\mathbf{B}_{12} = \mathbf{A}_{12}$ and $\mathbf{B}_{22} = \mathbf{A}_{22}$ through the following transformation

$$\mathbf{AZ} := \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{12}^* & \mathbf{A}_{22} \end{bmatrix} \begin{bmatrix} I_k & 0 \\ -\mathbf{A}_{22}^{-1}\mathbf{A}_{12}^* & I_{m-k} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11,2} & \mathbf{B}_{12} \\ 0 & \mathbf{B}_{22} \end{bmatrix}. \tag{4.6}$$

The quaternion matrix $\mathbf{Z}$ is not Hermitian, and $\mathbf{det}(\mathbf{Z})$ is not well defined. In order to express $\mathbf{det}(\mathbf{A})$ in terms of $\mathbf{det}(\mathbf{A}_{11,2})$ and $\mathbf{det}(\mathbf{B}_{22})$, we consider the transformation $\mathbf{Z}^*\mathbf{AZ}$ to get $\mathbf{Z}^*\mathbf{AZ} = \mathrm{diag}(\mathbf{A}_{11,2}, \mathbf{B}_{22}) =: \mathbf{F}$, where $\mathbf{A}_{11,2}$ and $\mathbf{B}_{22}$ are Hermitian and positive definite matrices.

Take the real counter parts on both sides of $\mathbf{Z}^*\mathbf{AZ} = \mathbf{F}$, the properties in (2.2) gives $\Upsilon_{\mathbf{Z}}^T\Upsilon_{\mathbf{A}}\Upsilon_{\mathbf{Z}} = \Upsilon_{\mathbf{F}}$ and the standard determinant of real matrix $\Upsilon_{\mathbf{F}}$ satisfies

$$\det(\Upsilon_{\mathbf{F}}) = (\det(\Upsilon_{\mathbf{Z}}))^2 \det(\Upsilon_{\mathbf{A}}), \tag{4.7}$$

where by writing the (2,1)-subblock of $\mathbf{Z}$ as $-\mathbf{A}_{22}^{-1}\mathbf{A}_{12}^* = \bar{A}_0 + \bar{A}_1\mathbf{i} + \bar{A}_2\mathbf{j} + \bar{A}_3\mathbf{k}$, and using the identity matrices in block columns 2,4,6,8 of $\Upsilon_{\mathbf{Z}}$:

$$\Upsilon_{\mathbf{Z}} = \left[\begin{array}{cc|cc|cc|cc} I & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \bar{A}_0 & I & -\bar{A}_1 & 0 & -\bar{A}_2 & 0 & -\bar{A}_3 & 0 \\ \hline 0 & 0 & I & 0 & 0 & 0 & 0 & 0 \\ \bar{A}_1 & 0 & \bar{A}_0 & I & -\bar{A}_3 & 0 & \bar{A}_2 & 0 \\ \hline 0 & 0 & 0 & 0 & I & 0 & 0 & 0 \\ \bar{A}_2 & 0 & \bar{A}_3 & 0 & \bar{A}_0 & I & -\bar{A}_1 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & I & 0 \\ \bar{A}_3 & 0 & -\bar{A}_2 & 0 & \bar{A}_1 & 0 & \bar{A}_0 & I \end{array}\right]$$

to eliminate the subblocks $\pm\bar{A}_i$ to zero, we get $\det(\Upsilon_{\mathbf{Z}}) = \det(I_{4m}) = 1$. Thus in (4.7), $\det(\Upsilon_{\mathbf{A}}) = \det(\Upsilon_{\mathbf{F}})$. The applications of (2.4) and the definition (2.3) to this equality give

$$\mathbf{det}(\mathbf{A}) = \mathbf{det}(\mathbf{F}) = \mathbf{det}(\mathbf{A}_{11,2})\mathbf{det}(\mathbf{B}_{22}). \tag{4.8}$$

For the real matrix Σ, it is obvious that

$$\det(\Sigma) = \det(\Sigma_{22}) \det(\Sigma_{11,2}). \tag{4.9}$$

By putting $C = \Sigma^{-1} = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}$, we conclude that $C_{11} = \Sigma_{11,2}^{-1}$ and

$$\begin{aligned} \mathrm{tr}(\Sigma^{-1}\mathbf{A}) &= \mathrm{tr}\left(\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \begin{bmatrix} \mathbf{A}_{11,2} + \mathbf{B}_{12}\mathbf{B}_{22}^{-1}\mathbf{B}_{12}^* & \mathbf{B}_{12} \\ \mathbf{B}_{12}^* & \mathbf{B}_{22} \end{bmatrix} \right) \\ &= \mathrm{tr}(C_{11}\mathbf{A}_{11,2}) + \mathrm{tr}(\mathbf{\Delta}_1) + \mathrm{tr}(\mathbf{\Delta}_2) = \mathrm{tr}(\Sigma_{11,2}^{-1}\mathbf{A}_{11,2}) + \mathrm{tr}(\mathbf{\Delta}_1) + \mathrm{tr}(\mathbf{\Delta}_2), \end{aligned} \tag{4.10}$$

where $\mathbf{\Delta}_1 = \Sigma_{11,2}^{-1}\mathbf{B}_{12}\mathbf{B}_{22}^{-1}\mathbf{B}_{12}^* + C_{12}\mathbf{B}_{12}^*$, $\mathbf{\Delta}_2 = C_{21}\mathbf{B}_{12} + C_{22}\mathbf{B}_{22}$.

Note that the differential of $\mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{12}^*$ satisfies

$$\mathrm{d}(\mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{12}^*) = (\mathrm{d}\mathbf{A}_{12})\mathbf{A}_{22}^{-1}\mathbf{A}_{12}^* + \mathbf{A}_{12}(\mathrm{d}\mathbf{A}_{22}^{-1})\mathbf{A}_{12}^* + \mathbf{A}_{12}\mathbf{A}_{22}^{-1}(\mathrm{d}\mathbf{A}_{12}^*),$$

in which the differential $\mathrm{d}(\mathbf{A}_{22}^{-1})$ can be derived by differentiating $\mathbf{A}_{22}^{-1}\mathbf{A}_{22} = I_{m-k}$ as

$$(\mathrm{d}\mathbf{A}_{22}^{-1})\mathbf{A}_{22} + \mathbf{A}_{22}^{-1}(\mathrm{d}\mathbf{A}_{22}) = 0, \quad \text{or equivalently,} \quad \mathrm{d}\mathbf{A}_{22}^{-1} = -\mathbf{A}_{22}^{-1}(\mathrm{d}\mathbf{A}_{22})\mathbf{A}_{22}^{-1}.$$

Since the exterior products of repeated differentials are zero, we then get $\{\mathrm{d}(\mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{12}^{*})\} \wedge \{\mathrm{d}\mathbf{A}_{12}\} \wedge \{\mathrm{d}\mathbf{A}_{22}\} = 0$. Thus

$$\begin{aligned}\{\mathrm{d}\mathbf{A}\} &= \{\mathrm{d}\mathbf{A}_{11}\} \wedge \{\mathrm{d}\mathbf{A}_{12}\} \wedge \{\mathrm{d}\mathbf{A}_{22}\} = \{\mathrm{d}(\mathbf{A}_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{12}^{*})\} \wedge \{\mathrm{d}\mathbf{A}_{12}\} \wedge \{\mathrm{d}\mathbf{A}_{22}\} \\ &= \{\mathrm{d}\mathbf{A}_{11,2}\} \wedge \{\mathrm{d}\mathbf{B}_{12}\} \wedge \{\mathrm{d}\mathbf{B}_{22}\}.\end{aligned} \tag{4.11}$$

Substituting (4.8)-(4.11) into $\mathsf{pdf}(\mathbf{A})\{\mathrm{d}\mathbf{A}\}$ in Lemma 4.8, we obtain

$$\begin{aligned}\mathsf{pdf}(\mathbf{A})\{\mathrm{d}\mathbf{A}\} &= \beta_{m,n}\left([\det(\Sigma_{11,2})]^{-2n}\left[\mathbf{det}(\mathbf{A}_{11,2})\right]^{2(n-m)+1}\mathrm{etr}(-2\Sigma_{11,2}^{-1}\mathbf{A}_{11,2})\right) \\ &\times\left([\det(\Sigma_{22})]^{-2n}\left[\mathbf{det}(\mathbf{B}_{22})\right]^{2(n-m)+1}\mathrm{etr}(-2\boldsymbol{\Delta}_1)\mathrm{etr}(-2\boldsymbol{\Delta}_2)\right)\{\mathrm{d}\mathbf{A}_{11,2}\} \wedge \{\mathrm{d}\mathbf{B}_{12}\} \wedge \{\mathrm{d}\mathbf{B}_{22}\},\end{aligned} \tag{4.12}$$

from which we see that $\mathbf{A}_{11,2}$ is independent of $\mathbf{B}_{12}, \mathbf{B}_{22}$, because of the density function factors. Notice that $\mathbf{A}_{\mathbf{11,2}}$ is $k \times k$, and $\left[\mathbf{det}(\mathbf{A}_{11,2})\right]^{2(n-m)+1} = \left[\mathbf{det}(\mathbf{A}_{11,2})\right]^{2((n-m+k)-k)+1}$. Moreover, the terms in (4.12) including $\mathbf{A}_{11,2}$ have close relations to the pdf of a Wishart matrix, therefore we can find the pdf of $\mathbf{A}_{\mathbf{11,2}}$ from $\mathsf{pdf}(\mathbf{A})$ so that $\mathsf{pdf}(\mathbf{A}_{\mathbf{11,2}})$ takes the form

$$\beta_{k,n-m+k}[\det(\Sigma_{11,2})]^{-2(n-m+k)}\left[\mathbf{det}(\mathbf{A}_{11,2})\right]^{2((n-m+k)-k)+1}\mathrm{etr}(-2\Sigma_{11,2}^{-1}\mathbf{A}_{11,2}),$$

which means $\mathbf{A}_{11,2} \sim \mathbf{W}_k(n-m+k, \Sigma_{11,2})$. The remaining terms in (4.12) correspond to the joint pdf of $\mathbf{B}_{12}$, $\mathbf{B}_{22}$, whose distributions will not be considered here. ∎

Theorem 4.9 includes the properties of a real Wishart matrix [29, Theorems 3.2.5 and 3.2.10] as special cases. With Theorem 4.9, the expectation of $\|\mathbf{G}^{\dagger}\|_F^2$ is deduced in the following theorem.

THEOREM 4.10. *Let the quaternion random matrix $\mathbf{G} \in \mathbb{Q}^{m\times n}(m \le n)$ be given by (4.4). Then the expectation of $\|\mathbf{G}^{\dagger}\|_F^2$ satisfies*

$$\mathsf{E}\|\mathbf{G}^{\dagger}\|_F^2 = \frac{m}{4(n-m)+2}.$$

Proof. It is obvious that each column in $\mathbf{G}$ follows $\mathbf{N}(0, 4I_m)$ and

$$\mathsf{E}\|\mathbf{G}^{\dagger}\|_F^2 = \mathsf{E}\left(\mathrm{tr}\left[(\mathbf{G}\mathbf{G}^{*})^{-1}\right]\right) = \mathsf{E}\sum_{i=1}^{m}(e_i^T\mathbf{A}^{-1}e_i) = \sum_{i=1}^{m}\mathsf{E}(e_i^T\mathbf{A}^{-1}e_i), \tag{4.13}$$

where e_i is the i-th column of the identity matrix I_m, and $\mathbf{A} = \mathbf{G}\mathbf{G}^{*} \sim \mathbf{W}_m(n, 4I_m)$.

For each fixed i, let $\Pi_{1,i}$ be the permutation matrix obtained by interchanging columns $1, i$ in the $m\times m$ identity matrix, and denote $\mathbf{C} = \Pi_{1,i}^T\mathbf{A}\Pi_{1,i} = \begin{bmatrix}\mathbf{C}_{11} & \mathbf{C}_{12}\\ \mathbf{C}_{21} & \mathbf{C}_{22}\end{bmatrix}$ with $\mathbf{C}_{11} \in \mathbb{Q}^{1\times 1}$, then $\mathbf{C} \sim \mathbf{W}_m(n, 4I_m)$ by Theorem 4.9(i). Moreover, $\left(e_i^T\mathbf{A}^{-1}e_i\right)^{-1} = \left(e_1^T\mathbf{C}^{-1}e_1\right)^{-1} = \mathbf{C}_{11} - \mathbf{C}_{12}\mathbf{C}_{22}^{-1}\mathbf{C}_{21}$.

According to Theorem 4.9(ii), $\left(e_i^T\mathbf{A}^{-1}e_i\right)^{-1} \sim \mathbf{W}_1(n-m+1, 4)$, indicating that there exists an $(n-m+1)$-dimensional quaternion column vector $\mathbf{z} \sim \mathbf{N}(0, 4I_{n-m+1})$ satisfying

$$\left(e_i^T\mathbf{A}^{-1}e_i\right)^{-1} = \|\mathbf{z}\|_2^2 \sim \chi^2_{4(n-m+1)}. \tag{4.14}$$

By the expectation of the inverted chi-squared distribution in [11, Proposition A.8], we know that

$$\mathsf{E}\left(e_i^T\mathbf{A}^{-1}e_i\right) = \mathsf{E}\frac{1}{\chi^2_{4(n-m+1)}} = \frac{1}{4(n-m)+2}.$$

The assertion in the theorem then follows. ∎

The theorem below provides a bound on the probability of a large deviation above the mean.

THEOREM 4.11. *Let the quaternion random matrix* $\mathbf{G} \in \mathbb{Q}^{m\times n}$ *with* $n-m \geq 1$ *be given by (4.4). Then for each* $t \geq 1$,

$$\mathsf{P}\left\{\|\mathbf{G}^{\dagger}\|_F^2 > \frac{3m}{4(n-m+1)}t\right\} \leq t^{-2(n-m)}. \tag{4.15}$$

Proof. According to (4.13)–(4.14), $Z = \|\mathbf{G}^{\dagger}\|_F^2 = \sum_{i=1}^{m} X_i$ with $X_i = e_i^T\mathbf{A}^{-1}e_i$ and $X_i^{-1} \sim \chi^2_{4(n-m+1)}$. Let $q = 2(n-m)$ and when $n-m \geq 1$, the result in [11, Lemma A.9] ensures that $\|X_i\|_{L^q} := [\mathsf{E}(|X_i|^q)]^{1/q} < \frac{3}{4(n-m+1)}$. Using the triangle inequality for the L^q-norm, we obtain

$$\|Z\|_{L^q} \leq \sum_{i=1}^{m}\|X_i\|_{L^q} < \frac{3m}{4(n-m+1)} =: \gamma.$$

With Markov's inequality, $\mathsf{P}\{Z \geq \gamma t\} = \mathsf{P}\{Z^q \geq \gamma^q t^q\} \leq \frac{\mathsf{E}(Z^q)}{\gamma^q t^q} < t^{-q} = t^{-2(n-m)}$, leading to the desired result. ∎

We now turn to the estimate of $\|\mathbf{G}^{\dagger}\|_2$. Note that $\|\mathbf{G}^{\dagger}\|_2 = (\lambda_{\min}(\mathbf{A}))^{-1/2}$, where $\lambda_{\min}(\mathbf{A})$ denotes the smallest eigenvalue of $\mathbf{A}$. We therefore need to study the pdf of the smallest eigenvalue of $\mathbf{A}$, based on the following lemma and a frame work in [3] for discussing the eigenvalues of a real Wishart matrix.

LEMMA 4.12 ([20]). *Let the quaternion Wishart matrix* $\mathbf{A} \sim \mathbf{W}_m(n, I_m)$, *then the* pdf *for the eigenvalues* $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m > 0$ *of* $\mathbf{A}$ *is given by*

$$f(\lambda_1,\lambda_2,\cdots,\lambda_m) = K_{m,n}\prod_{i=1}^{m}\lambda_i^{2(n-m)+1}\prod_{i<j}^{m}(\lambda_i-\lambda_j)^4\ \mathrm{e}^{-2\sum_{i=1}^{m}\lambda_i},$$

where $K_{m,n}^{-1} = 2^{-2mn}\pi^{2m}\prod_{i=1}^{m}\Gamma\Big(2(n-i+1)\Big)\Gamma\Big(2(m-i+1)\Big)$.

The following lemma gives the lower and upper bounds of the pdf of $\lambda_{\min}(\mathbf{A})$.

LEMMA 4.13. *Let the quaternion Wishart matrix* $\mathbf{A} \sim \mathbf{W}_m(n, I_m)$, *and* $f_{\lambda_{\min}}(\lambda)$ *denote the* pdf *of the smallest eigenvalue of quaternion Wishart matrix* $\mathbf{A}$, *then* $f_{\lambda_{\min}}(\lambda)$ *satisfies*

$$L_{m,n}\mathrm{e}^{-2m\lambda}\lambda^{2(n-m)+1} \leq f_{\lambda_{\min}}(\lambda) \leq L_{m,n}\mathrm{e}^{-2\lambda}\lambda^{2(n-m)+1}, \tag{4.16}$$

where

$$L_{m,n} = \frac{2^{2(n-m+1)}\pi^{-2}\Gamma(2n+2)}{\Gamma(2n-2m+4)\Gamma(2n-2m+2)\Gamma(2m)}. \tag{4.17}$$

Proof. For $\lambda \geq 0$, let $R_{m-1}(\lambda) = \{(\lambda_1,\lambda_2,\ldots,\lambda_{m-1}) : \lambda_1 \geq \cdots \geq \lambda_{m-1} \geq \lambda\} \subseteq \mathbb{R}^{1\times(m-1)}$. From the pdf of the eigenvalues of $\mathbf{A}$ in Lemma 4.12, we have

$$\begin{aligned} f_{\lambda_{\min}}(\lambda) &= \int_{R_{m-1}(\lambda)} f(\lambda_1,\lambda_2,\cdots,\lambda_{m-1},\lambda)\mathrm{d}\lambda_1\mathrm{d}\lambda_2\cdots\mathrm{d}\lambda_{m-1} \\ &= K_{m,n}\mathrm{e}^{-2\lambda}\lambda^{2(n-m)+1}\int_{R_{m-1}(\lambda)}\mathrm{e}^{-2\sum_{i=1}^{m-1}\lambda_i}\prod_{i=1}^{m-1}\lambda_i^{2(n-m)+1} \\ &\qquad \prod_{i=1}^{m-1}(\lambda_i-\lambda)^4\prod_{i=1}^{m-2}\prod_{j=i+1}^{m-1}(\lambda_i-\lambda_j)^4\mathrm{d}\lambda_1\mathrm{d}\lambda_2\cdots\mathrm{d}\lambda_{m-1}. \end{aligned}$$

By the inequality $(\lambda_i-\lambda)^4\le\lambda_i^4$, we find that

$$\begin{aligned} f_{\lambda_{\min}}(\lambda) &\le K_{m,n}\mathrm{e}^{-2\lambda}\lambda^{2(n-m)+1}\int_{R_{m-1}(0)}\mathrm{e}^{-2\sum_{i=1}^{m-1}\lambda_i}\prod_{i=1}^{m-1}\lambda_i^{2(n-m)+5}\\ &\qquad\prod_{i=1}^{m-2}\prod_{j=i+1}^{m-1}(\lambda_i-\lambda_j)^4\mathrm{d}\lambda_1\mathrm{d}\lambda_2\cdots\mathrm{d}\lambda_{m-1}\\ &=: K_{m,n}\mathrm{e}^{-2\lambda}\lambda^{2(n-m)+1}C_{m,n}. \end{aligned}$$

For the lower bound, set $\mu_i=\lambda_i-\lambda(i=1,\dots,m-1)$, then $\mu_1\ge\mu_2\ge\cdots\ge\mu_{m-1}\ge 0$, and

$$\begin{aligned} f_{\lambda_{\min}}(\lambda) &= K_{m,n}\mathrm{e}^{-2m\lambda}\lambda^{2(n-m)+1}\int_{R_{m-1}(0)}\mathrm{e}^{-2\sum_{i=1}^{m-1}\mu_i}\prod_{i=1}^{m-1}(\mu_i+\lambda)^{2(n-m)+1}\\ &\qquad\prod_{i=1}^{m-1}\mu_i^4\prod_{i=1}^{m-2}\prod_{j=i+1}^{m-1}(\mu_i-\mu_j)^4\mathrm{d}\mu_1\mathrm{d}\mu_2\cdots\mathrm{d}\mu_{m-1}\\ &\ge K_{m,n}\mathrm{e}^{-2m\lambda}\lambda^{2(n-m)+1}\int_{R_{m-1}(0)}\mathrm{e}^{-2\sum_{i=1}^{m-1}\mu_i}\prod_{i=1}^{m-1}\mu_i^{2(n-m)+5}\\ &\qquad\prod_{i=1}^{m-2}\prod_{j=i+1}^{m-1}(\mu_i-\mu_j)^4\mathrm{d}\mu_1\mathrm{d}\mu_2\cdots\mathrm{d}\mu_{m-1}\\ &= K_{m,n}\mathrm{e}^{-2m\lambda}\lambda^{2(n-m)+1}C_{m,n}. \end{aligned}$$

Note that $f(\lambda_1,\dots,\lambda_m)$ is a probability density function, therefore by the expression of $K_{m,n}$ in Lemma 4.12,

$$\int_{R_m(0)}\mathrm{e}^{-2\sum_{i=1}^{m}\lambda_i}\prod_{i=1}^{m}\lambda_i^{2(n-m)+1}\prod_{i=1}^{m-1}\prod_{j=i+1}^{m}(\lambda_i-\lambda_j)^4\mathrm{d}\lambda_1\mathrm{d}\lambda_2\cdots\mathrm{d}\lambda_m=K_{m,n}^{-1}.$$

It then follows that $C_{m,n}=K_{m-1,n+1}^{-1}$ and hence the inequality (4.16) holds, where $L_{m,n}=\frac{K_{m,n}}{K_{m-1,n+1}}$ and it takes the form (4.17) by Theorem 4.9(i). The assertion in the lemma then follows.□

THEOREM 4.14. *Let $\mathbf{G}\in\mathbb{Q}^{m\times n}$ be given by (4.4). Then*

$$\mathsf{P}\big\{\|\mathbf{G}^\dagger\|_2>\frac{\mathrm{e}\sqrt{4n+2}}{4(n-m+1)}t\big\}\le\frac{\pi^{-3}}{4(n-m+1)(2n-2m+3)}t^{-4(n-m+1)}, \tag{4.18}$$

and $\mathsf{E}\|\mathbf{G}^\dagger\|_2\le\frac{\mathrm{e}\sqrt{4n+2}}{2n-2m+2}$.

Proof. Note that the columns of $\mathbf{G}$ follow $\mathbf{N}(0,4I_m)$ law, therefore according to Theorem 4.9(i), $\mathbf{A}=\frac{1}{4}\mathbf{G}\mathbf{G}^*\sim\mathbf{W}_m(n,I_m)$.

Assume that $\lambda_{\min}$ is the smallest eigenvalue of $\mathbf{A}$. By Lemma 4.13, we know that

$$\begin{aligned} \mathsf{P}\big\{\lambda_{\min}<\gamma\big\} &= \int_0^\gamma f_{\lambda_{\min}}(t)\mathrm{d}t\le L_{m,n}\int_0^\gamma t^{2(n-m)+1}\mathrm{d}t\\ &\le \frac{2^{2(n-m+1)}\pi^{-2}(2n+1)^{2(n-m+1)}\Gamma(2m)}{\Gamma(2n-2m+4)\Gamma(2n-2m+2)\Gamma(2m)}\frac{\gamma^{2n-2m+2}}{2n-2m+2}\\ &= \frac{\pi^{-2}(4n+2)^{2n-2m+2}}{(2n-2m+3)[\Gamma(2n-2m+3)]^2}\gamma^{2n-2m+2}\\ &\approx \frac{\pi^{-3}}{4(n-m+1)(2n-2m+3)}\Big[\frac{\mathrm{e}\sqrt{4n+2}}{2n-2m+2}\Big]^{2(2n-2m+2)}\gamma^{2n-2m+2}\\ &=: C\gamma^{2n-2m+2}, \end{aligned}$$

where we have used the Stirling's approximation formula $\Gamma(n+1) = n! \approx \sqrt{2\pi n}\big(\frac{n}{\mathrm{e}}\big)^n$. Thus

$$\mathsf{P}\big\{\|\mathbf{G}^\dagger\|_2 > \tau\big\} = \mathsf{P}\big\{\lambda_{\min} < \frac{1}{4}\tau^{-2}\big\} \le \bar{C}\tau^{-2(2n-2m+2)},$$

for $\bar{C} = C/4^{2n-2m+2}$. The estimate in (4.18) is derived.

To estimate $\mathsf{E}\|\mathbf{G}^\dagger\|_2$, set $\ell = 2(n-m+1)$, then for any $a \ge 0$,

$$\mathsf{E}\|\mathbf{G}^\dagger\|_2 = \int_0^{+\infty} \mathsf{P}\big\{\|\mathbf{G}^\dagger\|_2 > \tau\big\}\mathrm{d}\tau \le a + \int_a^{+\infty} \mathsf{P}\big\{\|\mathbf{G}^\dagger\|_2 > \tau\big\}\mathrm{d}\tau \le a + \frac{\bar{C}a^{1-2\ell}}{2\ell-1},$$

where the right-hand side is minimized for $a = \bar{C}^{1/(2\ell)} = 2^{-1}C^{1/(2\ell)}$. Then

$$\mathsf{E}\|\mathbf{G}^\dagger\|_2 \le (1 + \frac{1}{2\ell-1})\bar{C}^{1/(2\ell)} \le 2\bar{C}^{1/(2\ell)} \le \frac{\mathrm{e}\sqrt{4n+2}}{2n-2m+2}.$$

The assertion for $\mathsf{E}\|\mathbf{G}^\dagger\|_2$ then follows. ∎

The spectral or Frobenius norm of $\mathbf{G}$ is also vital for our error analysis. For the real Gaussian matrix $\tilde{G}$, the expectation of spectral or Frobenius norm of the scaled matrix $\tilde{S}\tilde{G}\tilde{T}$ has been proven to satisfy the following sharp bounds [11, Proposition 10.1]:

$$\mathsf{E}\|\tilde{S}\tilde{G}\tilde{T}\|_F^2 = \|\tilde{S}\|_F^2\|\tilde{T}\|_F^2, \quad \mathsf{E}\|\tilde{S}\tilde{G}\tilde{T}\|_2 \le \|\tilde{S}\|_2\|\tilde{T}\|_F + \|\tilde{S}\|_F\|\tilde{T}\|_2. \tag{4.19}$$

Based on above results, we present the estimates for the norms of quaternion scaled matrix $\mathbf{SGT}$.

LEMMA 4.15. *Let $\mathbf{G} \in \mathbb{Q}^{m\times n}$ be given by (4.4), and $\mathbf{S} \in \mathbb{Q}^{l\times m}, \mathbf{T} \in \mathbb{Q}^{n\times r}$ be any two fixed quaternion matrices, then*

$$\mathsf{E}\|\mathbf{SGT}\|_F^2 = 4\|\mathbf{S}\|_F^2\|\mathbf{T}\|_F^2, \tag{4.20}$$

$$\mathsf{E}\|\mathbf{SGT}\|_2 \le 3(\|\mathbf{S}\|_2\|\mathbf{T}\|_F + \|\mathbf{S}\|_F\|\mathbf{T}\|_2). \tag{4.21}$$

Proof. Note that the distribution of $\mathbf{G}$ and Frobenius norm of a matrix are both invariant under unitary transformations. As a result, without loss of generality, we assume that $\mathbf{S}, \mathbf{T}$ are real diagonal matrices whose diagonal entries are exactly their singular values. Write $\mathbf{S} = S, \mathbf{T} = T$, it follows that

$$\mathsf{E}\|\mathbf{SGT}\|_F^2 = \mathsf{E}\sum_{k,j}(|s_{kk}\mathbf{g}_{kj}t_{jj}|)^2 = \sum_{k,j}|s_{kk}|^2|t_{jj}|^2\mathsf{E}|\mathbf{g}_{kj}|^2 = 4\|\mathbf{S}\|_F^2\|\mathbf{T}\|_F^2,$$

where $\mathsf{E}|\mathbf{g}_{kj}|^2 = 4$ because the quaternion number $\mathbf{g}_{kj}$ follows $\mathbf{N}(0,4)$ law.

For the spectral norm, by the real counter part of $\mathbf{SGT}$, we know that $\|\mathbf{SGT}\|_2 = \|\Upsilon_S\Upsilon_{\mathbf{G}}\Upsilon_T\|_2$ in which $\Upsilon_{\mathbf{G}}$ has dependent subblocks, and hence it is not a real Gaussian matrix. In order to apply the result in (4.19) to the quaternion spectral norm estimation, write $\Upsilon_{\mathbf{G}}$ in terms of its first block column $\mathbf{G}_\mathrm{c}$:

$$\Upsilon_{\mathbf{G}} = [J_0\mathbf{G}_\mathrm{c} \quad J_1\mathbf{G}_\mathrm{c} \quad J_2\mathbf{G}_\mathrm{c} \quad J_3\mathbf{G}_\mathrm{c}], \tag{4.22}$$

where $\mathbf{G}_\mathrm{c}$ is a real Gaussian matrix, $J_0 = I_{4m}$ and

$$J_1 = \begin{bmatrix} -e_2^T \\ e_1^T \\ e_4^T \\ -e_3^T \end{bmatrix} \otimes I_m, \quad J_2 = \begin{bmatrix} -e_3^T \\ -e_4^T \\ e_1^T \\ e_2^T \end{bmatrix} \otimes I_m, \quad J_3 = \begin{bmatrix} -e_4^T \\ e_3^T \\ -e_2^T \\ e_1^T \end{bmatrix} \otimes I_m, \tag{4.23}$$

and e_i is the i-th column of the 4×4 identity matrix.

Note that for four arbitrary real matrices $M_0, \ldots, M_3$ with the same rows,

$$\|[M_0 \;\; M_1 \;\; M_2 \;\; M_3]\|_2 = \|\sum_{i=0}^{3} M_i M_i^*\|_2^{1/2} \le 2 \max_{0\le i\le 3} \|M_i\|_2.$$

Using this inequality to evaluate the spectral norm of $\mathbf{SGT}$, we obtain

$$\|\mathbf{SGT}\|_2 \;\; = \|\Upsilon_S[J_0\mathbf{G}_c T \;\; J_1\mathbf{G}_c T \;\; J_2\mathbf{G}_c T \;\; J_3\mathbf{G}_c T]\|_2 \le 2 \max_{0\le k\le 3} \|\Upsilon_S J_k \mathbf{G}_c T\|_2 = 2\|\Upsilon_S \mathbf{G}_c T\|_2,$$

where we have used the facts $J_k^T \Upsilon_S J_k = \Upsilon_S$ and $\|\Upsilon_S J_k \mathbf{G}_c T\|_2 = \|\Upsilon_S \mathbf{G}_c T\|_2$.

Therefore by (4.19) and (2.5), we have

$$\mathsf{E}\|\mathbf{SGT}\|_2 \le 2\left(\|\Upsilon_S\|_2 \|T\|_F + \|\Upsilon_S\|_F \|T\|_2\right) = 2\|\mathbf{S}\|_2\|\mathbf{T}\|_F + 4\|\mathbf{S}\|_F\|\mathbf{T}\|_2.$$

By applying above estimates to evaluate $\mathsf{E}\|\mathbf{SGT}\|_2 = \mathsf{E}\|\mathbf{T}^*\mathbf{G}^*\mathbf{S}^*\|_2$, we obtain $\mathsf{E}\|\mathbf{SGT}\|_2 \le 2\|\mathbf{S}\|_F\|\mathbf{T}\|_2 + 4\|\mathbf{S}\|_2\|\mathbf{T}\|_F$. Take the average of the two upper bounds of $\mathsf{E}\|\mathbf{SGT}\|_2$, the assertion in (4.21) follows. ∎

4.3. Proofs of Theorems 4.1-4.3. Throughout this subsection, $\|\cdot\|_a$ denotes either the spectral norm or Frobenius norm.

Proof of Theorem 4.1. Let $\mathbf{Q}$ be the orthonormal basis for the range of the sample matrix $\mathbf{Y}_0 = \mathbf{A}\mathbf{\Omega}$. Set $\mathbf{\Omega}_i = \mathbf{V}_i^*\mathbf{\Omega}$ for $i = 1, 2$, then by a similar deduction to [11, Theorem 9.1], the following inequality

$$\|\widehat{\mathbf{A}}_{k+p}^{(0)} - \mathbf{A}\|_a^2 = \|(I_m - \mathbf{Q}\mathbf{Q}^*)\mathbf{A}\|_a^2 \le \|\Sigma_2\|_a^2 + \|\Sigma_2\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\|_a^2 \le \left(\|\Sigma_2\|_a + \|\Sigma_2\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\|_a\right)^2, \tag{4.24}$$

also holds for the quaternion case, in which $\mathbf{V}^*\mathbf{\Omega}$ follows the $\mathbf{N}(0, 4I_n)$ law. By Lemma 4.5, $\mathbf{\Omega}_1$, $\mathbf{\Omega}_2$ are disjoint submatrices of $\mathbf{V}^*\mathbf{\Omega}$ with the $k \times (k+p)$ matrix $\mathbf{\Omega}_1$ of full row rank with probability one.

By Jensen's inequality to (4.24), we know that

$$\mathsf{E}\|\widehat{\mathbf{A}}_{k+p}^{(0)} - \mathbf{A}\|_F \;\; \le \left(\mathsf{E}\|\widehat{\mathbf{A}}_{k+p}^{(0)} - \mathbf{A}\|_F^2\right)^{1/2} \le \left(\|\Sigma_2\|_F^2 + \mathsf{E}\|\Sigma_2\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\|_F^2\right)^{1/2},$$

where by conditioning on the value of $\mathbf{\Omega}_1$ and applying (4.20) to the scaled matrix $\Sigma_2\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger$,

$$\mathsf{E}\|\Sigma_2\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\|_F^2 = \mathsf{E}\left(\mathsf{E}\left[\|\Sigma_2\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\|_F^2 \mid \mathbf{\Omega}_1\right]\right) = 4\|\Sigma_2\|_F^2 \mathsf{E}\|\mathbf{\Omega}_1^\dagger\|_F^2,$$

which is exactly $\frac{4k}{4p+2}\|\Sigma_2\|_F^2$ according to Theorem 4.10. The assertion in Theorem 4.1 then follows. ∎

Proof of Theorem 4.2. From (4.24), it is obvious that $\mathsf{E}\|\widehat{\mathbf{A}}_{k+p}^{(0)} - \mathbf{A}\|_2 \le \|\Sigma_2\|_2 + \mathsf{E}\|\Sigma_2\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\|_2$, where by conditioning on the value of $\mathbf{\Omega}_1$ and applying (4.21) to the scaled matrix $\Sigma_2\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger$,

$$\begin{aligned}\mathsf{E}\|\Sigma_2\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\|_2 \;\; &= \mathsf{E}\left(\mathsf{E}\left[\|\Sigma_2\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\|_2 | \mathbf{\Omega}_1\right]\right) \le 3\mathsf{E}(\|\Sigma_2\|_2\|\mathbf{\Omega}_1^\dagger\|_F + \|\Sigma_2\|_F\|\mathbf{\Omega}_1^\dagger\|_2) \\ &\le 3\|\Sigma_2\|_2\left(\mathsf{E}\|\mathbf{\Omega}_1^\dagger\|_F^2\right)^{1/2} + 3\|\Sigma_2\|_F \mathsf{E}\|\mathbf{\Omega}_1^\dagger\|_2.\end{aligned}$$

The estimate for the expectation of the error then follows from Theorems 4.10 and 4.14.

For the power scheme, let $\tilde{\mathbf{Q}}$ be the orthonormal basis for the range of $\mathbf{Y}_q = \mathbf{C}\boldsymbol{\Omega} = (\mathbf{A}\mathbf{A}^*)^q\,\mathbf{A}\boldsymbol{\Omega} = \mathbf{U}\Sigma^{2q+1}\mathbf{V}^*$. By Jensen's inequality and a similar deduction to [11, Theorem 9.2], we know that

$$\mathsf{E}\|\widehat{\mathbf{A}}_{k+p}^{(q)} - \mathbf{A}\|_2 = \mathsf{E}\|(I_m - \tilde{\mathbf{Q}}\tilde{\mathbf{Q}}^*)\mathbf{A}\|_2 \le \Big(\mathsf{E}\|(I_m - \tilde{\mathbf{Q}}\tilde{\mathbf{Q}}^T)\mathbf{C}\|_2\Big)^{1/(2q+1)},$$

where $\sigma_1^{2q+1}, \ldots, \sigma_n^{2q+1}$ are the singular values of $\mathbf{C}$. The assertion for the power scheme comes true by invoking the result in (4.1). ∎

Remark 4.16. By using the relation $\sum_{j>k} \sigma_j^{2q+1} \le (\min(m,n)-k)\sigma_{k+1}^{2q+1}$, the spectral error in Theorem 4.2 is bounded by $\mathsf{E}\|\widehat{\mathbf{A}}_{k+p}^{(q)} - \mathbf{A}\|_2 \le \sigma_{k+1}\Big[1 + 3\sqrt{\frac{k}{4p+2}} + \frac{3\mathrm{e}\sqrt{4k+4p+2}}{2p+2}\sqrt{\min(m,n)-k}\Big]^{1/(2q+1)}$. The power scheme drives the extra factor in the error to one exponentially fast through increasing the exponent q, and by the time $q \sim \log(\min(m,n))$, $\mathsf{E}\|\widehat{\mathbf{A}}_{k+p}^{(q)} - \mathbf{A}\|_2 \sim \sigma_{k+1}$.

The analysis of deviation bounds for approximation errors in Theorem 4.3 relies on the following well-known concentration result [11, Proposition 10.3] for functions of a real Gaussian matrix.

LEMMA 4.17 ([11]). *Suppose that $h(\cdot)$ is a Lipschitz function on real matrices: $|h(X) - h(Y)| \le L\|X - Y\|_F$ for all $X, Y \in \mathbb{R}^{s\times t}$. Then for an $s \times t$ standard real Gaussian matrix G, $\mathsf{P}\{h(G) \ge \mathsf{E}h(G) + Lu\} \le \mathrm{e}^{-u^2/2}$.*

Proof of Theorem 4.3. For $t \ge 1$, define the parameterized event on which the spectral and Frobenius norms of $\boldsymbol{\Omega}_1$ are both controlled:

$$E_t = \left\{\boldsymbol{\Omega}_1 : \|\boldsymbol{\Omega}_1^\dagger\|_2 \le \frac{\mathrm{e}\sqrt{4k+4p+2}}{4(p+1)} \cdot t \ \text{ and } \ \|\boldsymbol{\Omega}_1^\dagger\|_F \le \sqrt{\frac{3k}{4p+4}} \cdot t\right\}. \tag{4.25}$$

By Theorems 4.11 and 4.14, the probability of the complement of this event satisfies a simple bound

$$\mathsf{P}(E_t^{\mathrm{c}}) \le t^{-(4p+4)} + t^{-4p} \le 2t^{-4p},$$

according to the estimates in (4.15)-(4.18).

Set $\bar{h}(\mathbf{X}) = \|\Sigma_2\mathbf{X}\boldsymbol{\Omega}_1^\dagger\|_F$, in which the real counter part of an $(n-k)\times k$ quaternion matrix $\mathbf{X}$ can be represented on the basis of $\mathbf{X}_{\mathrm{c}}$ as $\Upsilon_{\mathbf{X}} = [J_0\mathbf{X}_{\mathrm{c}}\ J_1\mathbf{X}_{\mathrm{c}}\ J_2\mathbf{X}_{\mathrm{c}}\ J_3\mathbf{X}_{\mathrm{c}}]$ for $J = [J_0\ J_1\ J_2\ J_3]$, and $J_k \in \mathbb{R}^{4(n-k)\times 4(n-k)}$ has similar structure to the one in (4.23).

Owing to (2.5)-(2.6), $\bar{h}(\mathbf{X}) = \frac{1}{2}\|\Upsilon_{\Sigma_2}\Upsilon_{\mathbf{X}}\Upsilon_{\boldsymbol{\Omega}_1^\dagger}\|_F$ and we could write $\bar{h}(\mathbf{X})$ as a function of $\mathbf{X}_{\mathrm{c}}$ with $h(\mathbf{X}_{\mathrm{c}}) := \bar{h}(\mathbf{X})$. Notice that $h(\mathbf{X}_{\mathrm{c}})$ is a Lipschitz function on real matrices:

$$\begin{aligned} |h(\mathbf{X}_{\mathrm{c}}) - h(\mathbf{Y}_{\mathrm{c}})| &= \Big|\|\Sigma_2\mathbf{X}\boldsymbol{\Omega}_1^\dagger\|_F - \|\Sigma_2\mathbf{Y}\boldsymbol{\Omega}_1^\dagger\|_F\Big| \le \|\Sigma_2(\mathbf{X}-\mathbf{Y})\boldsymbol{\Omega}_1^\dagger\|_F \\ &\le \|\Sigma_2\|_2\|\boldsymbol{\Omega}_1^\dagger\|_2\|\mathbf{X}-\mathbf{Y}\|_F = \|\Sigma_2\|_2\|\boldsymbol{\Omega}_1^\dagger\|_2\|\mathbf{X}_{\mathrm{c}}-\mathbf{Y}_{\mathrm{c}}\|_F, \end{aligned} \tag{4.26}$$

with a Lipschitz constant $L \le \|\Sigma_2\|_2\|\boldsymbol{\Omega}_1^\dagger\|_2$. With Jensen's inequality and Lemma 4.15, we get

$$\mathsf{E}[\bar{h}(\boldsymbol{\Omega}_2) \mid \boldsymbol{\Omega}_1] \le \Big(\mathsf{E}[\big(\bar{h}(\boldsymbol{\Omega}_2)\big)^2 \mid \boldsymbol{\Omega}_1]\Big)^{1/2} = 2\|\Sigma_2\|_F\|\boldsymbol{\Omega}_1^\dagger\|_F,$$

where $\bar{h}(\boldsymbol{\Omega}_2) = h\big((\boldsymbol{\Omega}_2)_{\mathrm{c}}\big)$, and $(\boldsymbol{\Omega}_2)_{\mathrm{c}}$ is a real Gaussian matrix. Applying Lemma 4.17, conditionally to the random variable $\bar{h}(\boldsymbol{\Omega}_2) = \|\Sigma_2\boldsymbol{\Omega}_2\boldsymbol{\Omega}_1^\dagger\|_F$ gives

$$P_{u,t} := \mathsf{P}\left\{\|\Sigma_2\boldsymbol{\Omega}_2\boldsymbol{\Omega}_1^\dagger\|_F > 2\|\Sigma_2\|_F\|\boldsymbol{\Omega}_1^\dagger\|_F + \|\Sigma_2\|_2\|\boldsymbol{\Omega}_1^\dagger\|_2 u \mid E_t\right\} \le \mathrm{e}^{-u^2/2}.$$

In (4.25), consider the upper bounds associated with the event E_t and substitute them into the above inequality, then we can get

$$\mathsf{P}\left\{\|\Sigma_2\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\|_F > \sqrt{\frac{3k}{p+1}}\|\Sigma_2\|_F t + \frac{\mathrm{e}\sqrt{4k+4p+2}}{4p+4}\|\Sigma_2\|_2 ut \mid E_t\right\} \le P_{u,t} \le \mathrm{e}^{-u^2/2}.$$

Using $\mathsf{P}(E_t^c) \le 2t^{-4p}$ to remove the conditioning, we obtain

$$\mathsf{P}\left\{\|\Sigma_2\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\|_F > \sqrt{\frac{3k}{p+1}}\Big(\sum_{j>k}\sigma_j^2\Big)^{1/2} t + ut\frac{\mathrm{e}\sqrt{4k+4p+2}}{4p+4}\sigma_{k+1}\right\} \le 2t^{-4p} + \mathrm{e}^{-u^2/2}.$$

In terms of (4.24), $\|\widehat{\mathbf{A}}_{k+p}^{(0)} - \mathbf{A}\|_F \le \|\Sigma_2\|_F + \|\Sigma_2\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\|_F$, the desired probability bound in (4.2) follows.

For the deviation bound of the spectral error, set $\tilde{h}(\mathbf{X}) = \|\Sigma_2\mathbf{X}\mathbf{\Omega}_1^\dagger\|_2$, and view $\tilde{h}(\mathbf{X})$ as a function of $\mathbf{X}_\mathrm{c}$, i.e. $\check{h}(\mathbf{X}_\mathrm{c}) = \tilde{h}(\mathbf{X})$, then

$$|\check{h}(\mathbf{X}_\mathrm{c}) - \check{h}(\mathbf{Y}_\mathrm{c})| \le \|\Sigma_2\|_2\|\mathbf{X}-\mathbf{Y}\|_2\|\mathbf{\Omega}_1^\dagger\|_2 \le \|\Sigma_2\|_2\|\mathbf{\Omega}_1^\dagger\|_2\|\mathbf{X}-\mathbf{Y}\|_F = \|\Sigma_2\|_2\|\mathbf{\Omega}_1^\dagger\|_2\|\mathbf{X}_\mathrm{c}-\mathbf{Y}_\mathrm{c}\|_F,$$

from which we know that $\check{h}(\cdot)$ is also a Lipschitz function with the Lipschitz constant $L \le \|\Sigma_2\|_2\|\mathbf{\Omega}_1^\dagger\|_2$. Using the upper bound for the expectation of $\tilde{h}(\mathbf{\Omega})$:

$$\mathsf{E}[\tilde{h}(\mathbf{\Omega}_2) \mid \mathbf{\Omega}_1] \le 3\left(\|\Sigma_2\|_2\|\mathbf{\Omega}_1^\dagger\|_F + \|\Sigma_2\|_F\|\mathbf{\Omega}_1^\dagger\|_2\right),$$

and the concentration result in Lemma 4.17, it follows that

$$\mathsf{P}\left\{\|\Sigma_2\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\|_2 > 3(\|\Sigma_2\|_2\|\mathbf{\Omega}_1^\dagger\|_F + \|\Sigma_2\|_F\|\mathbf{\Omega}_1^\dagger\|_2) + \|\Sigma_2\|_2\|\mathbf{\Omega}_1^\dagger\|_2 u \mid E_t\right\} \le \mathrm{e}^{-u^2/2}.$$

The bound in (4.3) could be derived from (4.24)-(4.25) with a similar technique. ∎

COROLLARY 4.18. *(Simple deviation bound for the spectral error of power scheme-free algorithm) With the notations in Theorem 4.1, we have the simple upper bound*

$$\|\widehat{\mathbf{A}}_{k+p}^{(0)} - \mathbf{A}\|_2 \le \left(1 + 18\sqrt{1+\frac{k}{p+1}}\right)\sigma_{k+1} + \frac{6\sqrt{4k+4p+2}}{p+1}\left(\sum_{j>k}\sigma_j^2\right)^{1/2}, \tag{4.27}$$

except with the probability $3\mathrm{e}^{-4p}$.

Proof. Taking $u = 2\sqrt{2p}, t = \mathrm{e}$ in Theorem 4.3 leads to

$$\begin{aligned}\|\widehat{\mathbf{A}}_{k+p}^{(0)} - \mathbf{A}\|_2 &\le \left(1 + \tfrac{3\mathrm{e}}{2}\sqrt{\tfrac{3k}{p+1}} + \tfrac{2\sqrt{2p}\mathrm{e}^2}{2\sqrt{p+1}}\sqrt{1+\tfrac{k}{p+1}}\right)\sigma_{k+1} + \tfrac{3\mathrm{e}^2\sqrt{4k+4p+2}}{4p+4}\left(\sum_{j>k}\sigma_j^2\right)^{1/2} \\ &\le \left(1 + (\tfrac{3\sqrt{3}\mathrm{e}}{2} + \sqrt{2}\mathrm{e}^2)\sqrt{1+\tfrac{k}{p+1}}\right)\sigma_{k+1} + \tfrac{6\sqrt{4k+4p+2}}{p+1}\left(\sum_{j>k}\sigma_j^2\right)^{1/2},\end{aligned}$$

from which the desired upper bound follows. ∎

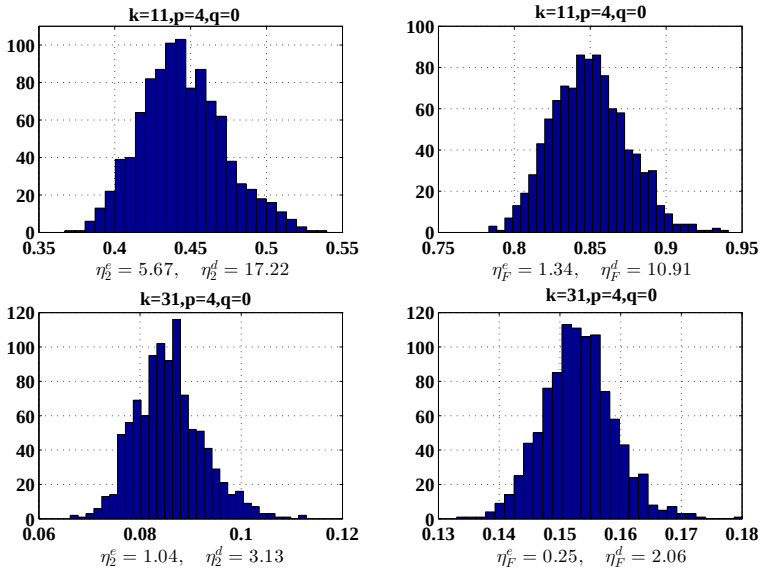

FIG. 5.1. *Approximation errors and upper bounds for a* 100×80 *matrix whose singular values decay very slowly (decay rate: 0.9). The left figures are for the estimates of spectral errors, while the right ones correspond to the Frobenius errors.*

5. Numerical examples. In this section, we give five examples to test the features of randomized QSVD algorithms. The following numerical examples are performed via MATLAB with machine precision $u = 2.22e-16$ in a laptop with Intel Core (TM) i5-8250U CPU @ 1.80GHz and the memory is 8 GB. Algorithms such as quaternion QR, QSVD are coded based on the structure-preserving scheme.

EXAMPLE 5.1. In this example, we test the rationality of estimated bounds for approximation errors $\|\widehat{\mathbf{A}}_{k+p}^{(q)} - \mathbf{A}\|_a$. To this end, we construct an $m \times n (m \geq n)$ quaternion random matrix $\mathbf{A}$ as $\mathbf{A} = \mathbf{U}\begin{bmatrix}\Sigma_1 \\ 0\end{bmatrix}\mathbf{V}^*$, where $\mathbf{U}, \mathbf{V}$ are quaternion Householder matrices taking the form $\mathbf{U} = I_m - 2\mathbf{u}\mathbf{u}^*, \mathbf{V} = I_n - 2\mathbf{v}\mathbf{v}^*$, $\mathbf{u}, \mathbf{v}$ are quaternion unit vectors, and $\Sigma_1 = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)$ is the real $n \times n$ diagonal matrix. Consider singular values with different decay rate as

(1) $\sigma_1 = 1, \sigma_{i+1}/\sigma_i = 0.9$ for $i = 1 \ldots, n-1$ or

(2) $\sigma_1 = 1, \sigma_{i+1}/\sigma_i = 0.1$ for $i = 1 \ldots, n-1$,

where in case (1), the smallest singular value is $\sigma_{80} \approx 2.18 \cdot 10^{-4}$, while in case (2), for the threshold $\theta = 10^{-15}$, the numerical rank of the matrix is 16.

For each case with different values of k, p, we run Algorithm 3.1 with $q = 0$ for 1000 times, and plot the histograms for exact values of $\|\widehat{\mathbf{A}}_{k+p}^{(0)} - \mathbf{A}\|_a$ with $a = 2, F$. Below each histogram, the upper bounds of the errors are listed, where we take $p = 4$ for all cases, and the bound η_a^e for average errors is estimated via Theorems 4.1 and 4.2, while the bound η_a^d for deviation errors is based on (4.2) and (4.27), respectively, in which $u = 2\sqrt{2p}, t = \mathrm{e}$. For $p \geq 4$, the bounds hold with probability 99.99%.

In Figure 5.1, it is observed that for case (1) with slow decay rate in the singular values, the upper bounds η_2^e and η_2^d are respectively about 15 and 40 times the actual values of $\|\widehat{\mathbf{A}}_{k+p}^{(0)} - \mathbf{A}\|_2$, while for the Frobenius error $\|\widehat{\mathbf{A}}_{k+p}^{(0)} - \mathbf{A}\|_F$, the estimated upper bounds η_F^e and η_F^d are much tighter, and they are only about 2 and 10 times the actual values, respectively.

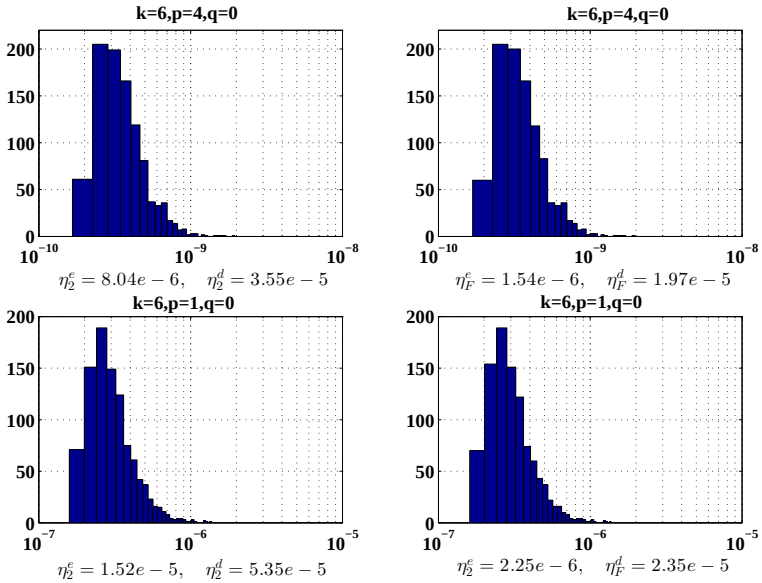

FIG. 5.2. *Approximation errors and upper bounds for a* 100×80 *matrix whose singular values decay very fast (decay rate: 0.1) The left figures are for the estimates of spectral errors, while the right ones correspond to Frobenius errors.*

In Figure 5.2 and for case (2) with fast decay rate in the singular values, a relative large oversampling size $p = 4$ gives upper bounds that are not sharp enough, and there may be a factor $\mathcal{O}(10^4)$ between the estimated upper bounds and actual approximation errors. When we take $p = 1$, the estimates for the upper bounds have been greatly enhanced. The reason is that the tested matrix $\mathbf{A}$ has fast decay rate in its singular values, therefore the orthonormal basis of $\mathcal{R}(\mathbf{A\Omega})$ gives a good approximation of an ℓ-dimensional ($\ell = k + p$) left dominant singular subspace of $\mathbf{A}$, which makes $\|\widehat{\mathbf{A}}_{k+p}^{(0)} - \mathbf{A}\|_2 \approx \sigma_{k+p+1}$, and when $p = 4$, it is much smaller than the estimated bound $\eta_2^e \approx \mathcal{O}(\sigma_{k+1})$.

Overall, the test results in Figures 5.1-5.2 illustrate the rationality of theoretical estimates for approximation errors.

EXAMPLE 5.2. In this example, we test how different values of q in the power scheme affect the approximation errors $\|\widehat{\mathbf{A}}_k^{(q)} - \mathbf{A}\|_a$. We use standard test image lena512[1] with 512×512 pixels. This color image is characterized by a 512×512 pure quaternion matrix $\mathbf{A}$ with entries $\mathbf{A}_{ij} = R_{ij}\mathbf{i} + G_{ij}\mathbf{j} + B_{ij}\mathbf{k}$, where R_{ij}, G_{ij}, B_{ij} represent the red, green and blue pixel values at the location (i, j) in the image, respectively. The singular values and adjacent singular value ratio σ_{k+1}/σ_k of $\mathbf{A}$ are depicted in Figure 5.3.

Based on the structure-preserving quaternion Householder QR and QMGS processes for getting the orthonormal basis matrix $\mathbf{Q}$, we take the oversampling $p = 4$ and depict the approximation errors $\|\widehat{\mathbf{A}}_k^{(q)} - \mathbf{A}\|_a$ for k ranging from 5 to 200 with step 5 in Figures 5.4–5.5, where svdQ plots the optimal rank-k approximation errors obtained via the structure-preserving QSVD algorithm [38].

It is observed that when $k \geq 5$, the adjacent singular value ratio is greater than 0.8, the power scheme with $q = 0$ gives the worst estimates for the rank-k approximation errors among three cases. In the quaternion Householder QR-based algorithm, the case with $q = 2$ behaves better

[1] lena512: https://www.ece.rice.edu/∼wakin/images/

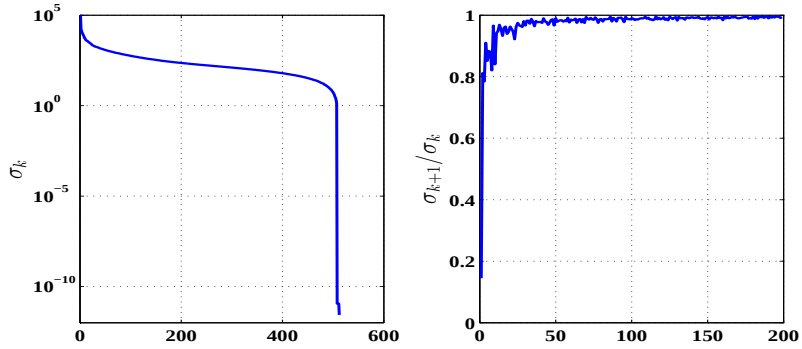

FIG. 5.3. *Singular values and adjacent singular value ratios for color image* lena512.

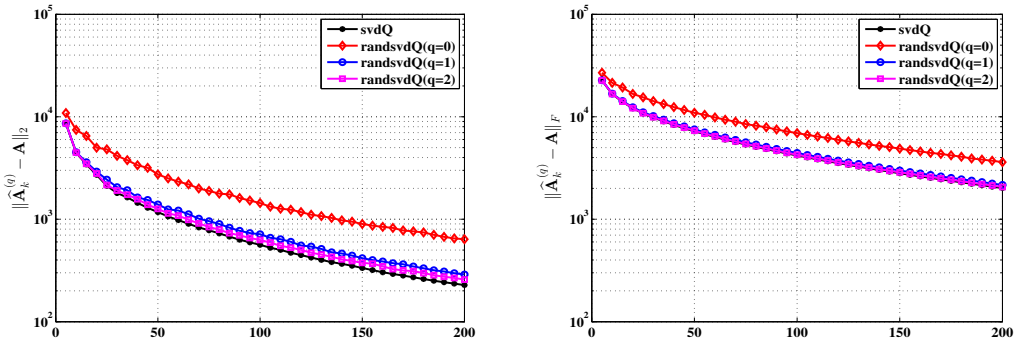

FIG. 5.4. *Errors incurred for different power schemes, in which the orthonormal basis* $\mathbf{Q}$ *in* randsvdQ *is obtained via quaternion Householder QR procedure.*

than that for $q = 1$, since a smaller adjacent singular value ratio $\left(\frac{\sigma_{k+1}}{\sigma_k}\right)^{2q+1}$ of $(\mathbf{A}\mathbf{A}^*)^q\mathbf{A}$ helps generate better basis matrix $\mathbf{Q}$ and rank-k matrix approximation. Although the approximation errors from randomized algorithms are not as accurate as the svdQ-based ones, they still deliver acceptable peak signal-to-noise ratio (PNSR) and relative approximate errors as listed in Table 5.1, in which the PSNR is defined by

$$\mathsf{PSNR}(\widehat{\mathbf{A}}_k^{(q)}, \mathbf{A}) = 10\log_{10}\frac{255^2 mn}{\|\widehat{\mathbf{A}}_k^{(q)} - \mathbf{A}\|_F^2}.$$

It is observed that $q = 1$ is acceptable for the desired accuracy.

In Figure 5.5, QMGS-based method is compared with quaternion Householder QR procedure. QMGS gives satisfactory approximations for $k < 160$ and $q = 1$ or 2, while for $q = 2$ and $k \geq 160$, the estimates become worse. That is partly because for $q = 2$, $\left(\frac{\sigma_1}{\sigma_{165}}\right)^{2q+1} = 1.1e + 13$ and $\mathbf{Y}_q = (\mathbf{A}\mathbf{A}^*)^q\mathbf{A}\mathbf{\Omega}$ tends to be an ill-conditioned matrix, which leads to a great loss of orthogonality in the matrix $\mathbf{Q}$ during the QMGS procedure. However, the low-rank approximation problem only captures the dominant SVD triplets, the target rank is usually small, and in the randomized algorithm we usually deal with the QMGS of a well-conditioned matrix, the QMGS with $q = 1$ is preferred, since it is more efficient than the quaternion Householder QR.

TABLE 5.1
The peak signal-to-noise ratio and relative approximating errors for randsvdQ

k	q	PNSR	$\frac{\|\widehat{\mathbf{A}}_k^{(q)}-\mathbf{A}\|_2}{\|\mathbf{A}\|_2}$	$\frac{\|\widehat{\mathbf{A}}_k^{(q)}-\mathbf{A}\|_F}{\|\mathbf{A}\|_F}$
50	1	24.7780	0.0115	0.0602
	2	25.0501	0.0106	0.0583
100	1	29.4102	0.0057	0.0353
	2	29.7303	0.0051	0.0340
150	1	32.8041	0.0035	0.0239
	2	33.1368	0.0032	0.0230

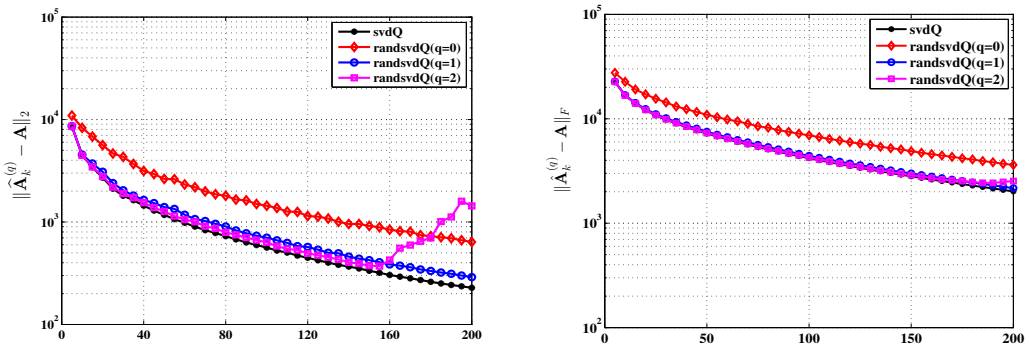

FIG. 5.5. *Errors incurred for different power schemes, in which the orthonormal basis* $\mathbf{Q}$ *in* randsvdQ *is obtained via quaternion MGS.*

EXAMPLE 5.3. In this example, we compare numerical behaviors of randeigQ and prandsvdQ algorithms in computing the rank-k approximation of a large quaternion Hermitian matrix. It is well known that the real Laplacian matrix plays important roles in image denoising, inpainting problems for the grayscale image. Recently in [1], complex Laplacian matrix is also discussed in the mixed graph with some directed and some undirected edges, and its zero eigenvalue is proved to be related to the connection of the mixed graph. Our example involves a quaternion graph Laplacian matrix for a color image, which is modified from real [11] and complex cases.

For this purpose, we begin resizing lena512 to a 60×60-pixel color image, owing to the restricted memory of Laptop. For each pixel i in color channel $s\in\{r,g,b\}$, form a vector $x_s^{(i)}\in\mathbb{R}^{25}$ by gathering the 25 intensities of the pixels in a 5×5 neighborhood centered at pixel i. Next, we form a 3600×3600 pure quaternion Hermitian weight matrix $\mathbf{W}=W_r\mathbf{i}+W_g\mathbf{j}+W_b\mathbf{k}$ with $\mathbf{w}_{ji}=\mathbf{w}_{ij}^*$, $\mathbf{w}_{ii}=0$, and $\mathbf{w}_{ij}=(w_r)_{ij}\,\mathbf{i}+(w_g)_{ij}\,\mathbf{j}+(w_b)_{ij}\,\mathbf{k}$ for $i<j$, which is determined by

$$(w_s)_{ij}=\exp\left\{-\|x_s^{(i)}-x_s^{(j)}\|_2^2/\sigma_s^2\right\},\quad j>i,\quad s\in\{r,g,b\}.$$

Here the entries in their strictly upper triangular part of W_s reflect the similarities between patches, and the parameter σ_s controls the level of sensitivity in each channel. By zeroing out all entries of skew-symmetric matrices W_r, W_g and W_b except the four largest ones in magnitude in each row, we obtain sparse weight matrices $\widetilde{W}_s$ and $\widetilde{\mathbf{W}}$. Similar to the complex case, let D be a diagonal matrix with $d_{ii}=\sum_j|\mathbf{w}_{ij}|$, and define the quaternion Laplacian matrix $\mathbf{L}$ as

$$\mathbf{L}=I-D^{-1/2}\widetilde{\mathbf{W}}D^{-1/2}.$$

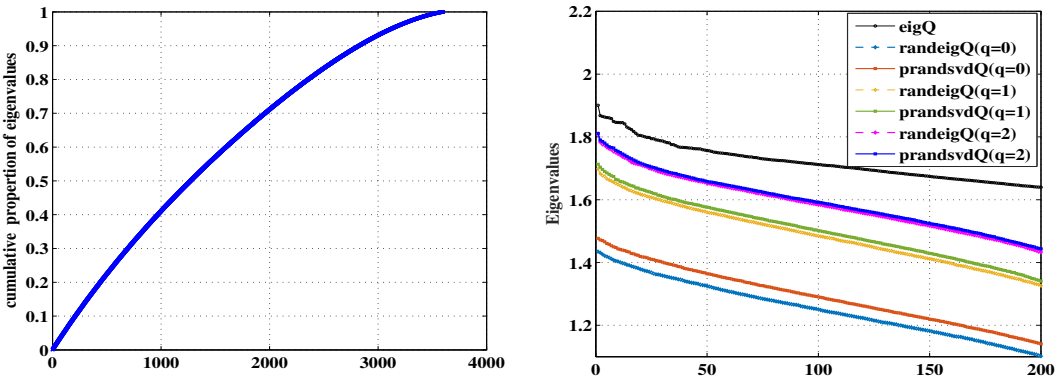

FIG. 5.6. *The cumulative proportion of eigenvalues of a quaternion Laplacian matrix and eigenvalues computed via* randeigQ *and* prandsvdQ *for* $k = 200, p = 10$.

For all $s \in \{r, g, b\}$, take $\sigma_s = 50$, store the 14400×3600 real matrix $\mathbf{L}_c$, and use structure-preserving algorithm eigQ [13] to compute all eigenvalues of $\mathbf{L}$. Here the Hermitian matrix $\mathbf{L}$ is a very extreme case with positive eigenvalues, and the smallest ratio σ_{k+1}/σ_k of adjacent eigenvalues (singular values) of $\mathbf{L}$ is greater than 0.98.

Take $k = 200, p = 10, q = 0, 1, 2$ to compare the eigenvalues of $\mathbf{L}$ via randeigQ, prandsvdQ. In all cases, the approximations of eigenvalues are not good enough, because $k = 200$ only captures less than 10% proportion of eigenvalues in this extreme case, as revealed in the left figure of Figure 5.6. Due to the quite slow decay rate of eigenvalues, when q is small, say for $q = 0$, the eigenvalues computed via randeigQ, prandsvdQ are not accurate enough, but prandsvdQ still approximates eigenvalues better than randeigQ, as predicted in Remark 3.3. The accuracy is improved as q increases, and for this extreme example, $q = 2$ is sufficient to guarantee the eigenvalues from two algorithms with almost the same accuracy. For general cases, we believe that randeigQ is as reliable as prandsvdQ but more efficient for practical low-rank Hermitian matrix approximation problems with dominant singular values.

EXAMPLE 5.4. In this example, we consider the color face recognition problem [15] based on color principal component analysis (CPCA) approach. Suppose that there are s training color image samples, denoted by $m \times n$ pure quaternion matrices $\mathbf{F}_1, \mathbf{F}_2, \ldots, \mathbf{F}_s$, and the average is $\mathbf{\Psi} = \frac{1}{s}\sum_{t=1}^{s} \mathbf{F}_t \in \mathbb{Q}^{m\times n}$. Let $\mathbf{X} = [\mathrm{vec}(\mathbf{F}_1) - \mathrm{vec}(\mathbf{\Psi}), \cdots, \mathrm{vec}(\mathbf{F}_s) - \mathrm{vec}(\mathbf{\Psi})]$, where $\mathrm{vec}(\cdot)$ means to stack the columns of a matrix into a single long vector. The core work of CPCA approach is to compute the left singular vectors corresponding to the first k largest singular values of $\mathbf{X}$, which are called the eigenfaces. The eigenfaces can also be obtained from the eigQ algorithm [13] applied to $\mathbf{X}\mathbf{X}^*$ or $\mathbf{X}\mathbf{X}^*$.

For color image samples, we use the Georgia Tech face database[2], and all images are manually cropped, and then resized to 120×120 pixels. The samples of the cropped images are shown in Figure 5.7. There are 50 persons to be used. The first ten face images per individual person are chosen for training and the remaining five face images are used for testing. The number of chosen eigenfaces, k, increases from 1 to 30. We need to compute k SVD triplets of a 14400×500 quaternion matrix $\mathbf{X}$, in which the 14400 rows refer to 120×120 pixels and the 500 columns refer to 50 persons with 10 faces each.

As revealed in [15], the matrix is very large and the svdQ algorithm does not finish the computation of the singular value decomposition of $\mathbf{X}$ in 2 hours and eigQ needs about seven times

[2]The Georgia Tech face database. http://www.anefian.com/research/face_reco.htm

FIG. 5.7. *Sample images for one individual of the Georgia Tech face database*

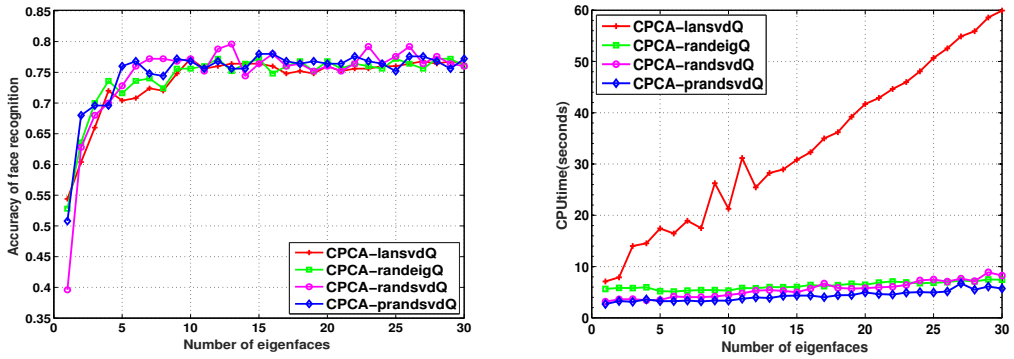

FIG. 5.8. *The color face recognition accuracy and CPU time by* lansvdQ*,* randsvdQ*,* randeigQ *and* prandsvdQ *methods with parameters* $p = 4, q = 0$.

of the running CPU time via the quaternion Lanczos-based algorithm (lansvdQ)[3]. In this experiment we consider the lansvdQ, randsvdQ, prandsvdQ algorithms of $\mathbf{X}$, and randeigQ algorithm of $\mathbf{X}^*\mathbf{X}$, where the orthonormal basis is derived based on quaternion MGS process, and in randeigQ, the matrix $\mathbf{X}^*\mathbf{X}$ is not explicitly formed. The detailed comparisons of recognition accuracy and running CPU time of candidate methods are depicted in Figure 5.8, in which the accuracy of face recognition is the percentage of correctly recognized persons for given 250 test images. For $p = 4$ and $q = 0$, randomized algorithms have higher recognition accuracy than lansvdQ, and are much more efficient than lansvdQ. Moreover, the preconditioning technique for randsvdQ can slightly enhance the efficiency of the algorithm. Unlike lansvdQ, the CPU time for randomized algorithms does not increase significantly with the target rank (number of eigenfaces). lansvdQ is much less efficient partly because it uses for-end loop and performs matrix-vector products at each iteration, while the randomized algorithms make full use of the matrix-matrix products that have been highly optimized for maximum efficiency on modern serial and parallel architectures [8].

EXAMPLE 5.5. In this example, we generalize the fast frequent directions via subspace embedding (SpFD) method [34] to the quaternion case. The corresponding algorithm is referred to as SpFDQ, and is compared with prandsvdQ through the color face recognition problem in Example 5.4.

[3]https://hkumath.hku.hk/∼mng/mng_files/LANQSVDToolbox.zip

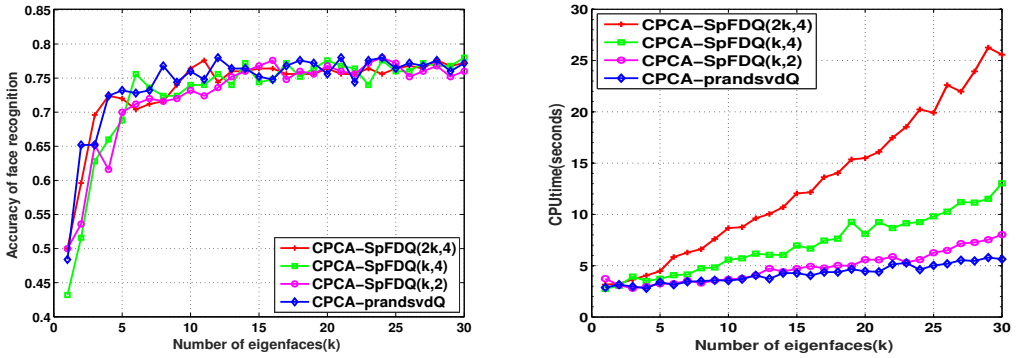

FIG. 5.9. *The color face recognition accuracy and CPU time by* SpFDQ(ℓ,t) *and* prandsvdQ *methods with parameters* $p=4, q=0$.

Given a real matrix $A\in\mathbb{R}^{m\times n}$ $(m\geq n)$, the SpFD(ℓ,t) algorithm squeezes the rows of A by pre-multiplying SP on A, where t is assumed to be a factor of m (if not, append zero rows to the end of A until m is), P is a random permutation matrix, and $S=\mathrm{diag}(S_1,\ldots,S_t)$ is a sparse sketching matrix with $S_i\in\mathbb{R}^{\ell\times\frac{m}{t}}$ being generated on a probability distribution. At the start of the algorithm, it extracts and shrinks the top ℓ important right singular vectors of a two-layered matrix $\begin{bmatrix}S_1PA\\S_2PA\end{bmatrix}$ via SVD, and then combines them with the next ℓ rows in SPA to form a new two-layered matrix. Repeat the procedure until the last ℓ rows of SPA is combined into the computation. Finally, an orthonormal basis $V_\ell\in\mathbb{R}^{n\times\ell}$ for the row space of SPA is obtained, and a rank-k approximation of A is derived based on the SVD of AV_ℓ. The algorithm consists of $(t-1)$ iterations, and the total cost is

$$2\mathrm{nnz}(A)(\ell+1)+[24n\ell^2+160\ell^3](t-1)+6m\ell^2+20\ell^3+2m\ell k+2mn\ell,\quad t>1,$$

where $m\geq n\geq\ell\geq k$, $m\gg\ell$. The choice of $t=2,\ell=k$ corresponds to an algorithm with the cheapest cost, while for $t=\lceil m/\ell\rceil$, SpFD(ℓ,t) reduces to a slight modification of FD in [7].

In the SpFDQ(ℓ,t) algorithm, $\mathbf{A}$ is taken to be the 14400×500 matrix $\mathbf{X}$ in Example 5.4, and the choice of sketching matrix S is the same as the real case. To perform a fair comparison, we also consider the preconditioned technique in the QSVD of a short-and-wide or tall-and-narrow quaternion matrix. During the $(t-1)$ rounds of QSVD in the iteration, due to the potential singularity of the sketching matrix S_i that might lead to a singular two-layered matrix, we apply quaternion Householder QR first and then implement the QSVD on a small-size matrix. In the last round of QSVD of $\mathbf{AV}_\ell$, the QSVD of $\mathbf{AV}_\ell$ is obtained via the QMGS of $\mathbf{AV}_\ell$ first and then applying QSVD to a small upper triangular factor.

The accuracy of face recognition and running CPU time of SpFDQ(ℓ,t) and prandsvdQ algorithms are shown in Figure 5.9. The depicted results demonstrate that SpFDQ$(k,2)$ is the most efficient one among all SpFDQ(ℓ,t) algorithms, while prandsvdQ is a little more efficient than SpFDQ$(k,2)$ when k increases. For the recognition accuracy, prandsvdQ has higher recognition accuracy for most parameter values of k, while there also exists a parameter, say for $k=22$, prandsvdQ has lower recognition accuracy than other candidate methods. That is partly because the sketching matrix S and random $\mathbf{\Omega}$ are randomly generated on specific distributions, and the recognition accuracy is sometimes affected by the properties of some specific random matrices.

In order to perform a fair comparison, in Table 5.2 we execute each algorithm 20 times, and display the average (avrg), maximal (max) and minimal (min) numbers of correctly recognized persons among 250 test faces for 50 persons, and the average running CPU time (avtime) is also

given. It is observed that when k is small, say for $k \leq 9$, there exist big fluctuations on the recognition accuracy of SpFDQ(k,2), and the average numbers of recognized faces increase when the sketching size in SpFDQ($2k$,2) is increased, but SpFDQ($2k$,2) still has lower recognition accuracy than prandsvdQ. When k increases, the difference of face recognition accuracy becomes smaller, while for the running time, prandsvdQ is the most efficient.

TABLE 5.2
Comparisons of SpFDQ*(ℓ,2) with* prandsvdQ *for PCA-based color image recognition problems*

SpFDQ(k,2)										
k	3	6	9	12	15	18	21	24	27	30
avrg	153.25	178.65	184.70	188.05	189.80	189.85	190.00	190.80	190.95	192.40
max	184	188	191	194	195	193	195	195	195	196
min	130	169	176	182	183	185	184	187	187	187
avtime	2.97	3.29	3.68	4.10	4.63	5.47	5.62	6.02	6.86	7.45
SpFDQ($2k$,2)										
k	3	6	9	12	15	18	21	24	27	30
avrg	161.35	178.75	184.85	190.30	189.25	190.00	188.95	190.60	190.85	191.80
max	172	186	190	194	192	192	192	193	196	196
min	150	172	180	186	186	187	186	188	186	188
avtime	3.18	3.92	4.72	5.60	6.96	7.91	9.27	10.47	12.09	13.56
prandsvdQ										
k	3	6	9	12	15	18	21	24	27	30
avrg	174.35	187.10	190.55	191.30	190.65	191.15	192.65	193.45	192.25	192.85
max	182	195	201	198	194	198	196	198	197	197
min	164	182	183	185	185	184	187	188	188	189
avtime	3.02	3.30	3.44	3.70	4.11	4.44	4.73	5.02	5.50	5.82

6. Conclusion. In this paper we have presented the randomized QSVD algorithm for quaternion low-rank matrix approximation problems. For large scale problems with a small target rank, the randomized algorithm compresses the size of the input matrix by the quaternion normal distribution-based random sampling, and approximates dominant SVD triplets with good accuracy and high efficiency. The approximation errors of the randomized algorithm are illustrated by the detailed theoretical analysis and numerical examples. Compared to the Lanczos-based QSVD (lansvdQ) and fast frequent direction via subspace embedding (SpFDQ) algorithms, the randomized algorithms display their effectiveness and efficiency for PCA-based color image recognition problems.

Acknowledgments. The authors are grateful to the handling editor and three anonymous referees for their useful comments and suggestions, which greatly improved the original presentation.

REFERENCES

[1] R. B. BAPAT, D. KALITA AND S. PATIB, *On weighted directed graphs*, Linear Algebra Appl., 436 (2012), pp. 99-111.
[2] N. L. BIHAN AND S. J. SANGWINE, *Quaternion principal component analysis of color images*, IEEE International Conference on Image Processing, 1 (2003), pp. 809-812.
[3] Z. Z. CHEN AND J. J. DONGARRA, *Condition numbers of Gaussian random matrices*, SIAM J. Matrix Anal. Appl., 27 (2005), pp. 603-620.

[4] K. L. Clarkson and D. P. Woodruff, *Numerical linear algebra in the streaming model*, STOC '09: Proc. 41st Ann. ACM 476 Symp. Theory of Computing, 2009.
[5] J. A. Duersch and M. Gu, *Randomized projection for rank-revealing matrix factorizations and low-rank approximations*, SIAM Rev., 62 (2020), pp. 661-682.
[6] T. A. Ell, N. L. Bihan and S. J. Sangwine, *Quaternion Fourier Transforms for Signal and Image Processing*, Wiley, Hoboken, NJ, USA, 2014.
[7] M. Ghashami, E. Liberty, J. M. Phillips and D. P. Woodruff, *Frequent directions: simple and deterministic matrix sketching*, SIAM J. Comput., 45 (2016), pp. 1762-1792.
[8] G. H. Golub and C. F. Van Loan, *Matrix Computations(4ed.)*, Johns Hopkins University Press, Baltimore, 2013.
[9] G. H. Golub and C. Reinsch, *Singular value decomposition and least squares solutions*, Numer. Math., 14 (1970), pp. 403-420.
[10] M. Gu, *Subspace iteration randomization and singular value problems*, SIAM J. Sci. Comput., 37 (2015), A1139-A1173.
[11] N. Halko, P. G. Martinsson and J. A. Tropp, *Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions*, SIAM Rev., 53 (2011), pp. 217-288.
[12] W. R Hamilton, *Elements of Quaternions*, Chelsea, New York, 1969.
[13] Z. G. Jia, M. S. Wei and S. T. Ling, *A new structure-preserving method for quaternion Hermitian eigenvalue problems*, J. Comput. Appl. Math., 239 (2013), pp. 12-24.
[14] Z. G. Jia, M. S. Wei, M. X. Zhao and Y. Chen, *A new real structure-preserving quaternion QR algorithm*, J. Comput. Appl. Math., 343 (2018), pp. 26-48.
[15] Z. G. Jia, M. K. Ng and G. J. Song, *Lanczos method for large-scale quaternion singular value decomposition*, Numer. Algorithms, 82 (2019), pp. 699-717.
[16] Z. G. Jia, M. K. Ng and G. J. Song, *Robust quaternion matrix completion with applications to image inpainting*, Numer. Linear Algebra Appl., 26 (2019), e2245.
[17] Z. G. Jia, M. K. Ng and W. Wang, *Color image restoration by saturation-value (SV) total variation*, SIAM J. Imag. Sci., 12 (2019), pp. 972-1000.
[18] Z. G. Jia, *The Eigenvalue Problem of Quaternion Matrix: Structure-Preserving Algorithms and Applications*, Science Press, Beijing, 2019.
[19] Z. G. Jia and M. K. Ng, *Structure preserving quaternion generalized minimal residual method*, SIAM J. Matrix Anal. Appl., 42 (2021), pp. 616-634.
[20] S. M. Li, *A theory of statistical analysis based on normal distrubution of quaternion (in Chinese)*, Ph.D Thesis, Sun Yat-sen University, 2001.
[21] Y. Li, M. S. Wei, F. X. Zhang and J. L. Zhao, *Real structure-preserving algorithms of Householder based transformations for quaternion matrices*, J. Comput. Appl. Math., 305 (2016), pp. 82-91.
[22] Y. Li, M. S. Wei, F. X. Zhang and J. L. Zhao, *A structure-preserving method for the quaternion LU decomposition*, Calcolo, 54 (2017), pp. 1553-1563.
[23] E. Liberty, F. Woolfe, P. Martinsson, V. Rokhlin and M. Tygert, *Randomized algorithms for the low-rank approximation of matrices*, Proceedings of the National Academy of Sciences, 104 (2007), pp. 20167-20172.
[24] M. T. Loots, *On the development of the quaternion normal distribution*, Master Thesis, University of Pretoria, Pretoria, 2010.
[25] M. W. Mahoney, *Randomized algorithms for matrices and data*, Foundations and Trends in Machine Learning, 3 (2011), pp. 123-224.
[26] P. G. Martinsson, V. Rokhlin and M. Tygert, *A randomized algorithm for the decomposition of matrices*, Appl. Comput. Harmon. Anal., 30 (2011), pp. 47-68.
[27] X. Meng and M.W. Mahoney, *Low-distortion subspace embeddings in input-sparsity time and applications to robust linear regression*, Proc. 45th Annu. ACM Symp. Theory Comput., 2013, pp. 91-100.
[28] T. Minemoto, T. Isokawa, H. Nishimura and N. Matsui, *Feed forward neural network with random quaternionic neurons*, Signal Processing, 136 (2017), pp. 59-68.
[29] R. J. Muirhead, *Aspects of Multivariate Statistical Theory*, Wiley, New York, NY, 1982.
[30] L. Rodman, *Topics in Quaternion Linear Algebra*, Princeton University Press, 2014.
[31] B. J. Saap, *Randomized algorithms for low rank matrix decomposition*, Technical Report, Computer and Information Science, University of Pennsylvania, 2011.
[32] S. J. Sangwine and N. L. Bihan, *Quaternion singular value decomposition based on bidiagonalization to a real or complex matrix using quaternion Householder transformations*, Appl. Math. Comput. 182 (2006), pp. 727-738.
[33] T. Sarlós, *Improved approximation algorithms for large matrices via random projections*, Proc. 47th Annu. IEEE Symp. Foundations Comput. Sci., 2006, pp. 143-152.
[34] D. Teng and D. L. Chu, *A fast frequent directions algorithm for low rank approximation*, IEEE Trans. Pattern Anal. Mach. Intell., 41 (2019), pp. 1279-1293.
[35] C. Y. Teng and K. T. Fang, *Statistical analysis based on normal distribution of quaternion*, International

Symposium on Contemporary Multivariate Analysis and its Applications, 1997.
[36] C. Y. Teng and S. M. Li, *Exterior differential form on quaternion matrices and its application (in Chinese)*, Acta Scientiarum Naturalium Universitatis SunYatseni, 38 (1999), pp. 12-16.
[37] M. H. Wang, W. H. Ma, *A structure-preserving method for the quaternion LU decomposition in quaternionic quantum theory*, Comput. Phys. Comm., 184 (2013), pp. 2182-2186.
[38] M. S. Wei, Y. Li, F. X. Zhang, J. L. Zhao, *Quaternion Matrix Computations*, Nova Science Publishers, 2018.
[39] F. Woolfe, E. Liberty, V. Rokhlin and M. Tygert, *A fast randomized algorithm for the approximation of matrices*, Appl. Comput. Harm. Anal., 25 (2008), pp. 335-366.
[40] W. J. Yu, Y. Gu and Y. H. Li, *Efficient randomized algorithms for the fixed-precision low-rank matrix approximation*, SIAM J. Matrix Anal. Appl., 39 (2018), pp. 1339-1359.
[41] F. Z. Zhang, *Quaternions and matrices of quaternion*, Linear Algebra Appl., 251 (1997), pp. 21-57.
[42] L. P. Zhang and Y. M. Wei, *Randomized core reduction for discrete ill-posed problem*, J. Comput. Appl. Math., 375 (2020), 112797.
[43] M. X. Zhao, Z. G. Jia, Y. F. Cai, X. Chen and D. W. Gong, *Advanced variations of two-dimensional principal component analysis for face recognition*, Neurocomputing, 452 (2021), pp. 653-664.