A Structure-Preserving One-Sided Jacobi Method for Computing the SVD of a Quaternion Matrix

Ru-Ru Ma*       Zheng-Jian Bai†

Abstract

In this paper, we provide a structure-preserving one-sided cyclic Jacobi method for computing the singular value decomposition of a quaternion matrix. In this method, the columns of the quaternion matrix are orthogonalized in pairs by using a sequence of orthogonal JRS-symplectic Jacobi matrices to its real counterpart. The quadratic convergence is also established under some mild conditions. Numerical tests are reported to illustrate the efficiency of the proposed method.

Keywords. Quaternion matrix; singular value decomposition; one-sided cyclic Jacobi method; Color image compression

AMS subject classifications. 65F18, 65F15, 15A18, 65K05, 90C26, 90C48

1 Introduction

The concept of quaternions was originally introduced by Hamilton [10]. Quaternions and quaternion matrices arise in various applications in applied science such as quaternionic quantum mechanics [1, 2, 6], color image processing [32] and field theory [19], etc. The quaternion matrix singular value decomposition (QSVD) was studied theoretically in 1997 by Zhang [35]. Recently, the QSVD has been an important tool in many applications such as color image processing [7, 17, 22, 24], signal processing [16, 33], and electroencephalography [5], etc.

Various numerical methods have been proposed for computing the QSVD. In [15, 16, 24], some algorithms were provided to calculate the singular value decomposition (SVD) of a quaternion matrix via utilizing its equivalent complex matrix. In [27], Sangwine and Le Bihan proposed a method for computing the QSVD based on bidiagonalization via quaternionic Householder

*School of Mathematical Sciences, Xiamen University, Xiamen 361005, People’s Republic of China (maruru7271@126.com). The research of this author is partially supported by the Fundamental Research Funds for the Central Universities (No. 20720180008).

†Corresponding author. School of Mathematical Sciences and Fujian Provincial Key Laboratory on Mathematical Modeling & High Performance Scientific Computing, Xiamen University, Xiamen 361005, People’s Republic of China (zjbai@xmu.edu.cn). The research of this author is partially supported by the National Natural Science Foundation of China (No. 11671337), the Natural Science Foundation of Fujian Province of China (No. 2016J01035), and the Fundamental Research Funds for the Central Universities (No. 20720180008).
transformations. In [18], Le Bihan and Sangwine gave an implicit Jacobi algorithm for computing the QSVD where the quaternion arithmetic is employed instead of a complex equivalent representation. In [3], Doukhnitch and Ozen presented a coordinate rotation digital computer algorithm for computing the QSVD.

In this paper, we propose a structure-preserving one-sided cyclic Jacobi method for computing the QSVD. This is motivated by the recent structure-preserving methods related to quaternion matrices. In particular, Jia et al. designed a real structure-preserving method for quaternion Hermitian eigenvalue problems by using the structure-preserving tridiagonalization of the real counterpart for quaternion Hermitian matrices [11]. Li et al. presented a structure-preserving algorithm for computing the QSVD, which uses the structure-preserving bidiagonalization of the real counterpart of quaternion matrices via Householder-based transformations [20, 21]. Ma et al. proposed a structure-preserving Jacobi algorithm for quaternion Hermitian eigenvalue problems [23]. In this paper, the columns of a rectangle quaternion matrix are orthogonalized in pairs by using a sequence of orthogonal JRS-symplectic Jacobi matrices to its real counterpart. When the updated quaternion matrix has sufficiently orthogonal columns, the SVD is obtained by column scaling. Numerical experiments show that our method is more efficient than the implicit Jacobi algorithm in [18] and gives almost the same singular values as the solver in quaternion toolbox for MATLAB [26].

The rest of this paper is organized as follows. In section 2 we give necessary preliminaries used in this paper. In section 3 a structure-preserving one-sided cyclic Jacobi algorithm is provided for computing the QSVD and the quadratic convergence is established under some assumptions. In section 4 we report some numerical experiments to indicate the efficiency of our algorithm and apply it to color image compression. Finally, some concluding remarks are given in section 5.

2 Preliminaries

In this section, we briefly review some necessary definitions and properties of quaternions and quaternion matrices. Quaternions were originally introduced by Hamilton in 1843 [9]. For more information on quaternions, one may refer to [13, 35] and references therein.

Throughout this paper, we need the following notation. Let \( \mathbb{R} \) and \( \mathbb{R}^{m \times n} \) be the set of all real numbers and the set of all \( m \times n \) real matrices respectively. Let \( \mathbb{H} \) and \( \mathbb{H}^{m \times n} \) be the set of all quaternions and the set of all \( m \times n \) quaternion matrices respectively. Let \( I_n \) be the identity matrix of order \( n \). Let \( A^T, \bar{A} \) and \( A^* \) stand for the transpose, conjugate and conjugate transpose of a matrix \( A \) accordingly. \( \| \cdot \|_F \) means the Frobenius matrix norm. For any \( n \times n \) square matrix \( A = (a_{pq}) \), we define

\[
\text{off}(A) := \sqrt{\sum_{p=1}^{n} \sum_{q=1}^{n} |a_{pq}|^2},
\]

where \( |a_{pq}| \) denotes the absolute value of \( a_{pq} \).

A quaternion \( a \in \mathbb{H} \) takes the form of

\[
a = a_0 + a_1 i + a_2 j + a_3 k,
\]
where \(a_0, a_1, a_2, a_3 \in \mathbb{R}\) and the quaternion units \(i, j, k\) satisfy the following rules

\[
i^2 = j^2 = k^2 = -1, \quad jk = -kj = i, \quad ki = -ik = j, \quad ij = -ji = k.
\]

The conjugate of \(a \in \mathbb{H}\) is given by \(\bar{a} = a^* = a_0 - a_1i - a_2j - a_3k\). For two quaternions \(a = a_0 + a_1i + a_2j + a_3k \in \mathbb{H}\) and \(b = b_0 + b_1i + b_2j + b_3k \in \mathbb{H}\), their product (i.e., the Hamilton product) is given by

\[
ab = a_0b_0 - a_1b_1 - a_2b_2 - a_3b_3 + (a_0b_1 + a_1b_0 + a_2b_3 - a_3b_2)i + (a_0b_2 - a_1b_3 + a_2b_0 + a_3b_1)j + (a_0b_3 + a_1b_2 - a_2b_1 + a_3b_0)k.
\]

The absolute value \(|a|\) of \(a = a_0 + a_1i + a_2j + a_3k \in \mathbb{H}\) is defined by

\[
|a| = \sqrt{a_0^2 + a_1^2 + a_2^2 + a_3^2} = \sqrt{\bar{a}a} = \sqrt{aa}.
\]

The multiplicative inverse of any nonzero quaternion \(0 \neq a \in \mathbb{H}\) is given by \(a^{-1} = \bar{a}/|a|^2\).

The Hamilton product is not commutative but associative and thus \(\mathbb{H}\) is an associative division algebra over \(\mathbb{R}\).

Suppose \(A = A_0 + A_1i + A_2j + A_3k \in \mathbb{H}^{m \times n}\) is a quaternion matrix, where \(A_0, A_1, A_2, A_3 \in \mathbb{R}^{m \times n}\). A real counterpart of \(A\) is defined by

\[
\Gamma_A = \begin{bmatrix} A_0 & A_2 & A_1 & A_3 \\ -A_2 & A_0 & A_3 & -A_1 \\ -A_1 & -A_3 & A_0 & A_2 \\ -A_3 & A_1 & -A_2 & A_0 \end{bmatrix}
\]

(2.1)

Next, we recall the definitions of JRS-symmetry and JRS-symplecticity [11]. Let

\[
J_n = \begin{bmatrix} 0 & 0 & -I_n & 0 \\ 0 & 0 & 0 & -I_n \\ I_n & 0 & 0 & 0 \\ 0 & I_n & 0 & 0 \end{bmatrix}, \quad R_n = \begin{bmatrix} 0 & -I_n & 0 & 0 \\ I_n & 0 & 0 & 0 \\ 0 & 0 & 0 & I_n \\ 0 & 0 & -I_n & 0 \end{bmatrix}, \quad S_n = \begin{bmatrix} 0 & 0 & 0 & -I_n \\ 0 & 0 & I_n & 0 \\ 0 & -I_n & 0 & 0 \\ I_n & 0 & 0 & 0 \end{bmatrix}.
\]

A matrix \(\Omega \in \mathbb{R}^{4n \times 4n}\) is called JRS-symmetric if \(J_n\Omega J_n^T = \Omega\), \(R_n\Omega R_n^T = \Omega\) and \(S_n\Omega S_n^T = \Omega\). A matrix \(\Omega \in \mathbb{R}^{4n \times 4n}\) is called JRS-symplectic if \(\Omega J_n\Omega^T = J_n\), \(\Omega R_n\Omega^T = R_n\) and \(\Omega S_n\Omega^T = S_n\). A matrix \(\Omega \in \mathbb{R}^{4n \times 4n}\) is called orthogonal JRS-symplectic if it is orthogonal and JRS-symplectic.

In the rest of this section, we recall some basic results on the relationship between a quaternion matrix and its real counterpart. First, we have the following properties of the real counterparts of quaternion matrices [11] [12] [29].

**Lemma 2.1** Let \(F, G \in \mathbb{H}^{m \times n}\), \(H \in \mathbb{H}^{m \times s}\), \(W \in \mathbb{H}^{m \times n}\), \(\alpha \in \mathbb{R}\). Then

1. \(\Gamma_{F+G} = \Gamma_F + \Gamma_G; \ \Gamma_{\alpha G} = \alpha \Gamma_G; \ \Gamma_{GH} = \Gamma_G \Gamma_H\).
2. \(\Gamma_{G^*} = \Gamma_G^T\).
(3) $\Gamma_W$ is JRS-symmetric.

(4) $W$ is unitary if $\Gamma_W$ is orthogonal.

(5) If $\Gamma_G$ is orthogonal, then it is also orthogonal JRS-symplectic.

On the SVD of a quaternion matrix, we have the following result [35, Theorem 7.2].

**Lemma 2.2** Let $A \in \mathbb{H}^{m \times n}$ be a quaternion matrix with $\text{rank}(A) = r$. Then there exist unitary quaternion matrices $U \in \mathbb{H}^{m \times m}$ and $V \in \mathbb{H}^{n \times n}$ such that

$$U^*AV = \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix},$$

(2.2)

where $\Sigma_r = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_r)$ and $\{\sigma_w\}_{w=1}^r$ are the positive singular values of $A$.

Finally, we have the following result on the equivalence between the eigenvalue problem of a quaternion matrix and the eigenvalue problem of its real counterpart [11].

**Lemma 2.3** Let $A = X + Yj$ be a quaternion matrix, where $X = A_0 + A_1i$ and $Y = A_2 + A_3i$ with $A_0, A_1, A_2, A_3 \in \mathbb{R}^{n \times n}$. Then there exists a unitary quaternion matrix

$$Q = \frac{1}{2}\begin{bmatrix} I_n & -jI_n & -iI_n & -kI_n \\ I_n & jI_n & -iI_n & kI_n \\ I_n & -jI_n & iI_n & kI_n \\ I_n & jI_n & iI_n & -kI_n \end{bmatrix}$$

such that

$$\Gamma_A = Q^*\begin{bmatrix} X + Yj & 0 & 0 & 0 \\ 0 & X - Yj & 0 & 0 \\ 0 & 0 & \bar{X} + \bar{Y}j & 0 \\ 0 & 0 & 0 & \bar{X} - \bar{Y}j \end{bmatrix}Q.$$  (2.3)

### 3 Structure-preserving one-sided cyclic Jacobi algorithm

In this section, we present a structure-preserving one-sided cyclic Jacobi algorithm for computing the SVD of a quaternion matrix $A = A_0 + A_1i + A_2j + A_3k \in \mathbb{H}^{m \times n}$, where $A_0, A_1, A_2, A_3 \in \mathbb{R}^{m \times n}$. The proposed structure-preserving one-sided cyclic Jacobi algorithm involves a sequence of orthogonal JRS-symplectic transformations $\Gamma_A \leftarrow \Gamma_A\Gamma_G$ such that the updated $\Gamma_A$ is closer to a column-orthogonal matrix than its predecessor. When the updated $\Gamma_A$ has sufficiently orthogonal columns, the column scaling of the updated $A$ leads to the SVD of $A$.

For simplicity, we assume that $m \geq n$. The real counterpart $\Gamma_A$ of $A$ is defined by (2.1). A one-sided cyclic Jacobi algorithm includes (a) choosing an index pair $(p, q)$ such that $1 \leq p < q \leq n$, (b) computing a cosine-sine group $(c_r, s_0, s_1, s_2, s_3)$ such that

$$G(p, q, \theta) = I_n + [e_p, e_q]^T\begin{bmatrix} c_r - 1 & s \\ -s & c_r - 1 \end{bmatrix}\begin{bmatrix} e_p^T \\ e_q^T \end{bmatrix} \in \mathbb{H}^{n \times n}$$

(3.1)
Proof. Note that such that the unitary quaternion matrix given by column of an \( m \times n \) matrix, i.e.,

\[
\begin{bmatrix}
\tilde{A}_0, \tilde{A}_2, \tilde{A}_1, \tilde{A}_3
\end{bmatrix}, \quad \tilde{A}_w \in \mathbb{R}^{m \times n} \quad w = 0, 1, 2, 3,
\]
we get the updated quaternion matrix \( \tilde{A} = \tilde{A}_0 + \tilde{A}_1 \mathbf{i} + \tilde{A}_2 \mathbf{j} + \tilde{A}_3 \mathbf{k} = AV \), where \( V = G^{(1)}G^{(2)} \cdots G^{(0)} \) is an \( n \times n \) unitary quaternion matrix. Finally, the column scaling of \( \tilde{A} \) yields the SVD of \( A \):

\[
AV = \tilde{A} = U \Sigma,
\]
where \( \Sigma = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n) \) with \( \sigma_w \geq 0 \) for \( w = 1, \ldots, n \) and \( U \in \mathbb{H}^{m \times n} \) satisfies \( U^*U = I_n \).

The following theorem presents the orthogonalization of any two columns of an \( m \times n \) quaternion matrix.

**Theorem 3.1** Let \( A(p, q) = [a_p, a_q] \), where \( a_w = a_{w0} + a_{w1} \mathbf{i} + a_{w2} \mathbf{j} + a_{w3} \mathbf{k} \in \mathbb{H}^m \) is the \( w \)-th column of an \( m \times n \) quaternion matrix \( A \) for \( w = p, q \). If \( a_p^*a_q \neq 0 \), then there exists a 2-by-2 unitary quaternion matrix given by

\[
G(p, q; \theta) = \begin{bmatrix}
c_r & s \\
-\bar{s} & c_r
\end{bmatrix}
\]

such that \( \tilde{A}(p, q) := A(p, q)G(p, q; \theta) \) has orthogonal columns, where \( c_r = \cos(\theta) \in \mathbb{R} \) and \( s = s_0 + s_1 \mathbf{i} + s_2 \mathbf{j} + s_3 \mathbf{k} \in \mathbb{H} \) with \( c_r^2 + |s|^2 = 1 \).

**Proof.** Note that

\[
\tilde{A}(p, q) = A(p, q)G(p, q; \theta)
\]

\[
= [a_p, a_q] \begin{bmatrix}
c_r & s \\
-\bar{s} & c_r
\end{bmatrix}
\]

\[
= [c_r a_p - a_q \bar{s}, a_p \bar{s} + c_r a_q]
\]

\[
:= [\tilde{a}_p, \tilde{a}_q]. \tag{3.3}
\]

By hypothesis \( a_p^*a_q \neq 0 \) and thus \( |a_p^*a_q| > 0 \). Define \( c_r \in \mathbb{R} \) and \( s = s_0 + s_1 \mathbf{i} + s_2 \mathbf{j} + s_3 \mathbf{k} \in \mathbb{H} \) by

\[
s_0 = \frac{\sin(\theta)}{|a_p^*a_q|} a_{pq0}, \quad s_1 = \frac{\sin(\theta)}{|a_p^*a_q|} a_{pq1}, \quad s_2 = \frac{\sin(\theta)}{|a_p^*a_q|} a_{pq2}, \quad s_3 = \frac{\sin(\theta)}{|a_p^*a_q|} a_{pq3}, \quad \sin(\theta) = tc_r,
\]

\[
c_r = \cos(\theta) = \frac{1}{\sqrt{1+t^2}}, \quad |s| = \frac{|t|}{\sqrt{1+t^2}}, \quad t = \begin{cases}
\frac{1}{\tau + \sqrt{1+\tau^2}}, & \text{if } \tau \geq 0 \\
\frac{1}{\tau - \sqrt{1+\tau^2}}, & \text{if } \tau < 0
\end{cases},
\]

\[
\tau = \frac{a_p^*a_q - a_q^*a_p}{2|a_p^*a_q|}, \tag{3.4}
\]

5
where \( a^*_p a_q := a_{pq0} + a_{pq1} i + a_{pq2} j + a_{pq3} k \) with

\[
\begin{align*}
    a_{pq0} &= a^T_{p0} a_{q0} + a^T_{p1} a_{q1} + a^T_{p2} a_{q2} + a^T_{p3} a_{q3}, \\
    a_{pq1} &= a^T_{p0} a_{q1} - a^T_{p1} a_{q0} - a^T_{p2} a_{q3} + a^T_{p3} a_{q2}, \\
    a_{pq2} &= a^T_{p0} a_{q2} + a^T_{p1} a_{q3} - a^T_{p2} a_{q0} - a^T_{p3} a_{q1}, \\
    a_{pq3} &= a^T_{p0} a_{q3} - a^T_{p1} a_{q2} + a^T_{p2} a_{q1} - a^T_{p3} a_{q0}.
\end{align*}
\]

It is easy to see that \( G(p, q; \theta) \) is unitary and

\[
s = \frac{tc_r}{|a^*_p a_q|} \times a^*_p a_q \quad \text{and} \quad t^2 + 2\tau t - 1 = 0.
\]

Thus,

\[
\tilde{a}^*_p \tilde{a}_q = (c_r a_p - a_q s)(a_p s + c_r a_q)
= c_r a^*_p a_p s + c_r^2 a^*_p a_q - sa^*_p a_p s - c_r a^*_q a_q s
= c_r \times a^*_p a_p \times \frac{tc_r}{|a^*_p a_q|} \times a^*_q a_q + c_r^2 \times a^*_p a_q - c_r \times a^*_q a_q \times \frac{tc_r}{|a^*_p a_q|} \times a^*_p a_q
- \frac{tc_r}{|a^*_p a_q|} \times a^*_p a_q \times a^*_q a_p \times \frac{tc_r}{|a^*_p a_q|} \times a^*_p a_q
= -c_r^2 \times a^*_p a_q \times (t^2 + \frac{a^*_q a_q - a^*_p a_p}{|a^*_p a_q|} t - 1)
= -c_r^2 \times a^*_p a_q \times (t^2 + 2\tau t - 1)
= 0.
\]

\[\text{(3.5)}\]

**Corollary 3.2** If the assumptions in Theorem 3.1 hold, then there exists a 2-by-2 unitary quaternion matrix

\[
G(p, q; \theta) = \begin{bmatrix} c_r & s \\ -\bar{s} & c_r \end{bmatrix}, \quad c_r = \cos(\theta) \in \mathbb{R} \text{ and } s = s_0 + s_1 i + s_2 j + s_3 k \in \mathbb{H}
\]

such that

\[
\Gamma_{G(p, q; \theta)}^T \Gamma_{A(p, q)}^T \Gamma_{A(p, q)} G(p, q; \theta) = \tilde{B}(p, q; p, q) \oplus \tilde{B}(p, q; p, q) \oplus \tilde{B}(p, q; p, q) \oplus \tilde{B}(p, q; p, q)
\]

for some

\[
\tilde{B}(p, q; p, q) = \begin{bmatrix} b_{pp} & 0 \\ 0 & b_{qq} \end{bmatrix}.
\]

**Proof.** By Theorem 3.1 we know that \( \tilde{A}(p, q) = A(p, q) G(p, q; \theta) \) has orthogonal columns for \( G(p, q; \theta) \) with \( c_r \) and \( s \) defined as in \[(3.4)\]. From (3.3) and (3.5) we have

\[
\tilde{B}(p, q; p, q) := \tilde{A}^*(p, q) \tilde{A}(p, q) = \begin{bmatrix} \tilde{a}^*_p \tilde{a}_p & \tilde{a}^*_p \tilde{a}_q \\ \tilde{a}^*_q \tilde{a}_p & \tilde{a}^*_q \tilde{a}_q \end{bmatrix} = \begin{bmatrix} \tilde{a}^*_p \tilde{a}_p & 0 \\ 0 & \tilde{a}^*_q \tilde{a}_q \end{bmatrix}.
\]
This shows that $\tilde{B}(p, q; p, q)$ is a real diagonal matrix, where $\tilde{a}_p = c_r a_p - a_q s$ and $\tilde{a}_q = a_p s + c_r a_q$ with $c_r$ and $s$ being defined in \((3.4)\). Using Lemma \(2.1\) and \((3.6)\) we find

\[
\Gamma_{G(p, q; \theta)}^T \Gamma_A(p, q) \Gamma_{G(p, q; \theta)} = \Gamma_{\tilde{A}(p, q)} = \Gamma_{\tilde{A}(p, q)}(p, q) = \Gamma_{\tilde{B}(p, q; p, q)} = \tilde{B}(p, p; q, q) \oplus \tilde{B}(p, q; p, p) \oplus \tilde{B}(p, q; p, q),
\]

where $b_{pp} = \tilde{a}_p^* \tilde{a}_p$ and $b_{qq} = \tilde{a}_q^* \tilde{a}_q$.

\[\square\]

**Remark 3.3** For $t$ defined in \((3.4)\), we see that the rotation angle satisfies $|\theta| \leq \pi/4$. Also, from \((3.3)\) and \((3.6)\) we have

\[
\begin{bmatrix}
\tilde{a}_p^* a_p & 0 \\
0 & \tilde{a}_q^* a_q
\end{bmatrix} = \begin{bmatrix}
\tilde{A}^*(p, q) & \tilde{A}(p, q)
\end{bmatrix} = G^*(p, q; \theta) A^*(p, q) A(p, q) G(p, q; \theta) = \begin{bmatrix}
c_r & a_p^* a_p & a_q^* a_q \\
0 & -s & c_r
\end{bmatrix} \begin{bmatrix}
\tilde{a}_p^* a_p & \tilde{a}_q^* a_q
\end{bmatrix} = \begin{bmatrix}
c_r & a_p^* a_p & a_q^* a_q \\
0 & -s & c_r
\end{bmatrix}.
\]

Since the Frobenius norm is unitary invariant we obtain

\[
(a_p^* a_p)^2 + (a_q^* a_q)^2 = (a_p^* a_p)^2 + 2|a_p^* a_q|^2 + (a_q^*)^2.
\]

**Remark 3.4** We observe from Theorem 3.1 that the matrix $\tilde{A}^* \tilde{A}$ agrees with $A^* A$ except in the $p$-th and $q$-th rows and the $p$-th and $q$-th columns. Let $A = [a_1, a_2, \ldots, a_n]$ and $A = [\tilde{a}_1, \tilde{a}_2, \ldots, \tilde{a}_n]$. Using \((3.7)\) we have

\[
\text{off}(\tilde{A}^* \tilde{A})^2 = \|A^* A\|^2_F - \sum_{t=1}^{n} (\tilde{a}_t^* \tilde{a}_t)^2
\]

\[
= \|A^* A\|^2_F - \sum_{t=1}^{n} (a_t^* a_t)^2 + ((a_p^* a_p)^2 + (a_q^* a_q)^2 - (\tilde{a}_p^* \tilde{a}_p)^2 - (\tilde{a}_q^* \tilde{a}_q)^2)
\]

\[
= \text{off}(A^* A)^2 - 2|a_p^* a_q|^2.
\]

Since $\|\Gamma_{G(p, q)}^T \Gamma_A\|^2_F = 4\|A^* A\|^2_F$, we know that $\text{off}(\Gamma_A^T \Gamma_A)^2 = 4\text{off}(A^* A)^2$. Using \((3.8)\), this implies that $\Gamma_A^T \Gamma_A$ is closer to a diagonal matrix with each orthogonal JRS-symplectic Jacobi rotation.

Based on Theorem 3.1 and Corollary 3.2, we present the following algorithm for generating a 2-by-2 unitary quaternion Jacobi matrix for orthogonalizing any two columns of an $m \times n$ quaternion matrix. This algorithm needs $96m + 30$ operations.

**Algorithm 3.5** Given $A(p, q) = [a_p, a_q]$, where $a_w = a_{w0} + a_{w1}i + a_{w2}j + a_{w3}k \in \mathbb{H}^m$ is the $w$-th column of an $m \times n$ quaternion matrix $A$ for $w = p, q$, this algorithm computes a cosine-sine group $(c_r, s_0, s_1, s_2, s_3)$ such that $\tilde{A}(p, q) = A(p, q) G(p, q; \theta)$ has two orthogonal columns.
Algorithm 3.6 gives the following basic iterative scheme:

\[
\Gamma_A^{(\ell+1)} = \Gamma_A^{(\ell)} \Gamma_G^{(\ell)}, \quad \ell = 0, 1, 2, \ldots, \tag{3.9}
\]
where \( A^{(0)} = A \) and \( G^{(\ell)} \in \mathbb{H}^{n \times n} \) is a unitary quaternion matrix defined in (3.1) with the cosine
sine group \( (c_r, s_0, s_1, s_2, s_3) \) being generated by Algorithm [3.5] Algorithm 3.6 may be seen as a
structure-preserving Jacobi algorithm for solving the eigenvalue problem of an \( n \times n \) quaternion
Hermitian matrix \( B := A^* A \) as in [23]:
\[
\Gamma_{B^{(\ell + 1)}} = \Gamma_A^{T} \Gamma_{A^{(\ell + 1)}} = \Gamma_{G^{(\ell)}}^{T} \Gamma_{A^{(\ell)}} \Gamma_{G^{(\ell)}}
\]
\[
= \Gamma_{G^{(\ell)}} \Gamma_{A^{(\ell)}} \Gamma_{G^{(\ell)}} = \Gamma_{G^{(\ell)}} \Gamma_{B^{(\ell)}} \Gamma_{G^{(\ell)}}, \quad \ell = 0, 1, 2, \ldots ,
\]
where \( B^{(0)} = A^* A \). Let
\[
A^{(\ell)} := [a_1^{(\ell)}, \ldots, a_p^{(\ell)}, \ldots, a_q^{(\ell)}, \ldots, a_n^{(\ell)}].
\]
Then
\[
B^{(\ell)} := (A^{(\ell)})^* A^{(\ell)} = \begin{pmatrix}
(a_1^{(\ell)})^* a_1^{(\ell)} & (a_1^{(\ell)})^* a_p^{(\ell)} & (a_1^{(\ell)})^* a_q^{(\ell)} & \cdots & (a_1^{(\ell)})^* a_n^{(\ell)} \\
(a_p^{(\ell)})^* a_1^{(\ell)} & (a_p^{(\ell)})^* a_p^{(\ell)} & (a_p^{(\ell)})^* a_q^{(\ell)} & \cdots & (a_p^{(\ell)})^* a_n^{(\ell)} \\
(a_q^{(\ell)})^* a_1^{(\ell)} & (a_q^{(\ell)})^* a_p^{(\ell)} & (a_q^{(\ell)})^* a_q^{(\ell)} & \cdots & (a_q^{(\ell)})^* a_n^{(\ell)} \\
\vdots & \ddots & \ddots & \ddots & \ddots \\
(a_n^{(\ell)})^* a_1^{(\ell)} & (a_n^{(\ell)})^* a_p^{(\ell)} & (a_n^{(\ell)})^* a_q^{(\ell)} & \cdots & (a_n^{(\ell)})^* a_n^{(\ell)}
\end{pmatrix} \quad (3.10)
\]
From Remark 3.4 we have
\[
\text{off}(B^{(\ell)})^2 = \|B^{(\ell)}\|_F^2 - \sum_{u=1}^{n} (b_{uu}^{(\ell)})^2
\]
\[
= \|(A^{(\ell)})^* A^{(\ell)}\|_F^2 - \sum_{u=1}^{n} (b_{uu}^{(\ell)})^2
\]
\[
= \|(G^{(\ell - 1)})^* B^{(\ell - 1)} G^{(\ell - 1)}\|_F^2 - \left( \sum_{u=1}^{n} (b_{uu}^{(\ell - 1)})^2 - (b_{pp}^{(\ell - 1)})^2 - (b_{qq}^{(\ell - 1)})^2 + (b_{pp}^{(\ell)})^2 + (b_{qq}^{(\ell)})^2 \right)
\]
\[
= \|B^{(\ell - 1)}\|_F^2 - \sum_{u=1}^{n} (b_{uu}^{(\ell - 1)})^2 + [(b_{pp}^{(\ell - 1)})^2 + (b_{qq}^{(\ell - 1)})^2 - (b_{pp}^{(\ell)})^2 - (b_{qq}^{(\ell)})^2]
\]
\[
= \text{off}(B^{(\ell - 1)})^2 - 2|b_{pq}^{(\ell - 1)}|^2.
\]
We see that \( \|\Gamma_{B^{(\ell)}}\|_F^2 = 4 \|B^{(\ell)}\|_F^2 \). Hence, \( \text{off}(\Gamma_{B^{(\ell)}})^2 = 4 \text{off}(B^{(\ell)})^2 \).
We have the following result on the linear convergence of Algorithm 3.6. The proof follows from [23] Theorem 3.3 and thus we omit it here.

**Theorem 3.7** Let \( \{\sigma_{w_{u=1}}\} \) be the \( n \) singular values of \( A \) and \( \Gamma_{A^{(\ell)}} \) be the matrix after \( \ell \) orthogonal JRS-symeplectic Jacobi updates generated by Algorithm 3.6. Then there exists a permutation \( \{\varphi_1, \varphi_2, \ldots, \varphi_n\} \) of \( \{1, 2, \ldots, n\} \) such that
\[
\lim_{\ell \to \infty} \Gamma_{A^{(\ell)}}^T \Gamma_{A^{(\ell)}} = \text{diag}(\sigma_{\varphi_1}^2, \ldots, \sigma_{\varphi_n}^2, \sigma_{\varphi_1}^2, \ldots, \sigma_{\varphi_n}^2, \sigma_{\varphi_1}^2, \ldots, \sigma_{\varphi_n}^2).
\]
Moreover,
\[
\text{off}(\Gamma_{A^{(\ell)}}^T \Gamma_{A^{(\ell)}}) \leq \left(1 - \frac{1}{N}\right)^\ell \text{off}(\Gamma_{A^{(0)}}^T \Gamma_{A^{(0)}}), \quad N := \frac{1}{2} n(n - 1).
\]

However, in Algorithm 3.6, the search for the optimal columns \( p \) and \( q \) needs \( O(n^2) \). To reduce the cost, one may adopt the scheme of cyclic-by-column as the cyclic Jacobi algorithm [8 Algorithm 8.4.3]. In the following procedure, we provide a structure-preserving one-sided cyclic Jacobi algorithm for orthogonalizing the columns of an \( m \times n \) quaternion matrix \( A \).

**Algorithm 3.8**

Given an \( m \times n \) quaternion matrix \( A = A_0 + A_1 i + A_2 j + A_3 k \in \mathbb{H}^{m \times n} \) and a tolerance \( \text{tol} > 0 \), this algorithm overlaps the real counterpart \( \Gamma_A \) by \( \Gamma_A \tilde{V} \), where \( \tilde{V} \) is orthogonal and \( \text{off}(V^T \Gamma_A^T \Gamma_A \tilde{V}) \leq \text{tol} \cdot \| \Gamma_A^T \Gamma_A \|_F \).

\[
\tilde{V} = \Gamma_{I_n}, \quad \zeta = \text{tol} \cdot \| \Gamma_A^T \Gamma_A \|_F
\]

while \( \text{off}(\Gamma_A^T \Gamma_A) > \zeta \)

for \( p = 1 : n - 1 \)

for \( q = p + 1 : n \)

\[
a_{p0} = A_0(:, p), \quad a_{p1} = A_1(:, p), \quad a_{p2} = A_2(:, p), \quad a_{p3} = A_3(:, p)
\]

\[
a_{q0} = A_0(:, q), \quad a_{q1} = A_1(:, q), \quad a_{q2} = A_2(:, q), \quad a_{q3} = A_3(:, q)
\]

\[
(c_r, s_0, s_1, s_2, s_3) = \text{GJSJR}(a_{p0}, a_{p1}, a_{p2}, a_{p3}, a_{q0}, a_{q1}, a_{q2}, a_{q3})
\]

\[
\Gamma_A = \Gamma_A \Gamma_{G(p,q,\theta)}
\]

\[
\tilde{V} = \tilde{V} \Gamma_{G(p,q,\theta)}
\]

end

end

Remark 3.9 In Algorithm 3.8, we only need to store the first \( m \) rows of \( \Gamma_A \), which reduce the total storage. The later numerical tests show that Algorithm 3.8 works much better than the implicit Jacobi algorithm in [18].

We now give the quadratic convergence analysis of Algorithm 3.8. Algorithm 3.8 gives the following iterative scheme:

\[
\Gamma_{A^{(\ell + 1)}} = \Gamma_{A^{(\ell)}} \Gamma_{G^{(\ell)}}, \quad \ell = 0, 1, 2, \ldots,
\]

where \( A^{(0)} = A \) and \( G^{(\ell)} \in \mathbb{H}^{n \times n} \) is a unitary quaternion matrix defined in (3.1) with the cosine-sine group \( (c_r, s_0, s_1, s_2, s_3) \) being generated by Algorithm 3.5. In fact, Algorithm 3.8 can be seen as a structure-preserving cyclic Jacobi algorithm for \( \Gamma_A^T \Gamma_A \).

We have the following theorem on the quadratic convergence of Algorithm 3.8. The proof can be seen as a generalization of [28 30].

**Theorem 3.10**

Let \( \{\sigma_w\}_{w=1}^n \) be the \( n \) singular values of \( A \) and \( \Gamma_{A^{(\ell)}} \) be the matrix after \( \ell \) orthogonal JRS-symplectic Jacobi updates generated by Algorithm 3.8. If \( \text{off}(\Gamma_{A^{(d)}}^T \Gamma_{A^{(d)}}) < \delta/2 \) for some \( d \geq 1 \) where \( 0 < 2\delta \leq \min_{w \neq w'} |\sigma_w^2 - \sigma_{w'}^2| \), then

\[
\text{off}(\Gamma_{A^{(d+N)}}^T \Gamma_{A^{(d+N)}}) \leq \sqrt{\frac{25}{72}} \cdot \frac{\text{off}(\Gamma_{A^{(d)}}^T \Gamma_{A^{(d)}})^2}{\delta}.
\]
Proof. Write $S^{(d)} := \Gamma_{A^{(d)}}^T \Gamma_{A^{(d)}} = D^{(d)} + E^{(d)} + (E^{(d)})^T$, where $D^{(d)}$ and $E^{(d)}$ are diagonal and strictly upper triangular, respectively. By using Theorem \[3.7\] and the Wielandt-Hoffman theorem (\[8\] Theorem 8.1.4) we have

$$|s^{(d)}_{ww} - \sigma_{\varphi_w}^2| \leq \|D^{(d)} - S^{(d)}\|_F < \frac{\delta}{2}, \quad 1 \leq w \leq n. \quad (3.11)$$

Thus we have for two distinct eigenvalues $\sigma_{\varphi_u}^2$ and $\sigma_{\varphi_v}^2$,

$$|s^{(d)}_{uu} - s^{(d)}_{vv}| = |(s^{(d)}_{uu} - \sigma_{\varphi_u}^2) - (s^{(d)}_{vv} - \sigma_{\varphi_v}^2) + (\sigma_{\varphi_u}^2 - \sigma_{\varphi_v}^2)|
\geq |\sigma_{\varphi_u}^2 - \sigma_{\varphi_v}^2| - |s^{(d)}_{uu} - \sigma_{\varphi_u}^2| - |s^{(d)}_{vv} - \sigma_{\varphi_v}^2|
> 2\delta - \frac{\delta}{2} - \frac{\delta}{2} = \delta. \quad (3.12)$$

Since $\text{off}(\Gamma_{A^{(d)}}^T \Gamma_{A^{(d)}})$ is decreasing, we know that $\text{off}(\Gamma_{A^{(d)}}^T \Gamma_{A^{(d)}}) < \delta/2$ and (3.12) hold for $\ell > d$.

We show the quadratic convergence of Algorithm 3.8. We first consider the case of one multiple singular value. Assume without loss of generality that only $\sigma_{\varphi_1}$ is a multiple singular value of $A^{(\ell)}$ ($\ell > d$) and the diagonal entries $s_{11}^{(\ell)}, s_{22}^{(\ell)}, \ldots, s_{n1n1}^{(\ell)}$ of $S^{(\ell)} := \Gamma_{A^{(\ell)}}^T \Gamma_{A^{(\ell)}}$ converge to $\sigma_{\varphi_1}^2$. Then, by appropriate row and column interchanges, we get a permutation matrix $P$ such that

$$\tilde{S}^{(\ell)} = P^T S^{(\ell)} P = \begin{bmatrix} \tilde{S}_{11}^{(\ell)} & \tilde{S}_{12}^{(\ell)} \\ \tilde{S}_{21}^{(\ell)} & \tilde{S}_{22}^{(\ell)} \end{bmatrix},$$

where $\tilde{S}_{11}^{(\ell)} \in \mathbb{R}^{4n_1 \times 4n_1}$ with diagonal entries converging to $\sigma_{\varphi_1}^2$.

We provide an upper bound for the quantity

$$\Phi_1^{(\ell)} := \sqrt{\sum_{1 \leq p \neq q \leq 4n_1} (s_{pq}^{(\ell)})^2}.$$

As in \[31\], it is easy to see that

$$T^{(\ell)} = \begin{bmatrix} I_{4n_1} & -\tilde{S}_{12}^{(\ell)} (\tilde{S}_{22}^{(\ell)} - \sigma_{\varphi_1}^2 I_{4n-4n_1})^{-1} \\ 0 & I_{4n-4n_1} \end{bmatrix} \begin{bmatrix} \tilde{S}_{11}^{(\ell)} - \sigma_{\varphi_1}^2 I_{4n_1} \\ \tilde{S}_{21}^{(\ell)} \end{bmatrix} = \begin{bmatrix} \tilde{S}_{11}^{(\ell)} - \sigma_{\varphi_1}^2 I_{4n_1} \\ \tilde{S}_{21}^{(\ell)} \end{bmatrix} \begin{bmatrix} \tilde{S}_{12}^{(\ell)} \\ \tilde{S}_{22}^{(\ell)} - \sigma_{\varphi_1}^2 I_{4n-4n_1} \end{bmatrix}$$
and the rank of $T^{(\ell)}$ is the same as $\tilde{S}^{(\ell)} - \sigma_{\varphi_1}^2 I_{4n}$, which implies that

$$\tilde{S}_{11}^{(\ell)} - \sigma_{\varphi_1}^2 I_{4n_1} = \tilde{S}_{12}^{(\ell)} (\tilde{S}_{22}^{(\ell)} - \sigma_{\varphi_1}^2 I_{4n-4n_1})^{-1} \tilde{S}_{21}^{(\ell)}.$$

Let $\tilde{\sigma}_{\varphi_w}^2$ be the eigenvalues of $\tilde{S}_{22}^{(\ell)}$. Note that

$$|\sigma_{\varphi_w}^2 - \tilde{\sigma}_{\varphi_w}^2| \leq \|\tilde{S}^{(\ell)} - \begin{bmatrix} \Sigma_{11}^{(\ell)} & 0 \\ 0 & \Sigma_{22}^{(\ell)} \end{bmatrix}\|_F \leq \text{off}(\tilde{S}^{(\ell)}) = \text{off}(S^{(\ell)}) \leq \frac{\delta}{2},$$
where $\Sigma^{(ℓ)}_{11} = \text{diag}(\hat{s}_{11}, \hat{s}_{22}, \ldots, \hat{s}_{4n_1,4n_1})$ with $\hat{s}_{ww}$ being the $(w,w)$ entry of $\hat{S}^{(ℓ)}_{11}$. Thus,

$$|\sigma^2_{\varphi_1} - \bar{\sigma}^2_{\varphi_1} w | \geq |\sigma^2_{\varphi_1} - \sigma^2_{\varphi_1} w | - |\sigma^2_{\varphi_1} - \bar{\sigma}^2_{\varphi_1} w | \geq 2\delta - \frac{\delta}{2} = \frac{3\delta}{2}.$$ 

This, together with (3.13), yields

\[
(\Phi^{(ℓ)}_{1})^2 = \text{off}(\hat{S}^{(ℓ)}_{11})^2 \leq \|\hat{S}^{(ℓ)}_{11} - \sigma^2_{\varphi_1} I_{4n_1}\|_F^2 \leq \frac{\|\hat{S}^{(ℓ)}_{12}\|_F^2^2}{\min\sigma^2_{\varphi_1} - \sigma^2_{\varphi_1} w |} \leq \frac{2\text{off}(S^{(ℓ)})^2}{9\delta^2} \|\hat{S}^{(ℓ)}_{12}\|_F^2 \leq \frac{2\text{off}(S^{(d)})^2}{9\delta^2} \|\hat{S}^{(ℓ)}_{12}\|_F^2 \tag{3.14a}
\]

\[
\leq \frac{\text{off}(S^{(d)})^4}{9\delta^2} \tag{3.14b}
\]

since $\|\hat{S}^{(ℓ)}_{12}\|_F^2 \leq 1/2 \text{off}(\hat{S}^{(ℓ)})^2 = 1/2 \text{off}(S^{(ℓ)})^2 \leq 1/2\text{off}(S^{(d)})^2$. The estimates in (3.14) is crucial for proving the quadratic convergence of the structure-preserving one-sided cyclic Jacobi algorithm.

As in (3.4), the rotation angle $\theta_\ell$ is chosen such that $|\theta_\ell| \leq \pi/4$. Using (3.10), (3.12) and $S^{(ℓ)} = \Gamma_{B^{(ℓ)}}$ we have

\[
|\sin \theta_\ell| \leq \frac{1}{2} \tan 2\theta_\ell = \frac{|b^{(\ell-1)}_{pq}|}{|b^{(\ell-1)}_{pq} - b^{(\ell-1)}_{pp}|} \leq \frac{|b^{(\ell-1)}_{pq}|}{\delta} \tag{3.15}
\]

This, together with $\frac{1}{2}\text{off}(B^{(\ell-1)})^2 - \frac{1}{2}\text{off}(B^{(\ell)})^2 = |b^{(\ell-1)}_{pq}|^2$, yields

\[
\text{off}(B^{(d)})^2 - 2 \sum_{\ell=d+1}^{d+N} |b^{(\ell-1)}_{pq}|^2 = \text{off}(B^{(d+N)})^2 \geq 0.
\]

Using (3.15) we have

\[
\sum \sin^2 \theta_\ell \leq \sum \frac{|b^{(\ell-1)}_{pq}|^2}{\delta^2} \leq \frac{\text{off}(B^{(d)})^2}{2\delta^2}. \tag{3.16}
\]

where $\sum$ means that we include in the sum only rotations of entries outside the first $n_1$ rows and the first $n_1$ columns of $B^{(ℓ)}$.

We now show the quadratic convergence of Algorithm 3.8 for the case of one multiple singular value. As in (30), for example, we take an $m \times 5$ quaternion matrix $A$. In this case, $B^{(0)} = A^*A \in \mathbb{H}^{5 \times 5}$. In the following, we show the effect of annihilating the entries in the first row and column of $B^{(d)}$. Since we are only interested in the off diagonal entries, which is updated when these entries are affected by the current rotations, the diagonal entries are all denoted by “$\times$.”
For the entries of the first row of \( B^{(d)} \), we have the following inequalities

\[
\begin{align*}
|b_{14}^{(d+4)}| &\leq |b_{54}^{(d+3)}||\sin \theta_{d+3}|, \\
|b_{13}^{(d+4)}| &\leq |b_{43}^{(d+2)}||\sin \theta_{d+2}| + |b_{53}^{(d+2)}||\sin \theta_{d+3}|, \\
|b_{12}^{(d+4)}| &\leq |b_{32}^{(d+1)}||\sin \theta_{d+1}| + |b_{42}^{(d+1)}||\sin \theta_{d+2}| + |b_{52}^{(d+1)}||\sin \theta_{d+3}|.
\end{align*}
\]  

Thus,

\[
\begin{align*}
|b_{12}^{(d+4)}|^2 + |b_{13}^{(d+4)}|^2 + |b_{14}^{(d+4)}|^2 &\leq (|b_{32}^{(d+1)}|^2 + |b_{42}^{(d+1)}|^2 + |b_{52}^{(d+1)}|^2)(\sin^2 \theta_{d+1} + \sin^2 \theta_{d+2} + \sin^2 \theta_{d+3}) \\
&\quad + (|b_{43}^{(d+2)}|^2 + |b_{53}^{(d+2)}|^2)(\sin^2 \theta_{d+2} + \sin^2 \theta_{d+3}) + |b_{54}^{(d+3)}|^2 \sin^2 \theta_{d+3} \\
&\leq (|b_{32}^{(d+1)}|^2 + |b_{42}^{(d+1)}|^2 + |b_{52}^{(d+1)}|^2 + |b_{43}^{(d+2)}|^2 + |b_{53}^{(d+2)}|^2 + |b_{54}^{(d+3)}|^2) \\
&\quad \times (\sin^2 \theta_{d+1} + \sin^2 \theta_{d+2} + \sin^2 \theta_{d+3}).
\end{align*}
\]

Since each rotation affects only two entries in each of the related columns or rows while the sum
of the squares of their absolute values is kept unchanged we have
\[
|b_{15}^{(d+3)}|^2 + |b_{25}^{(d+1)}|^2 + |b_{35}^{(d+2)}|^2 + |b_{45}^{(d+3)}|^2 = |b_{15}^{(d)}|^2 + |b_{25}^{(d)}|^2 + |b_{35}^{(d)}|^2 + |b_{45}^{(d)}|^2,
\]
\[
|b_{14}^{(d+2)}|^2 + |b_{24}^{(d+1)}|^2 + |b_{34}^{(d+2)}|^2 = |b_{14}^{(d)}|^2 + |b_{24}^{(d)}|^2 + |b_{34}^{(d)}|^2,
\]
\[
|b_{13}^{(d+1)}|^2 + |b_{23}^{(d+1)}|^2 = |b_{13}^{(d)}|^2 + |b_{23}^{(d)}|^2.
\]
Thus
\[
|b_{12}^{(d+4)}|^2 + |b_{13}^{(d+4)}|^2 + |b_{14}^{(d+4)}|^2
\leq
(|b_{13}^{(d)}|^2 + |b_{23}^{(d)}|^2 + |b_{14}^{(d)}|^2 + |b_{24}^{(d)}|^2 + |b_{34}^{(d)}|^2 + |b_{15}^{(d)}|^2 + |b_{25}^{(d)}|^2 + |b_{35}^{(d)}|^2 + |b_{45}^{(d)}|^2)
\times
(sin^2 \theta_{d+1} + sin^2 \theta_{d+2} + sin^2 \theta_{d+3})
\leq
\frac{1}{2} off(B^{(d)})^2 (sin^2 \theta_{d+1} + sin^2 \theta_{d+2} + sin^2 \theta_{d+3}).
\]
Moreover, the sum of the squares of the absolute values of these entries in the first row remains unchanged in the subsequent rotations.

Similarly, we have after successively annihilating the entries in the second row
\[
|b_{23}^{(d+7)}|^2 + |b_{24}^{(d+7)}|^2 \leq \frac{1}{2} off(B^{(d+4)})^2 (sin^2 \theta_{d+5} + sin^2 \theta_{d+6})
\leq \frac{1}{2} off(B^{(d)})^2 (sin^2 \theta_{d+5} + sin^2 \theta_{d+6}).
\]
Furthermore, for the third row we have
\[
|b_{34}^{(d+9)}|^2 \leq \frac{1}{2} off(B^{(d+7)})^2 sin^2 \theta_{d+8} \leq \frac{1}{2} off(B^{(d)})^2 sin^2 \theta_{d+8}.
\]
Finally, the fourth row above the diagonal is annihilated. From (3.19), (3.20) and (3.21) we obtain
\[
off(B^{(d+10)})^2 \leq off(B^{(d)})^2 \sum_{t=0}^{9} sin^2 \theta_{d+t} \leq off(B^{(d)})^2 \frac{off(B^{(d)})^2}{2\delta^2} = \frac{off(B^{(d)})^4}{2\delta^2},
\]
where the second inequality follows from (3.16).

Analogously to the proof of (3.22), using the equality off(S^{(d)})^2 = 4off(B^{(d)})^2, (3.14b) and (3.16) we have
\[
off(S^{(d+N)})^2 = 4off(B^{(d+N)})^2
= 4 \sum_{1 \leq p < q \leq n_1} |b_{pq}^{(d+N)}|^2 + 8 \sum_{p < q > n_1} |b_{pq}^{(d+N)}|^2
\leq off(\tilde{S}^{(d+N)})^2 + 4off(B^{(d)})^2 \left( \sum_{\ell} sin^2 \theta_{\ell} \right)
\leq off(\tilde{S}^{(d+N)})^2 + 4off(B^{(d)})^2 \frac{off(B^{(d)})^2}{2\delta^2}
\leq \frac{off(S^{(d)})^4}{9\delta^2} + \frac{off(S^{(d)})^4}{8\delta^2}.
\]
This shows that Algorithm 3.8 converges quadratically when there is only one multiple singular value.

Next, we show the quadratic convergence of Algorithm 3.8 for the case of more than one multiple singular values. If there exist \( l \) multiple singular values, then we have

\[
\text{off}(S^{d+N})^2 \leq \frac{9 + 8l}{72\delta^2}\text{off}(S^{d})^4. \tag{3.23}
\]

In the following we decrease the factor \( \frac{9 + 8l}{72\delta^2} \). Assume that \( \sigma_{\varphi_w} \) is a multiple singular value of \( A^{(\ell)} \) with multiplicity \( n_w \) for \( w = 1, \ldots, l \) and \( s^{(\ell)}_{11}, \ldots, s^{(\ell)}_{n_1n_1}, s^{(\ell)}_{n_1+1,n_1+1}, \ldots, s^{(\ell)}_{n_1+n_2,n_1+n_2}, \ldots, s^{(\ell)}_{n_1+\cdots+n_{l-1}+1,n_1+\cdots+n_{l-1}+1}, \ldots, s^{(\ell)}_{n_1+\cdots+n_{l-1}+n_l,n_1+\cdots+n_{l-1}+n_l} \) converge to \( \sigma_{\varphi_1}^2, \sigma_{\varphi_2}^2, \ldots, \sigma_{\varphi_l}^2 \) accordingly.

Then, by appropriate row and column interchanges, we get a permutation matrix \( P \) such that

\[
\tilde{S}^{(\ell)} = P^T S^{(\ell)} P = \begin{bmatrix}
\tilde{S}^{(\ell)}_{11} & \tilde{S}^{(\ell)}_{12} & \cdots & \tilde{S}^{(\ell)}_{1l} & \tilde{S}^{(\ell)}_{1,l+1} \\
\tilde{S}^{(\ell)}_{21} & \tilde{S}^{(\ell)}_{22} & \cdots & \tilde{S}^{(\ell)}_{2l} & \tilde{S}^{(\ell)}_{2,l+1} \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
\tilde{S}^{(\ell)}_{l1} & \tilde{S}^{(\ell)}_{l2} & \cdots & \tilde{S}^{(\ell)}_{ll} & \tilde{S}^{(\ell)}_{l,l+1} \\
\tilde{S}^{(\ell)}_{l+1,1} & \tilde{S}^{(\ell)}_{l+1,2} & \cdots & \tilde{S}^{(\ell)}_{l+1,l} & \tilde{S}^{(\ell)}_{l+1,l+1}
\end{bmatrix},
\]

where \( \tilde{S}^{(\ell)}_{ww} \in \mathbb{R}^{4n_w \times 4n_w} \) with diagonal entries converging to \( \sigma_{\varphi_w}^2 \) for \( w = 1, \ldots, l \).

Define the quantities

\[
\Phi^{(\ell)}_{ww} := \sqrt{\sum_{u=0}^{w-1} \sum_{4u+1 \leq p \neq q \leq 4u \times 4w} \left( \frac{s^{(\ell)}_{p,q}}{s^{(\ell)}_{w,u}} \right)^2}, \quad w = 1, \ldots, l,
\]

where \( n_0 = 0 \).

Analogous to the proof of (3.14a) we have

\[
\left( \Phi^{(\ell)}_{ww} \right)^2 \leq \frac{2\text{off}(S^{(\ell)})^2}{9\delta^2} \sum_{u=1, u \neq w}^{l+1} \| \tilde{S}^{(\ell)}_{wu} \|_F^2.
\]

We note that

\[
\sum_{w=1}^{l} \left( \sum_{u=1, u \neq w}^{l+1} \| \tilde{S}^{(\ell)}_{wu} \|_F^2 \right) \leq \sum_{1 \leq p \neq q \leq 4n} (s^{(\ell)}_{pq})^2 = \text{off}(\tilde{S}^{(\ell)})^2 = \text{off}(S^{(\ell)})^2.
\]

Hence,

\[
\sum_{w=1}^{l} \left( \Phi^{(\ell)}_{ww} \right)^2 \leq \frac{2\text{off}(S^{(\ell)})^2}{9\delta^2} \sum_{u=1, u \neq w}^{l+1} \left( \sum_{u=1, u \neq w}^{l+1} \| \tilde{S}^{(\ell)}_{wu} \|_F^2 \right) \leq \frac{2\text{off}(S^{(\ell)})^4}{9\delta^2} \leq \frac{2\text{off}(S^{(d)})^4}{9\delta^2}.
\]
Therefore, (3.23) is reduced to

\[
\text{off}(S(d+N))^2 \leq \sum_{t=1}^{l} (\Phi_{t}^{(d+N)})^2 + \text{off}(B^{(d)})^2 \left( \sum \sin^2 \theta_{t} \right) \\
\leq \frac{2\text{off}(S^{(d)})^4}{9\delta^2} + \frac{\text{off}(S^{(d)})^4}{8\delta^2} \\
\leq \frac{25}{72} \cdot \text{off}(S^{(d)})^4.
\]  

That is,

\[
\text{off}(\Gamma_{A(d+N)}^T \Gamma_{A(d+N)}) \leq \sqrt{\frac{25}{72}} \cdot \text{off}(\Gamma_{A(d)}^T \Gamma_{A(d)})^2 \frac{\delta}{\delta^2},
\]

which shows that Algorithm 3.8 is quadratically convergent when there exist more than one multiple singular values.

\[\square\]

Finally, we point out that, for an \( m \times n \) quaternion matrix \( A = A_0 + A_1i + A_2j + A_3k \in \mathbb{H}^{m \times n} \), Algorithm 3.8 generates a matrix \( \Gamma_{A(\Theta)} \) after \( \Theta \) orthogonal JRS-symplectic Jacobi updates such that \( A(\Theta) = AV(\Theta) \) has sufficiently orthogonal columns (which is measured by \( \text{off}(\Gamma_{A(\Theta)}^T \Gamma_{A(\Theta)}) \leq \text{tol} \cdot \|\Gamma_{A} \|_F \) for a prescribed tolerance \( \text{tol} > 0 \)), where \( V(\Theta) = G(0)G(1) \ldots G(\Theta) \in \mathbb{H}^{n \times n} \) is a unitary matrix. Then the SVD of \( A \) follows from column scaling of \( A(\Theta) = AV(\Theta) \), i.e.,

\[
A(\Theta) = AV(\Theta) = U(\Theta)\Sigma(\Theta),
\]

where \( \Sigma(\Theta) = \text{diag}(\sigma_{\varphi_1}(\Theta), \sigma_{\varphi_2}(\Theta), \ldots, \sigma_{\varphi_n}(\Theta)) \) with \( \sigma_{\varphi_w}(\Theta) \geq 0 \) for \( w = 1, \ldots, n \) and \( U(\Theta) \in \mathbb{H}^{m \times n} \) is such that \( (U(\Theta))^*U(\Theta) = I_n \).

4 Numerical Experiments

In this section, we present some numerical experiments to illustrate the effectiveness of Algorithm 3.8 for computing the SVD of a rectangle quaternion matrix and compare it with the quaternion toolbox for MATLAB [26] and the implicit Jacobi algorithm in [18]. We also apply the proposed algorithm to color image compression. All the numerical tests were carried out in MATLAB R2016a running on a laptop of 2.00 GHz CPU and 4GB of RAM.

Example 4.1 In this example we compute the SVD of 19 \( m \times n \) random quaternion matrices, where \( m \) ranges from 10 to 100 with increment of 5 and \( n = m/5 \).

Figures 1 and 2 show, respectively, the CPU time and the residual \( \|AV(\Theta) - U(\Theta)\Sigma(\Theta)\|_F \) at the final iterate of the corresponding algorithms for different quaternion matrix sizes.

We can see from Figure 1 that Algorithm 3.8 is more efficient than the implicit Jacobi algorithm in [18] as the matrix size becomes larger. We also observe from Figure 2 that both two algorithms obtain almost the same calculation accuracy.
Figure 1: Numerical results for Ex. 4.1

To further illustrate the effectiveness of Algorithm 3.8, Figure 3 depicts the computed singular values of a $100 \times 20$ quaternion matrix by using Algorithm 3.8 and the \texttt{svd} function in the quaternion toolbox for \texttt{MATLAB} [26]. We see from Figure 3 that both algorithms obtain almost the same singular values.

Figure 3: Computed singular values for one of the test quaternion matrices.
Example 4.2 In this example, we apply Algorithm 3.8 to image compression. We know a color image can be represented by a pure quaternion matrix \( A = (a_{ij})_{m \times n} = R_i + G_j + B_k \), where \( R, G, B \) represent the red, green, blue parts of the color image. We use Algorithm 3.8 to get the SVD of the color image without separating the color image into three channel images.

For demonstration purpose, in Example 4.2, we take the color images Snowberg, Rabbit and Eiffel Tower (Eiffel), whose sizes are 50 \( \times \) 50, 50 \( \times \) 50 and 50 \( \times \) 100 accordingly. Figure 4 shows the singular values of the original three color images. We can see that the singular values of these images decay very fast. One may use Algorithm 3.8 to compute the SVD of a color image \( A \) such that \( AV^{(\Theta)} = U^{(\Theta)}\Sigma^{(\Theta)} \), where \( U^{(\Theta)}, \Sigma^{(\Theta)}, \) and \( V^{(\Theta)} \) are given by (3.25). Then we can compress an image by a lower-rank matrix approximation:

\[
A_S = \sum_{w=1}^{S} \sigma_w^{(\Theta)} u_w^{(\Theta)} (v_w^{(\Theta)})^*, \tag{4.1}
\]

where \( \{\sigma_w^{(\Theta)}\}_{w=1}^S \) are the \( S \) largest singular values of \( A \), \( u_w^{(\Theta)} \) and \( v_w^{(\Theta)} \) are the left and right singular vectors of \( A \) corresponding to \( \sigma_w^{(\Theta)} \) for \( w = 1, \ldots, S \).

Figure 4: Singular values of three original color images.

Figure 5 displays the original image and four estimated images with \( S = 10, 20, 30 \) and 40. We observe from Figure 5 that small \( S \) already provides a good estimation of the original color image. Meanwhile, the storage requirements drop from \( 3mn \) to \( S(4m + 4n + 1) \). The peak signal-to-noise ratios (PSNRs) of the four estimated images are also listed in Table 1. The
PSNR between the original image \( f \) and a test image \( g \), both of size \( m \times n \), is defined by

\[
PSNR(f, g) = 10 \log_{10} \left( \frac{255^2}{MSE(f, g)} \right),
\]

where MSE means the mean squared error defined by

\[
MSE(f, g) = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} (f_{ij} - g_{ij})^2.
\]

From Table 1, we see that a lower-rank matrix approximation may provide a good image compression, where the needed singular values and associated left and right singular vectors can be obtained by our algorithm.

![Figure 5: Original and estimated images.](image)

| S     | 10     | 20     | 30     | 40     |
|-------|--------|--------|--------|--------|
| Snowberg | 34.7060 | 35.9495 | 36.1030 | 36.1010 |
| Rabbit  | 30.1050 | 33.8167 | 35.4383 | 35.6440 |
| Eiffel  | 24.6439 | 25.4894 | 25.7858 | 25.8698 |

Table 1: PSNR of the estimated color images.
5 Conclusions

In this paper, we have proposed a real structure-preserving one-sided cyclic Jacobi algorithm for computing the QSVD. This algorithm involves a sequence of column orthogonalizations in pairs via a sequence of orthogonal JRS-symplectic Jacobi rotations to the real counterpart of a quaternion matrix. The quadratic convergence is established under some assumptions. Finally, we report some numerical experiments to illustrate the efficiency of our algorithm. We also point out that the proposed algorithm can be used for many practical applications such as the pseudoinverse of a quaternion matrix and color image processing (e.g., image compression, image enhancement, and image denoising).

References

[1] A. I. Arbab, The quaternionic quantum mechanics, Applied Physics Research 3 (2011) 160–170.

[2] A. J. Davies and B. H. J. Mckellar, Observability of quaternionic quantum-mechanics, Phys. Rev. A 46 (1992) 3671–3675.

[3] E. Doukhnitch and E. Ozen, Hardware-oriented algorithm for quaternion-valued matrix decomposition, IEEE Transactions on Circuits and Systems–II: Express Briefs 58 (2011) 225–229.

[4] S. G. Eltaweel, Colour image compression, 2011 Seventh International Computer Engineering Conference (ICENCO), Giza, IEEE, December 2011.

[5] S. Enshaeifar, S. Kouchaki, C. C. Took, and S. Sanei, Quaternion singular spectrum analysis of electroencephalogram with application in sleep analysis, IEEE Transactions on Neural Systems and Rehabilitation Engineering 24 (2016) 57–67.

[6] D. Finkelstein, J. M. Jauch, and D. Speiser, Notes on quaternion quantum mechanics (CERN, Report 59-7), Logico-Algebraic Approach to Quantum Mechanics II: (1979) 367–421.

[7] S. Gai, G. W. Yang, M. H. Wan, and L. Wang, Denoising color images by reduced quaternion matrix singular value decomposition, Multidim. Syst. Sign. Process. 26 (2015) 307–320.

[8] G. H. Golub and C. F. Van Loan, Matrix Computations, 3rd ed., The Johns Hopkins University Press, Baltimore, 1996.

[9] W. R. Hamilton, On quaternions, Proceeding of the Royal Irish Academy, November 11, 1844.

[10] W. R. Hamilton. Elements of quaternions. Longmans, Green and Co., London, 1866.

[11] Z. G. Jia , M. S. Wei, S. T. Ling, A new structure-preserving method for quaternion Hermitian eigenvalue problems, J. Comput. Appl. Math. 239 (2013) 12–24.
[12] T. Jiang, *Algebraic methods for diagonalization of a quaternion matrix in quaternionic quantum theory*, J. Math. Phys. 46 (2005) 052106–052108.

[13] I. L. Kantor, A. S. Solodovnikov, *Hypercomplex Numbers, An Elementary Introduction to Algebras*, Springer, Berlin, 1989.

[14] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*, Prentice-Hall Series in Automatic Computation, Englewood Cliffs: Prentice-Hall, 1974.

[15] N. Le Bihan, *Traitement algébrique des signaux vectoriels. Application en séparation d’ondes sismiques*, PhD Thesis, INPG, 2001.

[16] N. Le Bihan and J. Mars, *Singular value decomposition of matrices of quaternions: A new tool for vector-sensor signal processing*, Signal Process. 84 (2004) 1177–1199.

[17] N. Le Bihan and S. J. Sangwine, *Quaternion principal component analysis of color images*, in: Proc. Int. Conf. Image Process. (ICIP 2003), vol. 1, pp. I-809-1–I-809-12 (2003).

[18] N. Le Bihan and S. J. Sangwine, *Jacobi method for quaternion matrix singular value decomposition*, Appl. Math. Comp. 187 (2007) 1265–1271.

[19] S. D. Leo, P. Rotelli, *Quaternion scalar field*, Phys. Rev. D 45 (1992) 575–579.

[20] Y. Li, M. S. Wei, F. X. Zhang, J. L. Zhao, *A fast structure-preserving method for computing the singular value decomposition of quaternion matrices*, Appl. Math. Comp. 235 (2014) 157–167.

[21] Y. Li, M. S. Wei, F. X. Zhang, J. L. Zhao, *Real structure-preserving algorithms of Householder based transformations for quaternion matrices*, J. Comput. Appl. Math. 305 (2016) 82–91.

[22] F. Liu, L. H. Ma, C. Liu, and Z. M. Lu, *Optimal blind watermarking for color images based on the U matrix of quaternion singular value decomposition*, Multimed Tools Appl. 77 (2018) 23483–23500.

[23] R.-R. Ma, Z. G. Jia and Z.-J. Bai, *A Structure-preserving Jacobi algorithm for quaternion hermitian eigenvalue problems*, Comput. Math. Appl. 75 (2018) 809–820.

[24] S. C. Pei, J. H. Cheng, and J. J. Ding, *Quaternion matrix singular value decomposition and its applications for color image processing*, In: Proc. 2003 Int. Conf. Image Processing (ICIP 2003), vol. 1, pp. 805–808 (2003).

[25] K. N. Plataniotis, A.N. Venetsanopoulos, (2000) *Color Image Compression. In: Color Image Processing and Applications. Digital Signal Processing*. Springer, Berlin, Heidelberg.

[26] S. J. Sangwine and N. L. Bihan, *Quaternion toolbox for Matlab*, [online], Software library. Available: [http://qtfm.sourceforge.net/](http://qtfm.sourceforge.net/)
[27] S. J. Sangwine and N. Le Bihan, Quaternion singular value decomposition based on bidiagonalization to a real or complex matrix using quaternion Householder transformations, Appl. Math. Comput. 182 (2006) 727–738.

[28] H. P. M. van Kempen, On the quadratic convergence of the special cyclic Jacobi method, Numer. Math. 9 (1966) 19–22.

[29] M. H. Wang, M. S. Wei, Y. Feng, An iterative algorithm for least squares problem in quaternionic quantum theory, Compu. Phys. Commun. 179 (2008) 203–207.

[30] J. H. Wilkinson, Note on the quadratic convergence of the cyclic Jacobi process, Numer. Math. 4 (1962) 296–300.

[31] J. H. Wilkinson, The QR algorithm for real symmetric matrices with multiple eigenvalues, The Computer J. 8, (1965) 85–87.

[32] Y. Xu, L. Yu, H. Xu, and T. Nguyen, Vector sparse representation of color image using quaternion matrix analysis, IEEE Transactions on Image Processing 24 (2015) 1315–1329.

[33] C. C. Yi, Y. Lv, Z. Dang, H. Xiao, and X. Yu, Quaternion singular spectrum analysis using convex optimization and its application to fault diagnosis of rolling bearing, Measurement 103 (2017) 321–332.

[34] B. B. Zhou and R. P. Brent, On the parallel implementations of the one-sided Jacobi algorithm for singular value decompositions. Proc. of 3rd Euromicro Workshop on Parallel and Distributed Processing. San Remo, Italy, Jan. (1995)401–408.

[35] F. Z. Zhang, Quaternions and matrices of quaternions, Linear Algebra Appl. 251 (1997) 21–57.