An Iterative algorithm for $\eta$-(anti)-Hermitian least-squares solutions of quaternion matrix equations

Fatemeh Panjeh Ali Beik
Vali-e-Asr University of Rafsanjan, beik.fatemeh@gmail.com

Salman Ahmadi-Asl
Vali-e-Asr University of Rafsanjan, s.ahmadiasl@stu.vru.ac.ir
AN ITERATIVE ALGORITHM FOR $\eta$-(ANTI)-HERMITIAN LEAST-SQUARES SOLUTIONS OF QUATERNION MATRIX EQUATIONS

FATEMEH PANJEH ALI BEIK† AND SALMAN AHMADI-ASL†.

Abstract. Recently, some research has been devoted to finding the explicit forms of the $\eta$-Hermitian and $\eta$-anti-Hermitian solutions of several kinds of quaternion matrix equations and their associated least-squares problems in the literature. Although exploiting iterative algorithms is superior than utilizing the explicit forms in application, hitherto, an iterative approach has not been offered for finding $\eta$-(anti)-Hermitian solutions of quaternion matrix equations. The current paper deals with applying an efficient iterative manner for determining $\eta$-Hermitian and $\eta$-anti-Hermitian least-squares solutions corresponding to the quaternion matrix equation $AXB + CYD = E$. More precisely, first, this paper establishes some properties of the $\eta$-Hermitian and $\eta$-anti-Hermitian matrices. These properties allow for the demonstration of how the well-known conjugate gradient least-squares (CGLS) method can be developed for solving the mentioned problem over the $\eta$-Hermitian and $\eta$-anti-Hermitian matrices. In addition, the convergence properties of the proposed algorithm are discussed with details. In the circumstance that the coefficient matrices are ill-conditioned, it is suggested to use a preconditioner for accelerating the convergence behavior of the algorithm. Numerical experiments are reported to reveal the validity of the elaborated results and feasibility of the proposed iterative algorithm and its preconditioned version.

Key words. Quaternion matrix equation, $\eta$-Hermitian matrix, $\eta$-Anti-Hermitian matrix, Iterative algorithm, Convergence, Preconditioner.

AMS subject classifications. 15A24, 65F10.

1. Introduction. Let us first present some notations and symbols used throughout this paper. We use $\mathbb{Q}^{m \times n}$ to refer the set of all $m \times n$ matrices over the quaternion ring

$$\mathbb{Q} = \{a_1 + a_2i + a_3j + a_4k \mid i^2 = j^2 = k^2 = ijk = -1, \ a_1, a_2, a_3, a_4 \in \mathbb{R}\},$$

and $\mathbb{R}^{m \times n}$ stands for the set of all $m \times n$ real matrices. For a given $m \times n$ matrix $A$, the symbols $A^T$, $\bar{A}$, $A^H$ and $\text{tr}(A)$ are respectively utilized to represent the transpose, the conjugate, the conjugate transpose and the trace of $A$. The real part of a quaternion $a$ is denoted by $\text{Re}(a)$, i.e., if $a = a_1 + a_2i + a_3j + a_4k$ then $\text{Re}(a) = a_1$. For given $a = a_1 + a_2i + a_3j + a_4k \in \mathbb{Q}$, the conjugate of $a$ is defined by $\bar{a} = a_1 - a_2i - a_3j - a_4k$. 

Received by the editors on January 15, 2015. Accepted for publication on June 15, 2015. Handling Editor: James G. Nagy.

†Department of Mathematics, Vali-e-Asr University of Rafsanjan, PO Box 518, Rafsanjan, Iran (f.beik@vru.ac.ir, beik.fatemeh@gmail.com; s.ahmadiasl@stu.vru.ac.ir).
The inner product over $\mathbb{Q}^{m \times n}$ is defined as follows:

$$\langle A, B \rangle = \text{Re}(\text{tr}(B^H A)) \quad \text{for} \ A, B \in \mathbb{Q}^{m \times n},$$

and the induced matrix norm is specified by

$$\|A\| = \sqrt{\langle A, A \rangle} = \sqrt{\text{Re}(\text{tr}(A^H A))}.$$

As a natural way, the inner product over $\mathbb{Q}^{m \times n} \times \mathbb{Q}^{m \times n}$ can be elucidated by

$$\langle [A_1, A_2], [B_1, B_2] \rangle = \langle A_1, B_1 \rangle + \langle A_2, B_2 \rangle,$$

where $[A_1, A_2]$ and $[B_1, B_2]$ belong to $\mathbb{Q}^{m \times n} \times \mathbb{Q}^{m \times n}$. The set of all $n \times n$ real symmetric and anti-symmetric matrices are respectively indicated by $\mathbb{S}\mathbb{R}^{n \times n}$ and $\mathbb{A}\mathbb{S}\mathbb{R}^{n \times n}$. A matrix $A \in \mathbb{Q}^{n \times n}$ is $\eta$-Hermitian if $A^\eta H = A$ and it is called $\eta$-anti-Hermitian if $A^\eta H = -A$ where $A^\eta H = -\eta A^H \eta$ and $\eta \in \{i, j, k\}$. The set of all $\eta$-Hermitian and $\eta$-anti-Hermitian matrices are respectively signified by $\eta\mathbb{H}\mathbb{Q}^{n \times n}$ and $\eta\mathbb{A}\mathbb{Q}^{n \times n}$.

The linear matrix equations have a crucial role in many branches of applied and pure mathematics such as control and system theory, stability theory, perturbation analysis, etc; for further details one may refer to [1, 2, 9, 18, 42, 43, 44] and the references therein. So far, several types of iterative algorithms have been proposed for solving various kinds of (coupled) matrix equations in the literature. For instance, Beik and Salkuyeh [3] have presented the Global Full Orthogonalization Method (Gl-FOM) and Global Generalized Minimum Residual (Gl-GMRES) method to determine the unique solution group $(X_1, X_2, \ldots, X_p)$ of the subsequent coupled linear matrix equations

$$\sum_{j=1}^{p} A_{ij}X_j B_{ij} = C_i,$$

where the nonsingular matrices $A_{ij} \in \mathbb{R}^{n \times n}$ and $B_{ij} \in \mathbb{R}^{n \times n}$ and the right hand side $C_i \in \mathbb{R}^{n \times n}$ are known matrices for $i, j = 1, 2, \ldots, p$. In [4], an efficient algorithm has been presented to solve the next coupled Sylvester-transpose matrix equations over the generalized centro-symmetric matrices

$$\sum_{j=1}^{q} A_{ij}X_j B_{ij} + C_{ij}X_j^TD_{ij} = C_i, \quad i = 1, \ldots, p,$$

in which $A_{ij}, C_{ij} \in \mathbb{R}^{n \times n}$, $B_{ij}, D_{ij} \in \mathbb{R}^{n \times s}$, and $C_i \in \mathbb{R}^{n \times s}$ are given matrices and the matrices $X_j \in \mathbb{R}^{n \times n}$ are unknown for $j = 1, 2, \ldots, q$. By making use of convex optimization theory, Cai and Chen [8] have proposed an iterative algorithm for finding the least-squares bisymmetric solutions of the coupled matrix equations
(A_1X B_1, A_2X B_2) = (C_1, C_2). Ding and Chen \cite{15, 16, 17} have developed some stationary iterative methods named gradient-based iterative methods to solve some kinds of matrix equations. In \cite{11, 13}, Dehghan and Hajarian have developed the well-known conjugate gradient (CG) method to solve the ensuring coupled matrix equations

\[
\begin{align*}
\sum_{i=1}^{l} A_i X B_i + \sum_{i=1}^{l} C_i Y D_i &= M \\
\sum_{i=1}^{l} E_i X F_i + \sum_{i=1}^{l} G_i Y H_i &= N
\end{align*}
\]

and

\[
\begin{align*}
A_{11} X_1 B_{11} + A_{12} X_2 B_{12} + \cdots + A_{1l} X_l B_{1l} &= C_1 \\
A_{21} X_1 B_{21} + A_{22} X_2 B_{22} + \cdots + A_{2l} X_l B_{2l} &= C_2 \\
&\vdots \\
A_{ll} X_1 B_{ll} + A_{l2} X_2 B_{ll} + \cdots + A_{ll} X_l B_{ll} &= C_l
\end{align*}
\]

respectively. In \cite{12}, the extended version of the CG method has been examined to solve the succeeding system of matrix equations

\[
\sum_{j=1}^{p} A_{ij} X_j B_{ij} = C_i, \quad i = 1, 2, \ldots, p,
\]

over the generalized bisymmetric matrices. In \cite{14}, the authors have exploited the idea of the CG method to construct an iterative algorithm to find the generalized reflexive and anti-reflexive solutions of the following system of linear operator equations

\[
\begin{align*}
\mathcal{F}_1(X) &= A_1 \\
\mathcal{F}_2(X) &= A_2 \\
&\vdots \\
\mathcal{F}_n(X) &= A_n
\end{align*}
\]

The outline of this paper is organized as follows. Before ending the current section, we momentarily describe our inspiration for presenting this paper and express two main problems of concern. Section 2 is devoted to proving some properties of the \(\eta\)-Hermitian and \(\eta\)-anti-Hermitian matrices. In Section 3 we propose an iterative algorithm to solve our main mentioned problems and analyze the convergence of the proposed algorithm. The preconditioned version of the algorithm is proposed to accelerate the convergence rate of the method in Section 4. In Section 5 some numerical examples are solved to demonstrate the validity of the established results and effectiveness of the proposed algorithm. Finally the paper is ended with a brief conclusion and some suggestions for further works in Section 6.
Iterative Algorithm for \( \eta \)-(Anti)-Hermitian Solutions of Quaternion Matrix Equations

1.1. Motivations and highlight points. Recently, there is a growing interest to find the solutions of quaternion matrix equations. A linear quaternion matrix equation is a linear matrix equation over the quaternion ring; see [20, 21, 22, 25, 30, 31] for more details. We would like to comment here that the quaternion matrices and quaternion matrix equations arise in the fields of quantum physics, signal and color image processing, computer science and so on; for further details see [3, 6, 23, 25, 33] and the references therein. Lately, some research has focused on developing iterative methods to solve such problems, for instance one may refer to [22, 28, 35].

Obtaining the explicit forms of the \( \eta \)-Hermitian and \( \eta \)-anti-Hermitian solutions of linear quaternion matrix equations have been studied in [19, 39, 40]. It is well-known that the explicit forms are not computationally practical even in the case that the size of coefficient matrices are moderate. More precisely, Yuan et al. [40] have derived the explicit forms of \( \eta \)-Hermitian and \( \eta \)-anti-Hermitian solutions of \( AXB + CYD = E \) and also the explicit forms of the solution pair \([X, Y]\) with least-norm theoretically. Nevertheless there exist some drawbacks for utilizing these forms such as:

- The complicated forms of the obtained explicit solutions and the requirement to compute the Moore-Penrose pseudoinverse which would be expensive for large scale quaternion least-squares problems.

- For deriving solutions by means of obtained explicit forms, Kronecker product of matrices is required to be explicitly formed. Therefore the size of original problem increases and causes high computational cost.

These disadvantages in using explicit forms inspire us to develop an iterative algorithm for computing the \( \eta \)-Hermitian and \( \eta \)-anti-Hermitian solutions of \( AXB + CYD = E \). We would like to emphasize that, to the best of our knowledge, a neat and feasible iterative technique to solve such solutions has not been examined so far. In the literature, the well-known CGLS method has been successfully implemented to solve different types of matrix equations; the reader may refer to [26, 27, 29, 30, 31] and the references therein. Hence, our main goal is to extend the CGLS method for resolving the least-squares problem corresponding to the quaternion matrix equation \( AXB + CYD = E \) over the \( \eta \)-Hermitian and \( \eta \)-anti-Hermitian matrix pair \([X, Y]\).

To this end, we first need to establish some properties of \( \eta \)-Hermitian and \( \eta \)-anti-Hermitian matrices. For improving the convergence speed of the proposed algorithm, the application of a preconditioner is examined.

1.2. Problem reformulation. For simplicity, we consider the linear operator \( \mathcal{M} : Q^{n \times n} \times Q^{n \times n} \rightarrow Q^{n \times n} \) such that

\[
\mathcal{M}(X, Y) = \mathcal{M}_1(X) + \mathcal{M}_2(Y),
\]
where $M_i : \mathbb{Q}^{n \times n} \to \mathbb{Q}^{n \times n}$ for $i = 1, 2$. The linear operators $M_1$ and $M_2$ are specified by

$$M_1(X) = AXB \quad \text{and} \quad M_2(Y) = CYD.$$  

Using (1.1), the matrix equation $AXB + CYD = E$ can be reformulated in the following form

$$\mathcal{M}(X, Y) = E.$$  

**Definition 1.1.** Let $L$ be a linear operator from $\mathbb{Q}^{m \times n}$ onto $\mathbb{Q}^{m \times n}$, then the adjoint of $L$ is denoted by $L^*$ and satisfies

$$\langle L(X), Z \rangle = \langle X, L^*(Z) \rangle.$$  

By straightforward computations and using the fact that $\text{tr}(XY) = \text{tr}(YX)$, we may conclude that the adjoint operators corresponding to the linear operators $M_1$ and $M_2$ have the next forms

$$M_1^*(Z) = A^HZB^H \quad \text{and} \quad M_2^*(Z) = C^HZD^H,$$

which implies that $M^* = (M_1^*, M_2^*)$.

In this paper, we focus on the solutions of the subsequent two problems.

**Problem 1.2.** Presume that the $n \times n$ quaternion matrices $A, B, C, D$ and $E$ are given. Find the matrices $X$ and $Y$ such that

$$\|AXB + CYD - E\| \quad \text{is minimized},$$

and $X \in \eta \mathbb{H}^{Q^{n \times n}}$ and $Y \in \eta \mathbb{A}^{Q^{n \times n}}$.

**Problem 1.3.** Presume that the $n \times n$ quaternion matrices $A, B, C, D$ and $E$ are given. Suppose that $S_E$ stands for the solution set of Problem 1.2. Given $\tilde{X} \in \eta \mathbb{H}^{Q^{n \times n}}$ and $\tilde{Y} \in \eta \mathbb{A}^{Q^{n \times n}}$, find $X^* \in \eta \mathbb{H}^{Q^{n \times n}}$ and $Y^* \in \eta \mathbb{A}^{Q^{n \times n}}$ such that

$$\|\tilde{X} - X^*\| + \|\tilde{Y} - Y^*\| = \min_{X,Y \in S_E} \|\tilde{X} - X\| + \|\tilde{Y} - Y\|.$$  

2. On the $\eta$-Hermitian and $\eta$-anti-Hermitian matrices. In this section, we scrutinize some properties of the $\eta$-Hermitian and $\eta$-anti-Hermitian matrices which have an essential role for constructing our algorithm to solve the mentioned problems. Exploiting the derived results, two new linear operators are also expounded.
Iterative Algorithm for \( \eta \)-(Anti)-Hermitian Solutions of Quaternion Matrix Equations

For an arbitrary given \( A \in \mathbb{R}^{n \times n} \), suppose that the operators \( \mathcal{H}(A) \) and \( \mathcal{S}(A) \) give the Hermitian and skew-Hermitian parts of matrix \( A \), respectively. That is,

\[
\mathcal{H} : \mathbb{R}^{n \times n} \rightarrow \mathbb{SR}^{n \times n}
\]

\[
A \mapsto \mathcal{H}(A) = \frac{1}{2}(A + A^T),
\]

and,

\[
\mathcal{S} : \mathbb{R}^{n \times n} \rightarrow \mathbb{ASR}^{n \times n}
\]

\[
A \mapsto \mathcal{S}(A) = \frac{1}{2}(A - A^T).
\]

The following theorem reveals that the set of all \( n \times n \) quaternion matrices can be written as a direct sum of \( \eta \mathbb{HQ}^{n \times n} \) and \( \eta \mathbb{AQ}^{n \times n} \). The theorem supplies a cardinal tool for extending the CGLS algorithm to solve Problems 1.2 and 1.3.

**Theorem 2.1.** Presume that \( \eta \mathbb{HQ}^{n \times n} \) and \( \eta \mathbb{AQ}^{n \times n} \) stand for the set of all \( n \times n \) \( \eta \)-Hermitian and \( \eta \)-anti-Hermitian matrices, respectively. Then,

\[
\mathbb{Q}^{n \times n} = \eta \mathbb{HQ}^{n \times n} \oplus \eta \mathbb{AQ}^{n \times n}.
\]

**Proof.** We only demonstrate the validity of the assertion for \( \eta = i \); the strategy of the proof for \( \eta = j \) (or \( \eta = k \)) is similar. For clarification, the proof is divided into three steps as follows:

**Step 1.** In this step, we show that for an arbitrary given matrix \( A \in \mathbb{Q}^{n \times n} \), there exist matrices \( B \in \mathbb{iHQ}^{n \times n} \) and \( C \in \mathbb{iAQ}^{n \times n} \) such that

\[
(2.1)\quad A = B + C.
\]

To do so, it is required to characterize the structure of the members of \( \mathbb{iHQ}^{n \times n} \) and \( \mathbb{iAQ}^{n \times n} \). Let \( B = B_1 + B_2i + B_3j + B_4k \) be an \( i \)-Hermitian matrix, i.e., \(-iB^Hi = B\). Straightforward computations reveal that

\[
(2.2)\quad B_1 + B_2i + B_3j + B_4k = B_1^T - B_2^T i + B_3^T j + B_4^T k.
\]

In view of (2.2), it can be deduced that

\[
B_1^T = B_1, \quad B_2^T = -B_2, \quad B_3^T = B_3, \quad \text{and} \quad B_4^T = B_4.
\]

On the other hand if \( B_1^T = B_1, \ B_2^T = -B_2, \ B_3^T = B_3 \) and \( B_4^T = B_4 \), then it is not difficult to verify that \( B = B_1 + B_2i + B_3j + B_4k \) is an \( i \)-Hermitian matrix. With a similar manner, it can be seen that \( C \in \mathbb{iAQ}^{n \times n} \) (i.e., \( iC^Hi = C \)) if and only...
if $C_1^T = -C_1$, $C_2^T = C_2$, $C_3^T = -C_3$, and $C_4^T = -C_4$. Consequently, we may set $A = B + C$ in which

$$B = H(A_1) + S(A_2)i + H(A_3)j + H(A_4)k,$$
$$C = S(A_1) + H(A_2)i + S(A_3)j + S(A_4)k,$$ 

where the operators $H$ and $S$ are defined as before. Evidently, $B \in i\mathbb{H}Q^{n \times n}$ and $C \in i\mathbb{A}Q^{n \times n}$.

**Step 2.** Here it is demonstrated that if $B \in i\mathbb{H}Q^{n \times n}$ and $C \in i\mathbb{A}Q^{n \times n}$ then $\langle B, C \rangle = 0$.

Suppose that $B = B_1 + B_2i + B_3j + B_4k$ and $C = C_1 + C_2i + C_3j + C_4k$, then

$$C^H B = (C_1^T - C_2^T i - C_3^T j - C_4^T k)(B_1 + B_2i + B_3j + B_4k)$$
$$= (C_1^T B_1 + C_2^T B_2 + C_3^T B_3 + C_4^T B_4) + (C_1^T B_2 - C_2^T B_1 - C_3^T B_4 + C_4^T B_3)i$$
$$+ (C_1^T B_3 + C_2^T B_4 - C_3^T B_1 - C_4^T B_2)j + (C_1^T B_4 - C_2^T B_3 + C_3^T B_2 - C_4^T B_1)k.$$ 

Thence, we have

$$\langle B, C \rangle = \text{Re}(\text{tr}(C^H B)) = \text{tr}(C_1^T B_1 + C_2^T B_2 + C_3^T B_3 + C_4^T B_4).$$

From the earlier discussions in the first step, it is not onerous to see that

$$\text{Re}(\text{tr}(C_i^T B_i)) = 0 \quad \text{for } i = 1, 2, 3, 4.$$

**Step 3.** In this part, it is shown that the splitting (2.1) is unique. In fact, if $A = B + C = B' + C'$ such that $B, B' \in i\mathbb{H}Q^{n \times n}$ and $C, C' \in i\mathbb{A}Q^{n \times n}$ then $B - B' = C - C'$. On the other hand, $B - B' \in i\mathbb{H}Q^{n \times n}$ and $C - C' \in i\mathbb{A}Q^{n \times n}$. Now Step 2 implies that $B = B'$ and $C = C'$. The result can be concluded immediately from Steps 1, 2 and 3. \(\blacksquare\)

The following corollary is a direct conclusion of Theorem 2.1.

**Corollary 2.2.** For each $A \in \mathbb{Q}^{n \times n}$, there exist unique matrices $U_1 \in \eta\mathbb{H}Q^{n \times n}$ and $U_2 \in \eta\mathbb{A}Q^{n \times n}$ such that $A = U_1^n + U_2^n$ and $\langle U_1^n, U_2^n \rangle = 0$.

Based on the above lemma, we introduce two useful linear operators as follows:

$$\mathcal{L}_1^n : \mathbb{Q}^{n \times n} \to \eta\mathbb{H}Q^{n \times n}$$
$$U \mapsto \mathcal{L}_1^n (U) = U_1^n,$$

and

$$\mathcal{L}_2^n : \mathbb{H}^{n \times n} \to \eta\mathbb{A}Q^{n \times n}$$
$$U \mapsto \mathcal{L}_2^n (U) = U_2^n.$$
We finish the current section by presenting the next two useful remarks.

**Remark 2.3.** In view of Theorem 2.1, we conclude that for given arbitrary matrices $U \in \mathbb{Q}^{n \times n}$, $Q \in \eta\mathbb{Q}^{n \times n}$ and $W \in \eta\mathbb{Q}^{n \times n}$, we have

\begin{equation}
\langle U, Q \rangle = \langle L_1^{\eta}(U), Q \rangle \quad \text{and} \quad \langle U, W \rangle = \langle L_2^{\eta}(U), W \rangle.
\end{equation}

**Remark 2.4.** Note that Theorem 2.1 helps us to derive the explicit forms of the operators $L_1^{\eta}(U)$ and $L_2^{\eta}(U)$. For instance, consider the case that $\eta = i$. Then,

\begin{align*}
L_1^{\eta}(U) &= \mathcal{H}(U_1) + \mathcal{S}(U_2)i + \mathcal{H}(U_3)j + \mathcal{H}(U_4)k, \\
L_2^{\eta}(U) &= \mathcal{S}(U_1) + \mathcal{H}(U_2)i + \mathcal{S}(U_3)j + \mathcal{S}(U_4)k,
\end{align*}

where the operators $\mathcal{H}$ and $\mathcal{S}$ are defined as before and $U = U_1 + U_2i + U_3j + U_4k$.

3. **Proposed iterative scheme.** The well-known CGLS method for resolving the following least-squares problem

\[ \|Ax - b\|_2, \]

is obtained by applying the conjugate gradient (CG) algorithm on the normal equations $A^TAx = A^Tb$ and the derived algorithm is called CGLS; see [7, 36] for more details. We would like to comment here that the CGLS method is also known as the CGNR method; for more details see [32, Chapter 8].

As a matter of fact, the CGLS method is an oblique projection technique onto

\[ \mathcal{K}_n = \left\{ r_0, (A^TA)^{-1}r_0, \ldots, (A^TA)^{-n}r_0 \right\}, \]

and orthogonal to $\mathcal{L}_n = \mathcal{A}\mathcal{K}_n$ where $r_0 = A^Tb - A^TxAx$ and $x_0$ is a given initial guess. It can be theoretically shown that in the absent of round-off errors, the CGLS method converges to the exact solution of the normal equations $A^TAx = A^Tb$ within finite number of steps [32].

The main objective of the present section is to exploit the idea of the CGLS approach for constructing an iterative manner for resolving Problems 1.2 and 1.3 and study its convergence properties. To this end, we mainly utilize the results obtained in the previous section. In the rest of this section, first, an iterative algorithm is offered for determining the solution set of Problem 1.2. Afterward an approach is proposed to solve the second problem.

3.1. **An iterative algorithm and its convergence properties.** Developing the idea of the CGLS method and using the linear operators introduced in the previous
section, we may propose an iterative algorithm to determine the least-squares the \( \eta \)-Hermitian and \( \eta \)-anti-Hermitian solution pair of the linear matrix equation \( AXB + CYD = E \). The proposed algorithm is given by Algorithm 3.1.

Algorithm 3.1. The developed CGLS method for Problem 1.2.

Data: Input \( A \in \mathbb{Q}^{n \times n} \), \( B \in \mathbb{Q}^{n \times n} \), \( C \in \mathbb{Q}^{n \times n} \), \( D \in \mathbb{Q}^{n \times n} \), \( E \in \mathbb{Q}^{n \times n} \), \( X(0) \in \eta HQ^{n \times n} \), \( Y(0) \in \eta AQ^{n \times n} \) and choose tolerance \( \epsilon \).

Initialization:
- \( k = 0 \);
- \( R(0) = E - AX(0)B - CY(0)D \);
- \( P_{0,x} = L_{1}^{\eta} (M_{1}^{\ast} (R(0))) \);
- \( P_{0,y} = L_{2}^{\eta} (M_{2}^{\ast} (R(0))) \);
- \( Q_{0,x} = P_{0,x} \);
- \( Q_{0,y} = P_{0,y} \);

While \( \| R(0) \| > \epsilon \) or \( \| P_{0,x} \|^2 + \| P_{0,y} \|^2 > \epsilon \) Do:
- \( \alpha_{k} = \frac{\| P_{0,x} \|^2 + \| P_{0,y} \|^2}{\| M(Q_{k,x} , Q_{k,y}) \|} \);
- \( X(k + 1) = X(k) + \alpha_{k} Q_{k,x} \);
- \( Y(k + 1) = Y(k) + \alpha_{k} Q_{k,y} \);
- \( R(k + 1) = R(k) - \alpha_{k} M(Q_{k,x} , Q_{k,y}) \);
- \( P_{k+1,x} = L_{1}^{\eta} (M_{1}^{\ast} (R(k + 1))) \);
- \( P_{k+1,y} = L_{2}^{\eta} (M_{2}^{\ast} (R(k + 1))) \);
- \( \beta_{k} = \frac{\| P_{k+1,x} \|^2 + \| P_{k+1,y} \|^2}{\| P_{k,x} \|^2 + \| P_{k,y} \|^2} \);
- \( Q_{k+1,x} = P_{k+1,x} + \beta_{k} Q_{k,x} \);
- \( Q_{k+1,y} = P_{k+1,y} + \beta_{k} Q_{k,y} \);
- \( k = k + 1 \);

EndDo

We will inspect the convergence behavior of Algorithm 3.1. First we need to recall the next lemma which is called “Projection Theorem” and its proof can be found in [37].

Lemma 3.2. Let \( X \) be a finite-dimensional inner product space, \( M \) be a subspace of \( X \), and \( M^\perp \) be the orthogonal complement of \( M \). For a given \( x \in X \), always, there exists an \( m_{0} \in M \) such that \( \| x - m_{0} \| \leq \| x - m \| \) for all \( m \in M \) where \( \| . \| \) is the norm associated with the inner product defined in \( X \). Moreover, \( m_{0} \in M \) is the unique minimization vector in \( M \) if and only if \( x - m_{0} \) \( \perp M \) which is equivalent to say that \( x - m_{0} \in M^\perp \).
Iterative Algorithm for \(\eta\)-(Anti)-Hermitian Solutions of Quaternion Matrix Equations

**Theorem 3.3.** Suppose that \(\tilde{R} = E - A\tilde{X}B - CYD\) where \([\tilde{X},\tilde{Y}]\in\eta\HH\Q^{n\times n}\times\eta\AQ^{n\times n}\). If \(L_i^0(M_i^0(\tilde{R})) = 0\) for \(i = 1, 2\), then \([\tilde{X},\tilde{Y}]\) is a solution pair of Problem 3.2.

**Proof.** Consider the following linear subspace

\[ \mathcal{V} = \{ V \mid V =AXB + CYD, \ X \in \eta\HH\Q^{n\times n} \text{ and } Y \in \eta\AQ^{n\times n} \} . \]

Presume that \(\tilde{X} \in \eta\HH\Q^{n\times n}\) and \(\tilde{Y} \in \eta\AQ^{n\times n}\), then \(\tilde{E} = A\tilde{X}B + CYD \in \mathcal{V}\). Lemma 3.2 implies that \([\tilde{X},\tilde{Y}]\) is a least-squares solution of \(AXB + CYD = E\) if and only if

\[
\langle E - \tilde{E},AXB + CYD \rangle = \langle \tilde{R},AXB + CYD \rangle = 0,
\]

for all \(X \in \eta\HH\Q^{n\times n}\) and \(Y \in \eta\AQ^{n\times n}\). On the other hand, invoking (2.3), we have

\[
\langle \tilde{R},AXB + CYD \rangle = \langle A^H\tilde{R}B^H,X \rangle + \langle C^H\tilde{R}D^H,Y \rangle
\]

\[
= \langle L_i^0(A^H\tilde{R}B^H),X \rangle + \langle L_i^0(C^H\tilde{R}D^H),Y \rangle
\]

\[
= \langle L_i^0(M_i^0(\tilde{R})),X \rangle + \langle L_i^0(M_i^0(\tilde{R})),Y \rangle .
\]

Consequently, from (3.1), the assumptions \(L_i^0(M_i^0(\tilde{R})) = 0\) and \(L_i^0(M_i^0(\tilde{R})) = 0\) ensure that \([\tilde{X},\tilde{Y}]\) is the least-squares solution pair of \(AXB + CYD = E\) which completes the proof.

**Lemma 3.4.** \(\exists \) Let \(f(Z)\) be a continuous, differentiable and convex function on subspace \(\mathcal{Y}\), then there exists \(Z^* \in \mathcal{Y}\) such that \(f(Z^*) = \min_{Z \in \mathcal{Y}} f(Z)\) if and only if the projection of the gradient \(\nabla f(Z^*)\) onto \(\mathcal{Y}\) equals to zero.

**Remark 3.5.** Using Lemma 3.4, we can present an alternative proof for Theorem 3.3. To this end, we define

\[
f : \Q^{n\times n} \times \Q^{n\times n} \to \mathbb{R}
\]

\[
[X,Y] \mapsto f(X,Y) = \|AXB + CYD - E\|^2.
\]

It is not difficult to establish that \(f(X,Y)\) is a continuous, differentiable and convex function. With a similar approach used in [26], the gradient of \(f(X,Y)\) can be obtained. To do so, consider the next auxiliary function

\[
g : \mathbb{R} \to \mathbb{R}
\]

\[
t \mapsto g(t) = f(X + tP_1, Y + tP_2)
\]

in which \(P_1\) and \(P_2\) are arbitrary quaternion matrices. It can be seen that

\[
g'(0) = \langle \nabla_X f(X,Y), P_1 \rangle + \langle \nabla_Y f(X,Y), P_2 \rangle .
\]
From (3.2) and some straightforward computations, we get
\[ g'(0) = 2 \langle AXB + CD - E, AP_1B \rangle + 2 \langle AXB + CD - E, CP_2D \rangle \]
\[ = 2 \langle A^H (AXB + CD - E) B^H, P_1 \rangle + 2 \langle C^H (AXB + CD - E) D^H, P_2 \rangle. \]  
(3.5)

In view of (3.4) and (3.5), it turns out that
\[ \nabla_X f(X, Y) = 2A^H (AXB + CD - E) B^H = -2A^H RB^H, \]
\[ \nabla_Y f(X, Y) = 2C^H (AXB + CD - E) D^H = -2C^H RD^H. \]

Now from the discussion of this remark, it is not difficult to conclude that if
\[ L_1^0 \left( A^H RB^H \right) = 0 \quad \text{and} \quad L_2^0 \left( C^H RD^H \right) = 0, \]
where \( R = E - AXB - CD, \) then \([X, Y]\) is the minimizer of (3.2) over the \( \eta \)-Hermitian and \( \eta \)-anti-Hermitian matrices.

The next theorem can be proved by mathematical induction and its proof is elaborated in Appendix A.

**Theorem 3.6.** Suppose that \( k \) steps of Algorithm 3.1 have been performed, i.e., \( \|P_{l,x}\|^2 + \|P_{l,y}\|^2 \neq 0 \) and \( \mathcal{M}(Q_{l,x}, Q_{l,y}) \neq 0 \) for \( l = 0, 1, \ldots, k. \) The sequences \( P_{l,x}, P_{l,y}, Q_{l,x} \) and \( Q_{l,y} \) produced by Algorithm 3.1 satisfy the following statements
\[ \langle P_{i,x}, P_{j,x} \rangle + \langle P_{i,y}, P_{j,y} \rangle = 0, \]  
(3.6)
\[ \langle \mathcal{M}(Q_{i,x}, Q_{i,y}), \mathcal{M}(Q_{j,x}, Q_{j,y}) \rangle = 0, \]  
(3.7)
\[ \langle P_{i,x}, Q_{j,x} \rangle + \langle P_{i,y}, Q_{j,y} \rangle = 0, \]  
(3.8)
for \( i, j = 0, 1, 2, \ldots, k \) (\( i \neq j \)).

**Remark 3.7.** Note that if in a specific step of Algorithm 3.1 say \( l \)-th step, we face to the situation that \( \|P_{l,x}\|^2 + \|P_{l,y}\|^2 = 0. \) Then,
\[ P_{l,x} = L_1^0 \left( \mathcal{M}_1^* \left( R(l) \right) \right) = 0 \quad \text{and} \quad P_{l,y} = L_2^0 \left( \mathcal{M}_2^* \left( R(l) \right) \right) = 0, \]
which, in view of Theorem 3.3, implies that \([X(l), Y(l)]\) is a solution pair of Problem 1.2. On the other hand, the subsequent computations reveal that if \( \mathcal{M}(Q_{l,x}, Q_{l,y}) = 0, \) then \( \|P_{l,x}\|^2 + \|P_{l,y}\|^2 = 0, \)
\[ \langle \mathcal{M}(Q_{l,x}, Q_{l,y}), R(l) \rangle = \langle \mathcal{M}_1(Q_{l,x}) + \mathcal{M}_2(Q_{l,y}), R(l) \rangle = \langle Q_{l,x}, L_1^0 \left( \mathcal{M}_1^* \left( R(l) \right) \right) \rangle + \langle Q_{l,y}, L_2^0 \left( \mathcal{M}_2^* \left( R(l) \right) \right) \rangle \]
\[ = \langle Q_{l,x}, P_{l,x} \rangle + \langle Q_{l,y}, P_{l,y} \rangle \]
\[ = \|P_{l,x}\|^2 + \|P_{l,y}\|^2. \]
The following theorem can be deduced from Theorem 3.8 immediately. The proof of the theorem is straightforward and is omitted.

**Theorem 3.8.** For any initial matrix pair \([X(0), Y(0)]\) \(\in \eta HQ^{n \times n} \times \eta HQ^{n \times n}\), Algorithm 3.1 converges to the exact solution of Problem 1.2 within finite number of steps in the absence of round-off errors.

In the sequent part of this subsection, our goal is to demonstrate that the least-norm solution of Problem 1.2 can be obtained by choosing appropriate initial matrices. To this end, let us define the next linear subspace over \(\eta HQ^{n \times n} \times \eta HQ^{n \times n}\),

\[
W = \{[W_1, W_2]: W_1 = L_1^\eta (A^H Z B^H) \text{ and } W_2 = L_2^\eta (C^H Z D^H)\}.
\]

where \(Z \in Q^{n \times n}\) is an arbitrary given matrix.

**Lemma 3.9.** Let \([\hat{X}, \hat{Y}]\) be a solution pair of Problem 1.2. Suppose that \([N_1, N_2] \in \eta HQ^{n \times n} \times \eta HQ^{n \times n}\). Then, \([\hat{X} + N_1, \hat{Y} + N_2]\) is a solution pair of Problem 1.2 if and only if \(AN_1 B + CN_2 D = 0\).

**Proof.** If \([\hat{X} + N_1, \hat{Y} + N_2] \in \eta HQ^{n \times n} \times \eta HQ^{n \times n}\) and \(AN_1 B + CN_2 D = 0\), then

\[
\left\| \begin{bmatrix} A & \hat{X} + N_1 \\ B + C (\hat{Y} + N_2) \end{bmatrix} D - E \right\|^2 = \left\| A \hat{X} B + C \hat{Y} D - E \right\|^2,
\]

which ensures that \([\hat{X} + N_1, \hat{Y} + N_2]\) is a solution pair of Problem 1.2. Conversely, suppose that \([\hat{X} + N_1, \hat{Y} + N_2] \in \eta HQ^{n \times n} \times \eta HQ^{n \times n}\) is a solution pair of Problem 1.2. By the hypophysis, \([\hat{X}, \hat{Y}]\) is a solution pair of Problem 1.2. Therefore

\[
\left\| A \hat{X} B + C \hat{Y} D - E \right\|^2 = \left\| A (\hat{X} + N_1) B + C (\hat{Y} + N_2) D - E \right\|^2.
\]

As \([\hat{X}, \hat{Y}]\) is a least-squares solution of \(AXB + CYD = E\), Lemma 3.2 implies that

\[
\left\langle A \hat{X} B + C \hat{Y} D - E, A \hat{X} B + C \hat{Y} D \right\rangle = 0.
\]

In view of the above relation and the next computations, we have

\[
\left\| A (\hat{X} + N_1) B + C (\hat{Y} + N_2) D - E \right\|^2 = \left\| A \hat{X} B + C \hat{Y} D - E \right\|^2 + \|AN_1 B + CN_2 D\|^2
\]

\[
- 2 \left\langle A \hat{X} B + C \hat{Y} D - E, AN_1 B + CN_2 D \right\rangle
\]

\[
= \left\| A \hat{X} B + C \hat{Y} D - E \right\|^2 + \|AN_1 B + CN_2 D\|^2.
\]

Now the result follows from (3.10) immediately. \(\square\)

**Theorem 3.10.** If the initial matrix pair \([X(0), Y(0)]\) belongs to \(W\), then Algorithm 3.1 converges to the least-norm solution pair of Problem 1.2.
Proof. Suppose that the initial matrices are chosen such that \([X(0), Y(0)] \in \mathcal{W}\). Presume that \(\ell\) steps of Algorithm 3.1 have been performed. In this case, for the sequence of approximate solutions \(\{X(j)\}_{j=1}^{\ell} \) and \(\{Y(j)\}_{j=1}^{\ell}\) produced by Algorithm 3.1, we have \([X(j), Y(j)] \in \mathcal{W}\) for some \(Z \in \mathbb{Q}^{n \times n}\). Suppose that \(X(j) \to X^*\) and \(Y(j) \to Y^*\) as \(j \to \infty\). From Lemma 3.9, it reveals that any arbitrary solution pair \([X, Y]\) of Problem 1.2 can be expressed as \([X^* + N_1, Y^* + N_2]\) such that \([N_1, N_2] \in \eta \mathbb{H}^{n \times n} \times \eta \mathbb{Q}^{n \times n}\) and \(AN_1B + CN_2D = 0\). On the other hand,  
\[
\langle X^*, N_1 \rangle + \langle Y^*, N_2 \rangle = \langle L_1^0 (A^H Z B^H), N_1 \rangle + \langle L_2^0 (C^H Z D^H), N_2 \rangle = \langle A^H Z B^H, N_1 \rangle + \langle C^T Z D^T, N_2 \rangle = \langle Z, AN_1B + C N_2D \rangle = \langle Z, AN_1B + C N_2D \rangle.
\]
Hence, we get  
\[
\|X^* + N_1\|^2 + \|Y^* + N_2\|^2 = \|X^*\|^2 + \|Y^*\|^2 + \|N_1\|^2 + \|N_2\|^2 \geq \|X^*\|^2 + \|Y^*\|^2.
\]
This implies that the solution pair \([X^*, Y^*]\) is the least-norm solution pair of Problem 1.2.

The next theorem illustrates that the sequence of approximate solutions generated via Algorithm 3.1 satisfies a minimization property.

**Theorem 3.11.** Let \([X(0), Y(0)] \in \eta \mathbb{H}^{n \times n} \times \eta \mathbb{Q}^{n \times n}\) be an arbitrary initial matrix pair, then the generated matrix solution pair \([X(k), Y(k)]\) at the \(k\)-th iteration step is the minimizer of the following optimization problem  
\[
\min_{[X,Y] \in U_k} \|AXB + CYD - E\|^2,
\]
where the affine subspace \(U_k\) is expounded as follows:

\[
U_k = [X(0), Y(0)] + \text{span} \{[Q_{0,0}, Q_{0,0}], [Q_{1,x}, Q_{1,y}], \ldots, [Q_{k-1,x}, Q_{k-1,y}]\}.
\]

**Proof.** Consider an arbitrary matrix pair \([X,Y] \in U_k\). From (3.11), there exist real numbers \(\beta_0, \beta_2, \ldots, \beta_{k-1}\) such that  
\[
[X,Y] = [X(0), Y(0)] + \sum_{i=0}^{k-1} \beta_i [Q_{i,x}, Q_{i,y}].
\]
Iterative Algorithm for \(\psi\)-(Anti)-Hermitian Solutions of Quaternion Matrix Equations

We define the next function with respect to \(\beta_0, \beta_1, \ldots, \beta_{k-1}\),

\[
 f(\beta_0, \beta_1, \ldots, \beta_{k-1}) = \|AXB + CYD - E\|^2
\]

(3.12)

It is well-known that the function (3.12) is continuous and differentiable with respect to the variables \(\beta_0, \beta_1, \ldots, \beta_{k-1}\). Straightforward computations reveal that

\[
 f(\beta_0, \beta_2, \ldots, \beta_{k-1}) = \left\|A \left( X(0) + \sum_{i=0}^{k-1} \beta_i Q_{i,x} \right) B + C \left( Y(0) + \sum_{i=0}^{k-1} \beta_i Q_{i,y} \right) D - E \right\|^2
\]

(3.13)

where \(R(0) = E - AX(0)B - CY(0)D\).

From Algorithm 3.1, it is seen that \(R(0)\) can be written as

\[
 R(0) = R(i) + \gamma_{i-1} (AQ_{i-1,x}B + CQ_{i-1,y}D) + \gamma_{i-2} (AQ_{i-2,x}B + CQ_{i-2,y}D) + \cdots + \gamma_0 (AQ_{0,x}B + CQ_{0,y}D),
\]

(3.14)

for some \(\gamma_0, \gamma_1, \ldots, \gamma_{i-1}\). Note that Lemma 3.4 demonstrates that the necessary and sufficient conditions for minimizing \(f(\beta_0, \beta_1, \ldots, \beta_{k-1})\) are

\[
 \frac{\partial f(\beta_0, \beta_1, \ldots, \beta_{k-1})}{\partial \beta_i} = 0, \quad i = 0, 1, \ldots, k - 1.
\]

Hence, the optimal solution is obtained as soon as

\[
 \beta_i = \frac{\langle R(0), AQ_{i,x}B + CQ_{i,y}D \rangle}{\|AQ_{i,x}B + CQ_{i,y}D\|^2}, \quad i = 0, 1, \ldots, k - 1.
\]

Using Eqs. (3.7), (3.8), (3.13) and some computations, we can simplify (3.14) as follows:

\[
 \beta_i = \frac{\langle (R(i), AQ_{i,x}B + CQ_{i,y}D) \rangle}{\|AQ_{i,x}B + CQ_{i,y}D\|^2} = \langle \mathcal{L}^H (A^H R(i)B^H), Q_{i,x} \rangle + \langle \mathcal{L}^H (C^H R(i)D^H), Q_{i,y} \rangle
\]

\[
 = \frac{\langle P_{i,x}, Q_{i,x} \rangle + \langle P_{i,y}, Q_{i,y} \rangle}{\|AQ_{i,x}B + CQ_{i,y}D\|^2} = \frac{\|P_{i,x}\|^2 + \|P_{i,y}\|^2}{\|AQ_{i,x}B + CQ_{i,y}D\|^2} = \alpha_i, \quad i = 0, 1, \ldots, k - 1.
\]
Now the result follows from the following fact immediately,
\[ \min_{\beta_i} f(\beta_0, \beta_1, \ldots, \beta_{k-1}) = \min_{[X,Y] \in U_k} \|AXB + CYD - E\|^2. \]

**Remark 3.12.** Let us define the sets \( U_{k-1} \) and \( U_k \) as follows:
\[
U_{k-1} = [X(0), Y(0)] + \text{span}\{[Q_{0,x}, Q_{0,y}], [Q_{1,x}, Q_{1,y}], \ldots, [Q_{k-1,x}, Q_{k-1,y}]\},
\]
\[
U_k = [X(0), Y(0)] + \text{span}\{[Q_{0,x}, Q_{0,y}], [Q_{1,x}, Q_{1,y}], \ldots, [Q_{k,x}, Q_{k,y}]\}.
\]
It is obvious that \( U_{k-1} \subseteq U_k \) which illustrates that
\[
\|R(k)\|^2 = \min_{U_k} \|AXB + CYD - E\|^2 \leq \min_{U_{k-1}} \|AXB + CYD - E\|^2 = \|R(k-1)\|^2.
\]
Evidently, the above inequality demonstrates that the sequence of the norm of residuals produced by Algorithm 3.1 is decreasing which shows that the algorithm is convergent.

### 3.2. The solution of Problem 2.
In this subsection, we briefly express how the second problem can be solved using Algorithm 3.1. Suppose that a matrix pair \([\hat{X}, \hat{Y}] \in \eta HQ^{n \times n} \times \eta AQ^{n \times n}\) is given. Evidently, we may find the solution pair of Problem 1.3 by finding the least-norm solution pair \([Z, W]\) of the following least-squares problem
\[
(3.15) \min \|AZB + CWD - F\|,
\]
where \(Z = X - \hat{X}, W = Y - \hat{Y}\) and \(F = E - A\hat{X}B - C\hat{Y}D\).

To this end, we exploit Algorithm 3.1 with the initial matrices \([Z(0), W(0)] \in W\), e.g. \([Z(0), W(0)] = [0, 0]\), for determining the least-norm solution pair \([Z^*, W^*]\). Then the solution pair of Problem 1.3 can be obtained by setting \([X^*, Y^*] = [\hat{X} + Z^*, \hat{Y} + W^*]\).

### 4. Application of a preconditioner.
In this section, we propose a preconditioned form of Algorithm 3.1 for solving Problem 1.2 to accelerate the convergence of the algorithm. Exploiting the offered preconditioner can be profitable when the quaternion coefficient matrices are extremely ill-conditioned. First, we need to recall some theorems and definitions.

**Theorem 4.1.** \[38\] Let \(A \in Q^{m \times n}\) with rank\((A) = r\). Then there exist unitary quaternion matrices \(U \in Q^{m \times n}\) and \(V \in Q^{n \times n}\) such that
\[
U^H AV = \begin{pmatrix}
\Sigma_r & 0 \\
0 & 0
\end{pmatrix},
\]
Iterative Algorithm for \(\eta\)-\(\text{(Anti)}\)-Hermitian Solutions of Quaternion Matrix Equations

where \(\Sigma_r = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_r)\), \(\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0\), and \(\sigma_1, \sigma_2, \ldots, \sigma_r\) are the nonzero singular values of quaternion matrix \(A\).

Let \(A = A_1 + iA_2 + jA_3 + kA_4 \in \mathbb{Q}^{m \times n}\) be an arbitrary quaternion matrix, then the real representation of \(A\) is specified as follows:

\[
(4.2)\quad A^R = \begin{pmatrix}
A_1 & -A_2 & -A_3 & -A_4 \\
A_2 & A_1 & -A_4 & A_3 \\
A_3 & A_4 & A_1 & -A_2 \\
A_4 & -A_3 & A_2 & A_1
\end{pmatrix}.
\]

From (4.1) and using the properties of the real representation operator (4.2), we get

\[
(U^R)^T A^R V^R = \begin{pmatrix}
\Sigma_r & 0 \\
0 & 0
\end{pmatrix}_R,
\]

which is the SVD decomposition of matrix \(A^R\). Consequently, the singular values of the quaternion matrix \(A\) and its real representation form \(A^R\) are the same.

**Definition 4.2.** The spectral condition number of a quaternion matrix \(A \in \mathbb{Q}^{m \times n}\) is defined by

\[
\text{cond}(A) = \frac{\|A\|_2}{\|A^+\|_2} = \frac{\sigma_1}{\sigma_r},
\]

where \(A^+\), stands for the Moore-Penrose pseudoinverse of the quaternion matrix \(A\).

By straightforward computations, it can be proved that \(\text{cond}(A) = \text{cond}(A^R)\). Developing the idea utilized in [28], the preconditioned form of Algorithm 3.1 to solve (1.2) can be presented. Consider the ensuing four nonsingular preconditioners

\[
Q_1 = \text{diag}(1/\|A(:,1)\|, 1/\|A(:,2)\|, \ldots, 1/\|A(:,n)\|),
\]

\[
Q_2 = \text{diag}(1/\|B(1,:)\|, 1/\|B(2,:)\|, \ldots, 1/\|B(n,:)\|),
\]

\[
Q_3 = \text{diag}(1/\|C(:,1)\|, 1/\|C(:,2)\|, \ldots, 1/\|C(:,n)\|),
\]

\[
Q_4 = \text{diag}(1/\|D(1,:)\|, 1/\|D(2,:)\|, \ldots, 1/\|D(n,:)\|),
\]

where for arbitrary given matrix \(Z\), \(\|Z(:,i)\|\) and \(\|Z(i,:)\|\) signify the Frobenius norm of \(i\)-th column and \(i\)-th row of matrix \(Z\), respectively. As discussed in [28], the idea behind these choices of preconditioners comes from an established theorem in [34, Theorem 3.5]. As a matter of fact after applying the preconditioners of the above forms, the condition numbers of the coefficient matrices of the obtained preconditioned problem become smaller; for more details see [28, 34].
Consider the equivalent form of (1.3) as follows:

\[
(4.3) \min \| AQ_1^{-1}XQ_2^{-1}Q_2B + CQ_3Q_4^{-1}YQ_4^{-1}Q_4D - E \|.
\]

Assume that

\[
\hat{A} = AQ_1, \quad \hat{B} = Q_2B, \quad \hat{C} = CQ_3, \quad \hat{D} = Q_4D, \quad \hat{X} = Q_1^{-1}XQ_2^{-1}, \quad \text{and} \quad \hat{Y} = Q_3^{-1}YQ_4^{-1}.
\]

Then (4.3) can be rewritten in the next form

\[
(4.4) \min \| \hat{A}\hat{X}\hat{B} + \hat{C}\hat{Y}\hat{D} - E \|.
\]

By employing Algorithm 3.1 to solve (4.4), we can obtain \(\hat{X}\) and \(\hat{Y}\). Afterward, the solutions of (4.3) can be obtained by setting \(X = Q_1\hat{X}Q_2\) and \(Y = Q_3\hat{Y}Q_4\).

5. Numerical experiments. In this section, we examine some test examples to numerically confirm the validity of the results elaborated in the previous sections and to demonstrate that Algorithm 3.1 is robust and effective for solving Problems 1.2 and 1.3. We comment here that all of the numerical experiments were performed on a Pentium 4 PC with a 2.67 GHz CPU and 4.00 GB of RAM using some Matlab code in MATLAB 8.1.0.604.

Example 5.1. Consider Problem 1.2 where

\[
A = \begin{pmatrix} 1 + 4i + 7j + 2k & 1 + 2i - 4j + 2k \\ 3 + i + 4j + 3k & 1 + 2i + 3j + 4k \end{pmatrix},
\]

\[
B = \begin{pmatrix} -6 + i + 5j + 8k & 3 + 2i + j + 4k \\ 7 + 9i + 3j + 2k & -2 - 3i + 4j - 5k \end{pmatrix},
\]

\[
C = \begin{pmatrix} -3 + 4i + j + 5k & 0 \\ 5 + i + 3j + 8k & 0 \end{pmatrix},
\]

\[
D = \begin{pmatrix} -7 + i + 2j - k & -1 + 2i + 9j - k \\ -3 - 3i + 2j + k & 1 + 2i + 3j + k \end{pmatrix},
\]

and

\[
E = \begin{pmatrix} -95.33 - 75.67i - 110j - 34.83k & -75 - 17.67i - 72.33j - 19.17k \\ -156.7 - 92.33i - 90j - 64.67k & -148.5 - 0.666i - 30.17j + 47.17k \end{pmatrix}.
\]

We aim to apply Algorithm 3.1 to solve Problem 1.2. In addition, for two given matrices \(\hat{X} \in iHQ^{2 \times 2}\) and \(\hat{Y} \in iAQ^{2 \times 2}\),

\[
\hat{X} = \begin{pmatrix} 3 + 5j + k & 2 - i + 2j + 2k \\ 2 + i + 2j + 2k & 1 + 3j + 4k \end{pmatrix},
\]

and
Iterative Algorithm for η-(Anti)-Hermitian Solutions of Quaternion Matrix Equations

\[ \hat{Y} = \begin{pmatrix} 4i & -2 - 0.5i - 2j - 0.5k \\ 2 - 0.5i + 2j + 0.5k & -2i \end{pmatrix}, \]

the solution pair of Problem 1.3 is determined.

It can be verified that the matrix equation \( AXB + CYD = E \) is consistent over the \( i \)-Hermitian and \( i \)-anti-Hermitian matrices as the following \( i \)-Hermitian and \( i \)-anti-Hermitian matrices satisfy \( AXB + CYD = E \),

\[ X = \begin{pmatrix} 1 + j + k & 0.5 + 0.5j + 0.5k \\ 0.5 + 0.5j + 0.5k & 0.33 + 0.33j + 0.33k \end{pmatrix}, \]

and,

\[ Y = \begin{pmatrix} i & 0.5i \\ 0.5i & 10i \end{pmatrix}. \]

In order to resolve Problem 1.2 we set \([X(0), Y(0)] = [0, 0]\) as the initial guess and exploit the subsequent stopping criterion,

\[ \| R(k) \| = \| F - AX(k)B - CY(k)D \| < 10^{-10}. \]

Utilizing Algorithm 3.1 the approximate least-norm solution pair of mentioned problem can be derived respectively as follows:

\[ X^{(21)} = \begin{pmatrix} 1 + j + k & 0.5 + 0.5j + 0.5k \\ 0.5 + 0.5j + 0.5k & 0.33 + 0.33j + 0.33k \end{pmatrix}, \]

and

\[ Y^{(21)} = \begin{pmatrix} i & 0.5i \\ 0.5i & 0 \end{pmatrix}, \]

where the corresponding residual norm is \( \| R(21) \| = \| F - AX(21)B - CY(21)D \| = 4.1507 \times 10^{-10} \). It can be seen that

\[ \| X \| + \| Y \| = 10.3118 \quad \text{and} \quad \| X^{(21)} \| + \| Y^{(21)} \| = 2.5166. \]

Therefore,

\[ \| X \| + \| Y \| > \| X^{(21)} \| + \| Y^{(21)} \|, \]

which confirms the established fact that performing Algorithm 3.1 with the initial approximate of the form (3.9) gives the least-norm solution pair of Problem 1.2. For
more details, the required iteration numbers and CPU time (sec) for the convergence and the Frobenius norm of residuals at different iterates are reported in Table 5.1.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>15</th>
<th>20</th>
<th>25</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU time</td>
<td>0.3327</td>
<td>0.4354</td>
<td>0.5618</td>
<td>0.6543</td>
</tr>
<tr>
<td>$|R(k)|$</td>
<td>0.0019</td>
<td>1.0057e-10</td>
<td>2.1703e-12</td>
<td>3.4083e-13</td>
</tr>
</tbody>
</table>

For solving Problem 1.3, Algorithm 3.1 is used to determine the solution pair of (3.15) in which

$$F = E - AXB - CYD = \begin{pmatrix} 143.2 + 203.3i - 33.5j + 143.2k & 147 + 35.83i + 155.7j + 185.3k \\ 374.8 + 278.7i + 65.5j + 163.3k & 146.5 - 56.83i + 176.8j - 158.3k \end{pmatrix}. $$

By the following choice of the initial guess $[Z(0), W(0)] = [0, 0]$, we derive

$$\|R(21)\| = \|F - AZ(21)B - CW(21)D\| = 5.0558 \times 10^{-11}. $$

The computed approximate solution pair for $[Z^*, W^*]$ is given by

$$Z^* \simeq Z^{(21)} = \begin{pmatrix} -2 - 4j & -1.5 + i - 1.5j - 1.5k \\ -1.5 - i - 1.5j - 1.5k & -0.6667 - 2.667j - 3.667k \end{pmatrix},$$

and

$$W^* \simeq W^{(21)} = \begin{pmatrix} -3i & 2 + i + 2j + 0.5k \\ -2 + i - 2j - 0.5k & 0 \end{pmatrix}. $$

Using the presented discussion in Subsection 3.2, the solution of Problem 1.3 is calculated by

$$X^* = \hat{X} + Z^* \simeq \begin{pmatrix} 1 + j + k & 0.5 + 0.5j + 0.5k \\ 0.5 + 0.5j + 0.5k & 0.33 + 0.33j + 0.33k \end{pmatrix},$$

and,

$$Y^* = \hat{Y} + W^* \simeq \begin{pmatrix} i & 0.5i \\ 0.5i & -2i \end{pmatrix}. $$

For more clarification, we exhibit the convergence curves of the methods in Figures 5.1 where

$$\varepsilon(k) = \log_{10} \|R(k)\|.$$
Iterative Algorithm for $\eta$-(Anti)-Hermitian Solutions of Quaternion Matrix Equations

Our goal is to find the least-norm solution pair of Problem 1.2.

**Example 5.2.** In this instance, we focus on Problem 1.2 such that

\[
A = A_1 + A_2i + A_3j + A_4k, \quad B = B_1 + B_2i + B_3j + B_4k, \quad C = C_1 + C_2i + C_3j + C_4k, \quad D = D_1 + D_2i + D_3j + D_4k, \quad E = E_1 + E_2i + E_3j + E_4k,
\]

where

- $A_1 = \text{triu}($hilb($n$)$)$, $A_2 = \text{triu}($ones($n, n$)$)$, $A_3 = \text{eye}($n$)$, $A_4 = \text{zeros}($n, n$)$,

- $B_1 = \text{tridiag}($n, $-1, 2, -1)$, $B_2 = \text{eye}($n$)$, $B_3 = \text{zeros}($n, n$)$, $B_4 = \text{tridiag}(n, 0.5, 6, -0.5)$,

- $C_1 = C_2 = C_3 = C_4 = \text{ones}($n, n$)$, $D_1 = D_2 = D_3 = D_4 = \text{ones}($n, n$)$,

- $E_1 = \text{hankel}(1 : n), E_2 = \text{zeros}($n, n$), E_3 = \text{zeros}($n, n$), E_4 = \text{zeros}($n, n$)$.

Our goal is to find the least-norm solution pair of Problem 1.2.

It can be seen that the matrix equation $AXB + CYD = E$ mentioned in Example 5.2 is not consistent over $k$-Hermitian and $k$-anti-Hermitian matrices. From Theorem 3.10 it is known that if the initial matrix group $[X(0), Y(0)]$ are chosen so that $[X(0), Y(0)] \in \mathcal{W}$, then the least-norm solution pair can be obtained. We apply
Algorithm 3.1 to solve Problem 1.2 for different values of $n$ with $[X(0), Y(0)] = [0, 0]$. The iterations are stopped as soon as

$$\eta(k) = \log_{10} \sqrt{\frac{\|P_{k,x}\|^2 + \|P_{k,y}\|^2}{\|P_{0,x}\|^2 + \|P_{0,y}\|^2}} < 10^{-5}. \tag{5.2}$$

The details of numerical results including the required iteration numbers and the Frobenius norm of residuals are recorded in Table 5.2 where $R(k) = F - AX(k)B - CY(k)D$. The convergence histories are depicted in Figure 5.2 where $\eta(k)$ defined by (5.2).

<table>
<thead>
<tr>
<th>$n$</th>
<th>$|X(k)| + |Y(k)|$</th>
<th>$\min |R(k)|$</th>
<th>Iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>13.3815</td>
<td>27.9922</td>
<td>76</td>
</tr>
<tr>
<td>40</td>
<td>38.8499</td>
<td>65.7652</td>
<td>178</td>
</tr>
<tr>
<td>60</td>
<td>107.3637</td>
<td>71.9070</td>
<td>287</td>
</tr>
</tbody>
</table>

**Table 5.2**

**Numerical results for Example 5.2**

**Fig. 5.2. Convergence history for Example 5.2**

**Remark 5.3.** We would like to comment here that it is not possible to use the presented scheme in [40] for $n = 40, 60$ during the programming process due to
Iterative Algorithm for \( \eta \)-(Anti)-Hermitian Solutions of Quaternion Matrix Equations

the required high computational costs. Whereas Algorithm 3.1 can be successfully handled which reveals the superiority of the proposed algorithm.

Example 5.4. In this example, we examine the efficiency of Algorithm 3.1 for solving sparse quaternion least-squares problems. The test matrices were chosen from HB (Harwell-Boeing) group of Tim Davis’s collection [10]. The four test matrices utilized in this experiment appear in structural problem (\texttt{bcsstm01}) and directed weighted graph problems (\texttt{gre_115}, \texttt{gre_185} and \texttt{gre_343}). The sparsity structure of mentioned matrices are illustrated in Figure 5.3 where “nz” stands for the number of nonzero entries of the matrix. In addition, the properties of the test matrices are presented in Table 5.3 for more details.

<table>
<thead>
<tr>
<th>Matrix</th>
<th>number of rows</th>
<th>number of columns</th>
<th>structure</th>
<th>condition number</th>
</tr>
</thead>
<tbody>
<tr>
<td>\texttt{bcsstm01}</td>
<td>48</td>
<td>48</td>
<td>symmetric</td>
<td>Inf</td>
</tr>
<tr>
<td>\texttt{gre_115}</td>
<td>115</td>
<td>115</td>
<td>unsymmetric</td>
<td>49.6709</td>
</tr>
<tr>
<td>\texttt{gre_185}</td>
<td>185</td>
<td>185</td>
<td>unsymmetric</td>
<td>103.015</td>
</tr>
<tr>
<td>\texttt{gre_343}</td>
<td>343</td>
<td>343</td>
<td>unsymmetric</td>
<td>111.976</td>
</tr>
</tbody>
</table>

The main objective is to find the solution pair of Problem 1.2 associated with the matrix equation \( AXB + CYD = E \) such that

\[
A = A_1 + A_2i + A_3j + A_4k, \quad B = B_1 + B_2i + B_3j + B_4k,
\]

\[
C = C_1 + C_2i + C_3j + C_4k, \quad D = D_1 + D_2i + D_3j + D_4k,
\]

\[
E = E_1 + E_2i + E_3j + E_4k.
\]

Presume that

\[
A_1 = B_1 = C_1 = D_1, \quad A_2 = B_2 = C_2 = D_2,
\]

\[
A_3 = B_3 = C_3 = D_3, \quad A_4 = B_4 = C_4 = D_4.
\]

The right hand side \( E \) is constructed such that \( X = \mathcal{L}_1(C) \) and \( Y = \mathcal{L}_2(D) \) satisfy \( AXB + CYD = E \). Hence, the problem is consistent over \( j \)-Hermitian and \( j \)-anti-Hermitian matrices. The following four instances are examined:

Case 1:

\[
A_1 = \text{tridiag}(-1, 2, -1), \quad A_2 = \text{bcsstm01}, \quad A_3 = \text{tridiag}(1, 3, 1), \quad A_4 = \text{bcsstm01}.
\]
Fig. 5.3. The sparsity structure of test matrices.

Case 2:

\[ A_1 = \text{tridiag}(-1, 2, -1), \ A_2 = \text{gre} \cdot 115, \ A_3 = \text{tridiag}(1, 3, 1), \ A_4 = \text{gre} \cdot 115. \]

Case 3:

\[ A_1 = \text{tridiag}(-1, 2, -1), \ A_2 = \text{gre} \cdot 185, \ A_3 = \text{tridiag}(1, 3, 1), \ A_4 = \text{gre} \cdot 185. \]

Case 4:

\[ A_1 = \text{tridiag}(-1, 2, -1), \ A_2 = \text{gre} \cdot 343, \ A_3 = \text{tridiag}(1, 3, 1), \ A_4 = \text{gre} \cdot 343. \]

We employed Algorithm 3.1 and its preconditioned version for Example 5.4, the obtained results are presented in the following figures. For Case 1, Algorithm 3.1 has slow convergence rate than preconditioned version since the matrix bcsstm01 is an ill-conditioned matrix. The convergence histories of Algorithm 3.1 and its preconditioned version for this case are depicted in Figure 5.4 where \( \varepsilon(k) \) specified by (5.1). However,
for Cases 2–4, the convergence rate of Algorithm 3.1 and its preconditioned version is almost equal as the coefficients are well conditioned. The convergence history of Algorithm 3.1 for Cases 2–4 is plotted in Figure 5.5.

---

**Fig. 5.4.** Convergence history for Case 1 of Example 5.4

**Fig. 5.5.** Convergence history for Cases 2, 3, 4 of Example 5.4
6. Conclusion and further works. The paper has been concerned with solving two main problems referred Problem 1.2 and Problem 1.3. More precisely, Problem 1.2 deals with solving the least-squares problem

\[ \| AXB + CYD - E \| = \min, \]

over \( X \in \eta HQ^{n \times n} \) and \( Y \in \eta AQ^{n \times n} \) matrices. The main aim of the second problem was to obtain the optimal approximate solution pair of Problem 1.2 with respect to an arbitrary given matrix pair. After establishing some properties of the \( \eta \)-Hermitian and \( \eta \)-anti-Hermitian matrices, the well-known CGLS iterative algorithm has been developed to construct an efficient and robust algorithm for solving Problems 1.2 and 1.3. The convergence properties of the algorithm have been analyzed and a preconditioned version of the algorithm has been suggested. Numerical results have illustrated the efficiency and feasibility of the presented algorithm. The proposed iterative scheme can be generalized for solving the following generalized quaternion least-squares problem over \( \eta \)-Hermitian and \( \eta \)-anti-Hermitian matrices,

\[ \min \left\| \sum_{j=1}^{n} \sum_{i=1}^{n} \left( A_i X_i B_i + C_i X_i^H D_i - E_i \right) \right\|. \]

Future work can investigate applying suitable preconditioners to speed up the convergence of the CGLS algorithm for solving the least-squares problems corresponding to coupled matrix equations.

Acknowledgment. The authors wish to thank anonymous referees for their helpful comments.

REFERENCES

Iterative Algorithm for $\eta$-(Anti)-Hermitian Solutions of Quaternion Matrix Equations


Appendix A. Proof of Theorem 3.6  
Because of commutative property of the inner product $\langle . , . \rangle$, we only need to prove the validity of Eqs. (3.6), (3.7) and (3.8) for $1 \leq i < j \leq k$. For $k = 1$, we have

$$
\langle P_{0,x}, P_{1,x} \rangle + \langle P_{0,y}, P_{1,y} \rangle
= \langle P_{0,x}, P_{0,x} - \alpha_0 L_1^1 \left( A^T (M(Q_{0,x}, Q_{0,y})) B^T \right) \rangle
+ \langle P_{0,y}, P_{0,y} - \alpha_0 L_2^2 \left( C^T (M(Q_{0,x}, Q_{0,y})) D^T \right) \rangle
= \| P_{0,x} \|^2 + \| P_{0,y} \|^2 - \alpha_0 \langle A P_{0,x} B, M(Q_{0,x}, Q_{0,y}) \rangle - \alpha_0 \langle C P_{0,y} D, M(Q_{0,x}, Q_{0,y}) \rangle
= \| P_{0,x} \|^2 + \| P_{0,y} \|^2 - \alpha_0 \langle M(Q_{0,x}, Q_{0,y}), M(Q_{0,x}, Q_{0,y}) \rangle
= 0.
$$
Iterative Algorithm for \(\eta\)-(Anti)-Hermitian Solutions of Quaternion Matrix Equations

\[
\langle M(Q_{0,x}, Q_{0,y}), M(Q_{1,x}, Q_{1,y}) \rangle \\
= \langle M(Q_{0,x}, Q_{0,y}), M(P_{1,x} + \beta_0 Q_{0,x}, P_{1,y} + \beta_0 Q_{0,y}) \rangle \\
= \beta_0 \| M(Q_{0,x}, Q_{0,y}) \|^2 + \langle M(Q_{0,x}, Q_{0,y}), M(P_{1,x}, P_{1,y}) \rangle \\
= \beta_0 \| M(Q_{0,x}, Q_{0,y}) \|^2 \\
\quad + \frac{1}{\alpha_0} \left( \langle M_1^* (R_0 - R_1), P_{1,x} \rangle + \langle M_2^* (R_0 - R_1), P_{1,y} \rangle \right) \\
= \beta_0 \| M(Q_{0,x}, Q_{0,y}) \|^2 \\
\quad + \frac{1}{\alpha_0} \left( \langle L_0^* (M_1^* (R_0 - R_1), P_{1,x}) + \langle L_2^* (M_2^* (R_0 - R_1)), P_{1,y} \rangle \right) \\
= \beta_0 \| M(Q_{0,x}, Q_{0,y}) \|^2 \\
\quad + \frac{1}{\alpha_0} (\langle P_{0,x} - P_{1,x}, P_{1,x} \rangle + \langle P_{0,y} - P_{1,y}, P_{1,y} \rangle) \\
= 0.
\]

Evidently, it can be seen that

\[
\langle P_{1,x}, Q_{0,x} \rangle + \langle P_{1,y}, Q_{0,y} \rangle = \langle P_{1,x}, P_{0,x} \rangle + \langle P_{1,y}, P_{0,y} \rangle = 0.
\]

Hence, the equations (3.6)–(3.8) are valid for \(k = 1\). Suppose that assertions (3.6)–(3.8) are true for \(i \leq s\). To complete the proof, in the sequel, we demonstrate the validity of the conclusions for \(k = s + 1\). For \(i < s\), it can be deduced that

\[
\langle P_{i,x}, P_{s+1,x} \rangle + \langle P_{i,y}, P_{s+1,y} \rangle \\
= \langle P_{i,x}, P_{s,x} - \alpha_s L_1^* (A^H (M(Q_{s,x}, Q_{s,y})) B^H) \rangle \\
\quad + \langle P_{i,y}, P_{s,y} - \alpha_s L_2^* (C^H (M(Q_{s,x}, Q_{s,y})) D^H) \rangle \\
= -\alpha_s \langle M_1(P_{i,x}), M(Q_{s,x}, Q_{s,y}) \rangle \\
\quad - \alpha_s \langle M_2(P_{i,y}), M(Q_{s,x}, Q_{s,y}) \rangle \\
= -\alpha_s \langle M_1(Q_{i,x} - \beta_{i-1} Q_{i-1,x}), M(Q_{s,x}, Q_{s,y}) \rangle \\
\quad - \alpha_s \langle M_2(Q_{i,y} - \beta_{i-1} Q_{i-1,y}), M(Q_{s,x}, Q_{s,y}) \rangle \\
= 0,
\]

Electronic Journal of Linear Algebra  ISSN 1081-3810
A publication of the International Linear Algebra Society
http://math.technion.ac.il/iic/ela
\( \langle M(Q_{1,x}, Q_{1,y}) , M(Q_{s+1,x}, Q_{s+1,y}) \rangle \)
\[
= \langle M(Q_{1,x}, Q_{1,y}) , M(P_{s+1,x} + \beta_s Q_{s,x} , P_{s+1,y} + \beta_s Q_{s,y}) \rangle \\
= \langle M(Q_{1,x}, Q_{1,y}) , M(P_{s+1,x}, P_{s+1,y}) \rangle \\
= \frac{1}{\alpha_i} \langle (R(i) - R(i + 1), M(P_{s+1,x}, P_{s+1,y}) \rangle \\
= \frac{1}{\alpha_i} \langle M_1^*(R(i) - R(i + 1)), P_{s+1,x} \rangle \\
\quad + \frac{1}{\alpha_i} \langle M_2^*(R(i) - R(i + 1)), P_{s+1,y} \rangle \\
= \frac{1}{\alpha_i} \langle \left[ \langle L_1^0(Q_1,x) - M_1^*(R(i + 1)) \rangle , P_{s+1,x} \right] \\
\quad + \frac{1}{\alpha_i} \langle \left[ \langle L_2^0(Q_2,x) - M_2^*(R(i + 1)) \rangle , P_{s+1,y} \right] \\
= -\frac{1}{\alpha_i} (\langle P_{s+1,x}, P_{s+1,y} \rangle + \langle P_{s+1,y}, P_{s+1,y} \rangle) = 0,
\]

and,
\[
\langle Q_{1,x}, P_{s+1,x} \rangle + \langle Q_{1,y}, P_{s+1,y} \rangle \\
= \langle Q_{1,x}, P_{s,x} - \alpha_s L_1^0(A^H(M(Q_{s,x}, Q_{s,y})) B^H) \rangle \\
\quad + \langle Q_{1,x}, P_{s,x} - \alpha_s L_2^0(C^H(M(Q_{s,x}, Q_{s,y})) D^H) \rangle \\
= -\alpha_s \langle M_1(Q_{1,x}, M(Q_{s,x}, Q_{s,y})) \rangle - \alpha_s \langle M_2(Q_{1,y}, M(Q_{s,x}, Q_{s,y})) \rangle \\
= -\alpha_s \langle M(Q_{1,x}, Q_{1,y}), M(Q_{s,x}, Q_{s,y}) \rangle = 0.
\]

For \( i = s \), we have
\[
\langle P_{s,x}, P_{s+1,x} \rangle + \langle P_{s,y}, P_{s+1,y} \rangle \\
= \langle P_{s,x}, P_{s,x} - \alpha_s L_1^0(A^H(M(Q_{s,x}, Q_{s,y})) B^H) \rangle \\
\quad + \langle P_{s,y}, P_{s,y} - \alpha_s L_2^0(C^H(M(Q_{s,x}, Q_{s,y})) D^H) \rangle \\
= \|P_{s,x}\|^2 + \|P_{s,y}\|^2 - \alpha_s \langle M_1(P_{s,x}), M(Q_{s,x}, Q_{s,y}) \rangle \\
\quad - \alpha_s \langle M_2(P_{s,y}), M(Q_{s,x}, Q_{s,y}) \rangle \\
= \|P_{s,x}\|^2 + \|P_{s,y}\|^2 - \alpha_s \langle M(P_{s,x}, P_{s,y}), M(Q_{s,x}, Q_{s,y}) \rangle \\
= \|P_{s,x}\|^2 + \|P_{s,y}\|^2 - \alpha_s \langle M(Q_{s,x}, Q_{s,y}), M(Q_{s,x}, Q_{s,y}) \rangle = 0,
\]
Iterative Algorithm for \(\eta\)-(Anti)-Hermitian Solutions of Quaternion Matrix Equations

\[
\langle M(Q_{s,x}, Q_{s,y}) , M(Q_{s+1,x}, Q_{s+1,y}) \rangle
= \langle M(Q_{s,x}, Q_{s,y}) , M(P_{s+1,x} + \beta_s Q_{s,x}, P_{s+1,y} + \beta_s Q_{s,y}) \rangle
= \beta_s \| M(Q_{s,x}, Q_{s,y}) \|^2 + \langle M(Q_{s,x}, Q_{s,y}) , M(P_{s+1,x}, P_{s+1,y}) \rangle
= \beta_s \| M(Q_{s,x}, Q_{s,y}) \|^2 + 1 \frac{\alpha_s}{\alpha_s} \langle R(s) - R(s+1), M(P_{s+1,x}, P_{s+1,y}) \rangle
= \beta_s \| M(Q_{s,x}, Q_{s,y}) \|^2 + 1 \frac{\alpha_s}{\alpha_s} \langle \mathcal{M}_1^*(R(s) - R(s+1)), P_{s+1,x} \rangle
+ 1 \frac{\alpha_s}{\alpha_s} \langle \mathcal{M}_2^*(R(s) - R(s+1)), P_{s+1,y} \rangle
= \beta_s \| M(Q_{s,x}, Q_{s,y}) \|^2 + 1 \frac{\alpha_s}{\alpha_s} \langle \mathcal{L}_1^*(\mathcal{M}_1^*(R(s) - R(s+1))), P_{s+1,x} \rangle
+ 1 \frac{\alpha_s}{\alpha_s} \langle \mathcal{L}_2^*(\mathcal{M}_2^*(R(s) - R(s+1))), P_{s+1,y} \rangle
= \beta_s \| M(Q_{s,x}, Q_{s,y}) \|^2 + 1 \frac{\alpha_s}{\alpha_s} \langle P_{s,x} - P_{s+1,x}, P_{s+1,x} \rangle
+ 1 \frac{\alpha_s}{\alpha_s} \langle P_{s,y} - P_{s+1,y}, P_{s+1,y} \rangle = 0,
\]

and,

\[
\langle Q_{s,x}, P_{s+1,x} \rangle + \langle Q_{s,y}, P_{s+1,y} \rangle
= \langle Q_{s,x}, P_{s,x} - \alpha_s \mathcal{L}_1^*(A^H M(Q_{s,x}, Q_{s,y}) B^H) \rangle
+ \langle Q_{s,y}, P_{s,y} - \alpha_s \mathcal{L}_2^*(C^H M(Q_{s,x}, Q_{s,y}) D^H) \rangle
= \| P_{s,x} \|^2 + \| P_{s,y} \|^2 - \alpha_s \langle Q_{s,x}, A^H M(Q_{s,x}, Q_{s,y}) B^H \rangle
- \alpha_s \langle Q_{s,y}, C^H M(Q_{s,x}, Q_{s,y}) D^H \rangle
= \| P_{s,x} \|^2 + \| P_{s,y} \|^2 - \alpha_s \| M(Q_{s,x}, Q_{s,y}) \|^2 = 0.
\]

Now the result can be concluded from the principle of mathematical induction.